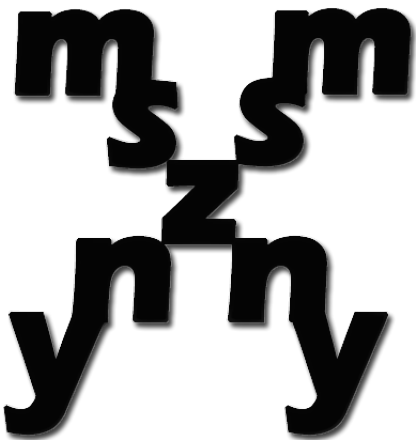
XXII. Magyar Számítógépes Nyelvészeti Konferencia



Szerkesztette:
Berend Gábor
Gosztolya Gábor
Vincze Veronika

Szeged, 2026. január 29-30.

Szerkesztette:
Berend Gábor, Gosztolya Gábor, Vincze Veronika
{berendg,ggabor,vinczev}@inf.u-szeged.hu

Felelős kiadó:
Szegedi Tudományegyetem
TTIK, Informatikai Intézet
6720 Szeged, Árpád tér 2.

ISBN: 978-963-688-100-9

Szeged, 2026. február

Az MSZNY 2026 konferencia szervezője:
HUN-REN–SZTE Mesterséges Intelligencia Kutatócsoport

Támogatók:
Neumann János Számítógéptudományi Társaság
SZTE Informatikai Intézet

Előszó

2026. január 29–30-án immáron huszonkettedik alkalommal kerül sor a Magyar Számítógépes Nyelvészeti Konferencia megrendezésére. A konferencia fő célkitűzése a kezdetek óta állandó: lehetőséget biztosítani a nyelv- és beszédtechnológia területén végzett kutatások eredményeinek ismertetésére és megvitatására, ezen felül pedig a különféle hallgatói projektek, illetve ipari alkalmazások bemutatására.

Az idei évben a 18 beküldött cikkből gondos mérlegelést követően 15 cikk került elfogadásra, melyek témája a nyelv- és beszédtechnológia számos szakterületét lefedi a legújabb nyelvi modellek bemutatásától kezdve a beszédtechnológia eredményein keresztül egészen a magyarra adaptált nagy multimodális modellek erőforráshatékony létrehozásáig.

A konferencia programját két plenáris előadás is gazdagítja, amelyeket Vincze Veronika és Török Balázs fog megtartani. Vincze Veronika meghívásának apropóját az adta, hogy a közelmúltban kihirdetett, HTE Top 50 Nők a Mesterséges Intelligencia Területén elnevezésű díj 1. helyezettje lett. Előadásának címe *Szólj, s ki vagy, elmondom - A nyelvhasználati jellegzetességek szerepe osztályozási feladatokban.* A konferencia második napján Török Balázs plenáris előadását hallgathatjuk meg a MOZAIK Kiadó képviseletében *Nagy nyelvi modellek alkalmazása számonkérésben és digitális tananyagfejlesztésben* címmel.

A korábbi szokásoknak megfelelően díjazzuk a konferencia legjobb cikkét, mely a legjelentősebb eredményekkel járul hozzá a magyarországi nyelv- és beszédtechnológiai kutatásokhoz. Ezen felül immár nyolcadik alkalommal osztjuk ki a legjobb bírálónak járó díját, amellyel a bírálók fáradságos, ugyanakkor nélkülözhetetlen munkáját kívánjuk elismerni.

A szervezőbizottság nevében,
Ács Judit,
Berend Gábor,
Gosztolya Gábor,
Ligeti-Nagy Noémi,
Nemeskey Dávid Márk,
Novák Attila,
Simon Eszter,
Sztahó Dávid,
Vincze Veronika

Tartalomjegyzék

Nyelvmodellek

Nagy nyelvi modellek által generált szövegek felismerése magyar nyelven

Kiss Mihály[1], Berend Gábor[1]

Szegedi Tudományegyetem, Informatikai Intézet
kiss.mihaly@stud.u-szeged.hu, berendg@inf.u-szeged.hu

Kivonat Manapság rengeteg mesterséges intelligencia (MI) által generált tartalom keletkezik, amit az is elősegít, hogy különböző modellek, mint például a ChatGPT, Claude vagy Gemini, széles körben elérhetők. Ebből adódóan egyre nagyobb szükség van olyan megoldásokra, amelyek képesek az ember és gép által generált tartalmak megkülönböztetésére. Míg angol nyelven már számos megoldás létezik, magyar nyelven egyelőre nem született igazán érdemi kutatás. A cikkben ezért kifejezetten magyar nyelvre készített és vizsgált modelleket mutatunk be az ember és MI által generált szövegek elkülönítésére.
Magyar nyelvre finomhangolt enkóder modelleket használtunk, öt különböző modell teljesítményét összehasonlítva. A modellek hasonló F1 értéket értek el. A szavaztatás a publikus korpuszon nem javított, a heterogénebb kiegészített korpuszon viszont növelte az F1-pontszámot.
Az eredményeket ipari detektorokkal is összevetettük, és azt találtuk, hogy a finomhangolt modellek több kereskedelmi megoldásnál is jobb teljesítményt nyújtanak, és csak egyetlen ipari detektor előzi meg őket.
Kulcsszavak: MI, detektálás, predikció, enkóder

1. Bevezetés

Az elmúlt években rengeteget fejlődtek a nagy nyelvi modellek, így az általuk generált szövegek sokkal jobban hasonlítanak az emberek által írt szövegekre, mint korábban. Manapság már rengeteg MI-generált szöveg keletkezik, és ez már a tudományos életben is észlelhető, egyre több publikációban, cikkben lehet szintetikus szöveget találni (Liang és mtsai, 2024b). Korábbi kutatások kimutatták, hogy az MI-konferenciákra beküldött cikkek 6,5–16,9%-a erősen támaszkodhat nagy nyelvi modellekre, vagy akár teljesen át is írhatták azokat, ami jól mutatja az egyre növekvő használatot a tudományos közegben (Liang és mtsai, 2024a). Ezt a trendet tovább erősíti, hogy a fejlett modellek széles körben elérhetők, és ma már minimális technikai tudással rendelkező felhasználók is hozzáférhetnek, használhatják azokat.

A fentiek miatt egyre nagyobb a szükség az olyan automatikus megoldásokra, amelyek képesek megkülönböztetni az ember által írt szövegeket a szintetikusan létrehozott szövegektől. Ezt az igényt számos közelmúltbeli verseny és workshop hangsúlyozza, amelyek kifejezetten erre a problémára készültek (Sarvazyan

és mtsai, 2023; Wang és mtsai, 2024; Chamezopoulos és mtsai, 2024; Wang és mtsai, 2025).

Noha egyre jobb minőségű megoldásokkal találkozhatunk, az MI segítségével létrehozott és az emberi szövegek megkülönböztetése sok kihívást rejt magában. A legjobb módszerek általában mély neurális hálózatú architektúrákra épülnek, legtöbbször transformer alapú modellek, amelyeket finomhangolnak nagyobb, annotált adathalmazokon (Sarvazyan és mtsai, 2024; Marchitan és mtsai, 2024). Frissebb tanulmányok megmutatták, hogy ezeknek a transformer modelleknek kombinálással vett valószínűségi kimenete még tovább javíthatja az eredményeket és a robusztusságot az MI-generált szövegek felismerésében (Gu és Meng, 2024; Kiss és Berend, 2025).

A területtel foglalkozó cikkek túlnyomó többségének fókuszában az angol nyelvre vonatkozó megoldások állnak. Jelen cikk elsődleges motivációja, hogy a magyar nyelven generált szövegek felismerésének problémáját – legjobb tudomásunk szerint – elsőként vizsgáljuk tudományos igényességgel, amely keretein belül egy új publikus teszthalmazt is létrehoztunk. A cikkben kettő megoldás eredményeit mutatjuk be. Az első megoldást három publikus adathalmazból összegyűjtött szövegekre, a másikat pedig egy korábban saját szkriptekkel gyűjtött adathalmazra támaszkodva hoztuk létre.

Az eredmények kiértékelése során fontos metrika a fals pozitív ráta (FPR) és a fals negatív ráta (FNR). Az FPR azt jelzi, hogy a modell milyen gyakran sorol tévesen emberi szöveget MI-generáltnak, míg az FNR azt mutatja meg, hogy az MI által generált szövegek mekkora részét nem ismeri fel. Ebben a doménben az egyik legfontosabb, hogy emberi szöveget ne osztályozzunk MI-generáltnak, ezért érdemes minél alacsonyabb fals pozitív rátára törekedni.

A két adathalmazra kapott eredmények azt mutatják, hogy a publikus adathalmazon finomhangolt modellek teljesítménye nem kiemelkedő, míg a saját gyűjtésű, kiegészített adathalmaz jóval robusztusabb és pontosabb modelleket eredményezett. A szavaztatásos módszerek a publikus adatokon nem hoztak érdemi javulást, a kiegészített adatokon azonban csökkentették a fals pozitív rátát és növelték az F1-pontszámot.

2. Kapcsolódó irodalom

A gép által generált szövegek detektálásában elért legújabb fejlesztések kiemelték a modellek szavaztatásával nyert (ensemble) tanulási módszerek hatékonyságát (Gu és Meng, 2024; Kiss és Berend, 2025). Gu és Meng (2024) finomhangolt transformer modellek szavaztatásával vett részt SemEval-2024 8. versenykiírásán (Wang és mtsai, 2024). Megközelítésük hatékonyan kezelte a kiegyensúlyozatlan címkék problémáját, amivel kimagasló teljesítményt voltak képesek elérni.

Mivel a standard ensemble módszerek magas számítási költséggel járnak, amelyek több független modell betanítását igénylik, számos megközelítést javasoltak a több modell döntésére építő megoldások erőforrásigényének csökkentésére. Az egyik ilyen módszer az ún. *snapshot ensemble* technika (Huang és mtsai, 2017), amely a finomhangolás különböző időpontjaiban rögzíti a modell aktuális

állapotának "pillanatfelvételeit". Bár ez a megközelítés csökkenti a finomhangolás költségeit azáltal, hogy elkerüli a különálló modellek betanításának szükségességét, általában nem éri el a standard ensemble módszer teljesítményét.

A felügyelt tanításon alapuló modellek mellett népszerű megközelítést jelentenek a gépileg generált szövegek felismerésének feladatában az ún. *zero-shot* módszerek (Mitchell és mtsai, 2023; Bao és mtsai, 2024). Ezen módszerek nem igényelnek finomhangolást, ehelyett (nagy) generatív nyelvi modelleknek az egyes szövegekhez társított valószínűségek elemzésére támaszkodnak.

3. Adathalmazok

3.1. Publikus adathalmaz

A publikus adathalmaz három másik adathalmazból lett készítve mintavételezéssel. Ezek az adathalmazok nyilvánosak, és elérhetőek. Az ember által írt szövegeket két forrásból szereztük be: egyrészt a magyar Wikipedia szócikkeiből Niklaus (2023), másrészt pedig a Hungarian Webcorpus 2 Nemeskey (2020) korpuszban található webes tartalmakból történő mintavételezésre támaszkodtunk. Ezáltal az adathalmazban reprezentálva vannak általános internetről kinyert szövegek, illetve tudományos szöveg is. Ebből a két adathalmazból lett véletlenszerűen kiválasztva 5000 szöveg, egyenlő arányban. A gép által generált szövegekre vonatkozó tanítóadat a boapps (2024) adathalmazból származik, amelyben magyar nyelven megfogalmazott utasításokra nagy nyelvi modellek által generált válaszok szerepelnek. Ebből az adathalmazból szintén kiválasztottunk 5000 szöveget, és így végső formában megkaptuk a 10,000 szövegből álló adathalmazunkat. A szövegek hosszának az elemzése a 2. ábrán látható. Egy rövid példa az MI generált szövegekből:

Kérdés: Írj egy rövid mondást az időről és annak múlásáról.

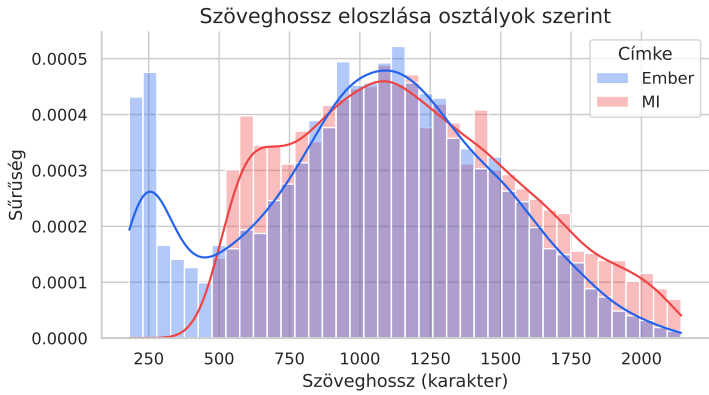
Válasz: Az idő olyan, mint a homok a tenyeredben – mindig csak úgy vesszük észre, hogy kicsúszik közöttünk.

1. ábra: Példa MI generált válaszra.

Alapvetően elmondható, hogy sok rövid szöveg is megtalálható az adathalmazban, amelyekről általában nehezen megmondható, hogy ki írta valójában.

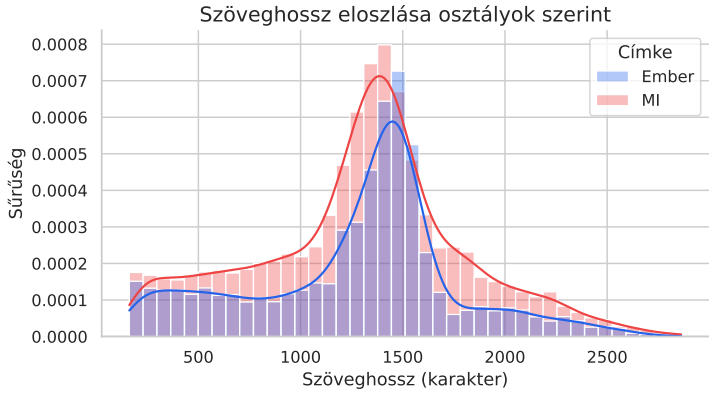
3.2. Kiegészített adathalmaz

A modellek fejlesztéséhez egy általunk összeállított kiegészített adathalmaz is felhasználásra került. A saját gyűjtésű adathalmazt kiegészítettnek nevezzük, mivel célja a publikus adathalmazban nem reprezentált szövegtípusok és nyelvi



2. ábra: Publikus tanítóhalmaz szövegek hosszának elemzése

jellemzők pótlása. Az adatgyűjtés és annotáció teljes mértékben a saját erőforrásainkból valósult meg. Az elmondható, hogy az adathalmaz diverz, sok témakört lefed. Ez alatt lehet érteni fórumok, esszék, tudományos cikkek, beadandókból előállított szövegeket. A kiegészített adathalmaz célja az volt, hogy a publikus adathalmazból hiányzó sokszínűséget lefedje, illetve az is, hogy megmutassuk azt, hogy egy minőségi adathalmazon finomhangolt modell jobb eredményt fog nyújtani. A szövegek hosszára vonatkozó statisztikák a 3. ábrán láthatók.

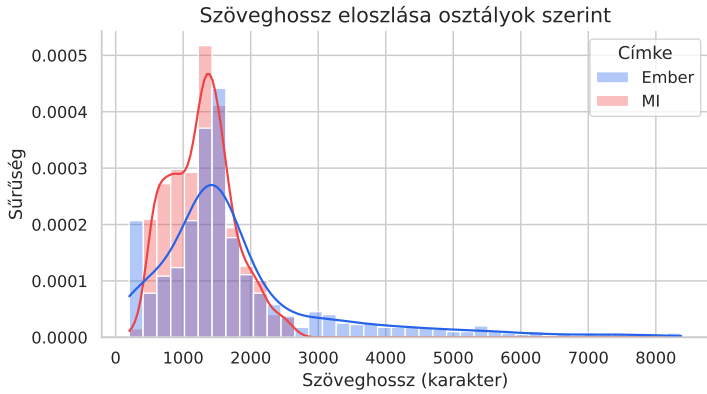


3. ábra: Kiegészített tanítóhalmaz szövegeinek hosszának elemzése

3.3. Teszt adathalmaz

A teszt adathalmaz a kiegészített illetve a publikus adathalmaz keveréke. Mindkettőből 1000-1000 szöveg lett beletéve, azon belül pedig 500-500 darab az em-

ber és az MI osztályból. Ezáltal a teszthalmaz diverz, amelyben mindkét kísérlet adathalmaza reprezentálva van, tehát mindkettő kísérletnek azonosan kedvez. A szövegek hossza a 4. ábrán látható. A teszt adathalmaz nyilvánosan elérhető[1].



4. ábra: Teszthalmaz szövegek hosszának elemzése

4. Felhasznált modellek

A kísérletekben több, magyar nyelvre finomhangolt transformer alapú modell nyelvi modellt használtunk. Ezáltal nem csak az adathalmazok minőségét tudjuk vizsgálni, hanem a modellek különböző előtanítási megoldásait is. A huBERT (Nemeskey, 2021), PULI-BERT-Large (Yang és mtsai, 2023) és a huDeBERTa-MLM modellek klasszikus, maszkolt nyelvi modellezésen (MLM) alapuló előtanítással készültek, amely a BERT-jellegű modellek általános előtanítási feladata. A huDeBERTa-MLSM illetve a huDeBERTa-LSM a standard maszkolt nyelvi modellezéses előtanítási feladat helyett látens szemantikus tulajdonságok modellezésére lettek tanítva a Berend (2024) által bevezetett megoldások szerint.

A huDeBERTa prefixű modellek azzal az előnyös tulajdonsággal rendelkeznek, hogy azok az előtanítás célfüggvényét leszámítva teljesen azonos módon lettek létrehozva (azonos modellarchitektúra, azonos előtanító korpusz, illetve azonos előtanítási hiperparaméterek használata mellett), így az ezen modellek finomhangolásával nyert eredmények összehasonlítása lehetővé teszi az előtanítás megválasztásának szerepének vizsgálatát.

A vizsgált modellek méreteiben jelentős különbségek vannak: a PULI-BERT-Large egy 24 rétegű modell 340 millió paraméterrel, míg a huBERT-base és a huDeBERTa család modelljei 12 rétegből állnak, előbbi 110, utóbbi pedig 134 millió paraméterrel. A különböző modellek használata lehetővé teszi annak vizsgálatát

[1] `https://huggingface.co/datasets/mihalykiss/Hun_ai_test`

is, hogy a modellek méretei milyen mértékben járulnak hozzá az MI-generált szövegek felismerésének pontosságához.

5. Kísérletek

5.1. Modellek finomhangolása

Az előző pontban bemutatott modelleket mindkét adathalmaz alapján finomhangoltuk. A finomhangolások során alkalmazott hiperparamétereket a tanítóadattól és az aktuálisan finomhangolt modelltől függetlenül, azonos módon választottuk meg a kapott eredmények összehasonlíthatósága kedvéért. Az adatbázisokat minden esetben azonos módon – 80% tanító, 20% validációs halmaz – osztottuk fel.

A finomhangolás során minden epoch végén elvégeztünk egy validációs halmazon történő kiértékelést. Alapértelmezetten modellek finomhangolása 5 epochon keresztül zajlott, de lehetőség volt az ennél korábban történő megállásra (early stopping) a validációs halmazon elért F1-pontszám figyelembe vétele alapján. Minden alapmodellel 3 finomhangolási kísérletet hajtottunk végre úgy, hogy a modelleknek az osztályozásért felelős súlyait különböző véletlenszerű inicializációval hoztuk létre. Egy finomhangoláson belül a validációs halmazon legjobb eredményt elérő modellt tartottuk meg.

Mindegyik vizsgált enkóder alapmodell rendelkezik egy 512 tokenes bemeneti korláttal, amennyiben egy input szövegnek a hossza ezt meghaladta, úgy az utána következő részt levágtuk. A kísérletek során alkalmazott további hiperparamétereink a következők voltak: tanulási ráta: 2e-5, lineáris tanulási ráta csökkentés, warmup ratio = 0,1.

5.2. Szavaztatásos megoldások

A tradicionális szavaztatásos megoldások, – ahol több egymástól függetlenül tanított modell együttese hozza meg a végső predikciót – széles körben ismert és használt megoldás, amellyel az egyes modellek eredményessége és robusztussága jellemzően javítható. Ez a megoldás természetesen a mesterséges intelligencia által generált szövegek felismerésére is alkalmas (Abburi és mtsai, 2023; Gu és Meng, 2024).

A megoldásainkban szavaztatásos megközelítést alkalmaztunk, amely során az azonos alapmodell különböző véletlenszerű inicializációval finomhangolt változatainak valószínűségi kimeneteit uniform módon átlagoltuk a végső predikció meghatározásához. Általában ez a megoldás javulást szokott hozni az eredmények tekintetében, azonban az erőforrásigénye drasztikusan megnő, mivel több modellt kell finomhangolni, és a kiértékelés költsége is az alkalmazott modellek számával arányosan nő.

6. Eredmények

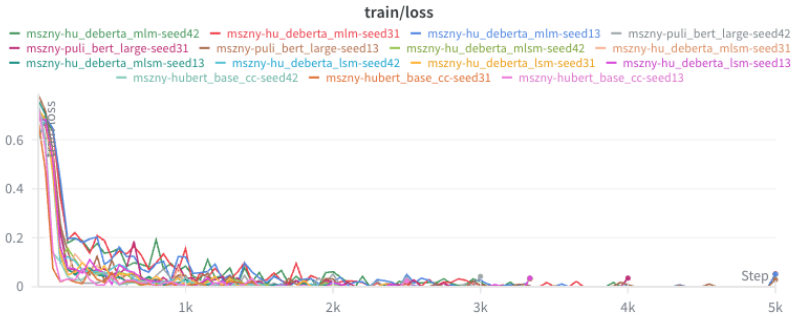
6.1. Publikus adathalmaz eredményei

A publikus adathalmazon történő finomhangolás eredményei azt mutatják az 1. táblázatban, hogy a modellek nagyon közeli eredményeket produkálnak. Ettől valamelyest lemaradnak a huDeBERTa-MLM modell finomhangolásával kapott eredmények.

A legjobb átlagos eredményt a PULI-BERT-Large modell érte el, ugyanakkor megjegyzendő, hogy ennek a modellnek a kapacitása a többi vizsgált modell kapacitásának többszöröse, így valós gyakorlatban való alkalmazása a többi finomhangolt modell használatához képest számottevő erőforrásigénnyel rendelkezik (egy 24 rétegből álló, 1024 dimenziós rejtett reprezentációkat alkalmazó modellről van szó, míg a többi vizsgált modell csupán 12 rétegből áll, és a rejtett vektoraik 768 dimenziósak).

1. táblázat. Publikus adathalmazon tanított modellek eredményeinek átlaga

Modellek	Pontosság	F1-pontszám	FPR	FNR
huBERT-base	$0,786 \pm 0,009$	$0,754 \pm 0,013$	$0,082 \pm 0,008$	$0,345 \pm 0,017$
PULI-BERT-Large	$\mathbf{0,794 \pm 0,002}$	$\mathbf{0,770 \pm 0,005}$	$0,100 \pm 0,017$	$\mathbf{0,311 \pm 0,016}$
huDeBERTa-MLM	$0,759 \pm 0,005$	$0,700 \pm 0,011$	$\mathbf{0,046 \pm 0,013}$	$0,436 \pm 0,021$
huDeBERTa-MLSM	$0,789 \pm 0,011$	$0,759 \pm 0,016$	$0,085 \pm 0,002$	$0,337 \pm 0,023$
huDeBERTa-LSM	$0,782 \pm 0,005$	$0,746 \pm 0,013$	$0,074 \pm 0,016$	$0,361 \pm 0,027$



5. ábra: Publikus adathalmazon tanított modellek tanítási veszteségének ábrázolása

Az 5. ábrán jól megfigyelhető, hogy a tanulási folyamat az elején meredeken, gyorsan konvergál, és a veszteség már a korai fázisokban is alacsony. Ez egy

általános probléma ebben a témakörben, amely azért gond, mivel a modell nem tanul lényeges új mintázatokat.

A szavaztatásos módszerek ebben az esetben nem hoztak érdemi javulást. Ez arra utalhat, hogy a publikus korpusz nem tükrözi a valóságban fellelhető magyar nyelvű szövegek heterogenitását, és nem tartalmaz elég nehezen osztályozható példát. A szavaztatásos módszerek eredménye a publikus adathalmazra a 2. táblázatban található. A fals pozitív rátán (FPR oszlop) a legtöbb esetben segített a szavaztatás, amely egy kritikus érték ebben az osztályozási feladatban.

2. táblázat. Publikus adathalmaz modelljei szavaztatva

Modellek	Pontosság	F1-pontszám	FPR	FNR
huBERT-base szavaztatás	$0,786$	$0,752$	$0,078$	$0,350$
PULI-BERT-Large szavaztatás	$\mathbf{0,793}$	$\mathbf{0,767}$	$0,094$	$\mathbf{0,320}$
huDeBERTa-MLM szavaztatás	$0,760$	$0,703$	$\mathbf{0,050}$	$0,431$
huDeBERTa-MLSM szavaztatás	$0,792$	$0,762$	$0,082$	$0,334$
huDeBERTa-LSM szavaztatás	$0,780$	$0,741$	$0,072$	$0,369$

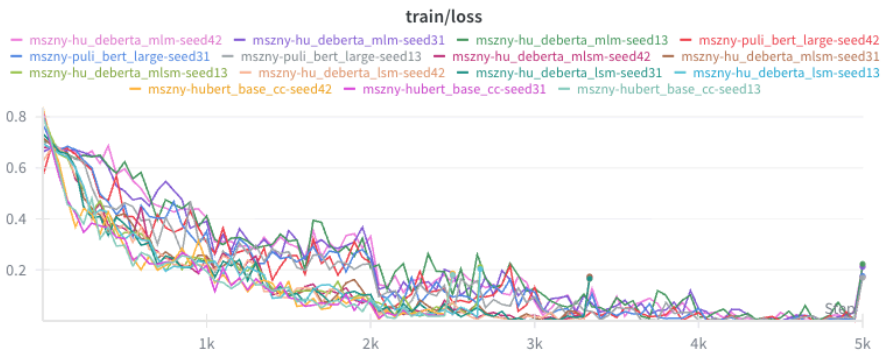
6.2. Kiegészített adathalmaz eredményei

A kiegészített adathalmazon kapott eredmények jelentősen felülmúlják a publikus adatok használatával mért értékeket. A legjobb eredményt önálló modellként a huDeBERTa-MLSM érte el 0,905 F1-pontszámmal, a további eredmények a 3. táblázatban láthatóak.

3. táblázat. Kiegészített adathalmazon tanított modellek eredményeinek átlaga

Modellek	Pontosság	F1-pontszám	FPR	FNR
huBERT-base	$0,885 \pm 0,021$	$0,891 \pm 0,015$	$0,165 \pm 0,071$	$0,065 \pm 0,030$
PULI-BERT-Large	$0,899 \pm 0,006$	$0,902 \pm 0,003$	$\mathbf{0,138 \pm 0,035}$	$0,065 \pm 0,022$
huDeBERTa-MLM	$0,827 \pm 0,015$	$0,841 \pm 0,012$	$0,257 \pm 0,033$	$0,089 \pm 0,013$
huDeBERTa-MLSM	$\mathbf{0,901 \pm 0,005}$	$\mathbf{0,905 \pm 0,003}$	$0,139 \pm 0,033$	$\mathbf{0,059 \pm 0,023}$
huDeBERTa-LSM	$0,883 \pm 0,005$	$0,888 \pm 0,008$	$0,164 \pm 0,023$	$0,070 \pm 0,032$

A veszteségfüggvények viselkedése a kiegészített tanítóhalmazon történ finomhangolás esetén jelentősen eltér a korábban látottaktól, ahogy azt az 5 és a 6. ábrák összevetése is jól mutatja. Míg a korábbi esetben látott veszteségfüggvény-értékek nagyon hamar 0 környékivé váltak, addig ebben az esetben a veszteségfüggvény csak később kezd el csökkenni, ami azt jelzi számunkra, hogy a modell sokkal komplexebb példákat kapott a kiegészített adatokon történő tanítás során.



6. ábra: Kiegészített adathalmazon tanított modellek tanítási veszteségének ábrázolása

Míg a publikus adathalmazon a szavaztatás hatása nem volt jelentős, a kiegészített adathalmazon történő finomhangolás során minden modell esetében javulást eredményezett az F1-pontszám tekintetében, ahogy azt a 4. táblázatban láthatjuk. A legjobb teljesítményt nyújtó modellnek a szavaztatást követően is a huDeBERTa-MLSM modell bizonyult, csakúgy mint az önálló eredményeknél. Az is látható, hogy a szavaztatás minden esetben képes volt a fals pozitív ráta csökkentésére. A kapott eredmények alapján a kiegészített adathalmaz sokkal inkább tekinthető valós felhasználásra alkalmasnak, mivel a magyar nyelvű szövegek jellemzőit jobban reprezentálja, amelyet a modell is jobban meg tud tanulni.

4. táblázat. Kiegészített adathalmaz modelljei szavaztatva

Szavaztatott modellek	Pontosság	F1-pontszám	FPR	FNR
huBERT-base	0,892	0,898	0,164	0,052
PULI-BERT-Large	0,909	0,912	**0,128**	0,054
huDeBERTa-MLM	0,834	0,846	0,247	0,085
huDeBERTa-MLSM	**0,912**	**0,916**	0,130	**0,046**
huDeBERTa-LSM	0,895	0,900	0,153	0,057

6.3. Külső (kereskedelmi) detektor eredményei

A saját finomhangolt megoldásainkat összevetettük több piacon elérhető detektorral[2]. A tesztelt kereskedelmi detektorok mindegyike támogatja a magyar

[2] `https://preds.hu`, `https://pangram.com`, `https://zerogpt.com`

nyelvet. API hívásokon keresztül küldtünk ki kéréseket, a korábban bemutatott teszthalmazra. Az 5. táblázatban láthatóak az eredmények, amelyről elmondható, hogy a kereskedelmi detektorok közül egyedül a Preds modellje ért el az általunk bemutatottaknál jobb eredményt, és ennek a modellnek a legalacsonyabb a fals pozitív rátája is.

5. táblázat. Modellek összehasonlítása

Modellek	Pontosság	F1-pontszám	FPR	FNR
Legjobb publikus modell	0,793	0,767	0,094	0,320
Legjobb kiegészített modell	0,912	0,916	0,130	**0,046**
Preds	**0,954**	**0,952**	**0,007**	0,085
PangramLabs	0,789	0,744	0,034	0,388
ZeroGPT	0,554	0,509	0,352	0,539

7. Összegzés

A célunk az volt, hogy tudományos igényességgel elsőként vizsgáljuk az MI-generált szövegek felismerésének lehetőségeit. Több adathalmazzal kísérleteztünk, az első adathalmaz egy publikus, már kész adathalmazokból összeszedett korpusz, a második pedig egy általunk gyűjtött adathalmaz, amelyben diverz forrásokból szerzett szövegek vannak.

A publikus korpuszon a modellek gyorsan, már az első epochokban 0 közeli veszteséget mutatnak, ami arra utal, hogy a feladat túl könnyű volt. A legjobb publikus eredmény szavaztatás nélkül 0,770 F1-pontszámot ért el, szavaztatással pedig 0,767 F1-pontszám volt, ahol a szavaztatás általában nem hozott javulás.

Ezzel szemben a kiegészített adathalmaz heterogénebb adathalmaz lett, ahol lényegesen jobb eredmények születtek. A legjobb önálló modell 0,905 F1-pontszámot (huDeBERTa-MLSM), míg a szavaztatásos megoldás 0,916 F1-pontszámot ért el. Ez azzal állhat összefüggésben, hogy a kiegészített adathalmaznál a modellek predikciói diverzebbek voltak, amely pedig az adathalmaz nagyobb varianciájából adódhatott.

A megoldásainkat összevetettük több piacon elérhető detektorral is, ahol a Preds modellje érte el a legjobb teljesítményt a pontosság és az F1-pontszám tekintetében. A másik két kereskedelmi detektor gyengébb eredményeket mutatott, mint az általunk létrehozott publikus modellek. A Preds modellje kiemelkedően alacsony fals pozitív rátát produkált.

Ennek egyik lehetséges oka, hogy a kereskedelmi megoldások várhatóan jóval nagyobb és diverzebb tanítóadaton voltak betanítva, illetve a problémára optimalizált hiperparaméterekkel és megoldásokkal finomhangolták a modelleket. Eredményeink ugyanakkor azt mutatják, hogy már a publikus, kisebb adathalmazokon finomhangolt modellekkel is elérhető némely kereskedelmi detektor pontossága.

Köszönetnyilvánítás

A cikk a Bolyai János Kutatási Ösztöndíj támogatásával készült. A kutatás az Európai Unió támogatásával valósult meg, az RRF-2.3.1-21-2022-00004 azonosítójú, Mesterséges Intelligencia Nemzeti Laboratórium projekt keretében. A MILAB SZTE nyelvtechnológia projekt nevében köszönetet mondunk az ELKH Cloud (lásd: Héder és mtsai (2022); `https://science-cloud.hu/` használatáért, ami hozzájárult a publikált eredmények eléréséhez.

Hivatkozások

Abburi, H., Roy, K., Suesserman, M., Pudota, N., Veeramani, B., Bowen, E., Bhattacharya, S.: A simple yet efficient ensemble approach for AI-generated text detection (2023), `https://arxiv.org/abs/2311.03084`

Bao, G., Zhao, Y., Teng, Z., Yang, L., Zhang, Y.: Fast-detectGPT: Efficient zero-shot detection of machine-generated text via conditional probability curvature. In: The Twelfth International Conference on Learning Representations (2024), `https://openreview.net/forum?id=Bpcgcr8E8Z`

Berend, G.: Neurális nyelvi modellek látens szemantikus információ alapján történő maszkolásmentes előtanítása. In: XX. Magyar Számítógépes Nyelvészeti Konferencia. pp. 85–95. Szegedi Tudományegyetem, Informatikai Intézet, Szeged (2024), `https://acta.bibl.u-szeged.hu/78418/`

boapps: alpaca-hu-v2: Hungarian instruction-follower dataset. `https://huggingface.co/datasets/boapps/alpaca-hu-v2` (2024), accessed: 2025-11-22

Chamezopoulos, S., Herrmannova, D., De Waard, A., Herrmannova, D., Rosati, D., Kashnitsky, Y.: Overview of the DagPap24 shared task on detecting automatically generated scientific paper. In: Ghosal, T., Singh, A., Waard, A., Mayr, P., Naik, A., Weller, O., Lee, Y., Shen, S., Qin, Y. (szerk.) Proceedings of the Fourth Workshop on Scholarly Document Processing (SDP 2024). pp. 7–11. Association for Computational Linguistics, Bangkok, Thailand (Aug 2024), `https://aclanthology.org/2024.sdp-1.2`

Gu, R., Meng, X.: AISPACE at SemEval-2024 task 8: A class-balanced soft-voting system for detecting multi-generator machine-generated text. In: Ojha, A.K., Doğruöz, A.S., Tayyar Madabushi, H., Da San Martino, G., Rosenthal, S., Rosá, A. (szerk.) Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024). pp. 1476–1481. Association for Computational Linguistics, Mexico City, Mexico (Jun 2024), `https://aclanthology.org/2024.semeval-1.212`

Héder, M., Rigó, E., Medgyesi, D., Lovas, R., Tenczer, S., Török, F., Farkas, A., Emődi, M., Kadlecsik, J., Mező, Gy., Pintér, Á., Kacsuk, P.: The past, present and future of the ELKH cloud. Információs Társadalom 22(2), 128 (aug 2022), `https://doi.org/10.22503/inftars.xxii.2022.2.8`

Huang, G., Li, Y., Pleiss, G., Liu, Z., Hopcroft, J.E., Weinberger, K.Q.: Snapshot ensembles: Train 1, get m for free. In: International Conference on Learning Representations (2017), `https://openreview.net/forum?id=BJYwwY9ll`

Kiss, M., Berend, G.: SzegedAI at GenAI detection task 1: Beyond binary - soft-voting multi-class classification for binary machine-generated text detection across diverse language models. In: Alam, F., Nakov, P., Habash, N., Gurevych, I., Chowdhury, S., Shelmanov, A., Wang, Y., Artemova, E., Kutlu, M., Mikros, G. (szerk.) Proceedings of the 1stWorkshop on GenAI Content Detection (GenAIDetect). pp. 166–172. International Conference on Computational Linguistics, Abu Dhabi, UAE (Jan 2025), https://aclanthology.org/2025.genaidetect-1.15/

Liang, W., Izzo, Z., Zhang, Y., Lepp, H., Cao, H., Zhao, X., Chen, L., Ye, H., Liu, S., Huang, Z., Mcfarland, D., Zou, J.Y.: Monitoring AI-modified content at scale: A case study on the impact of ChatGPT on AI conference peer reviews. In: Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., Berkenkamp, F. (szerk.) Proceedings of the 41st International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 235, pp. 29575–29620. PMLR (21–27 Jul 2024a), https://proceedings.mlr.press/v235/liang24b.html

Liang, W., Zhang, Y., Wu, Z., Lepp, H., Ji, W., Zhao, X., Cao, H., Liu, S., He, S., Huang, Z., Yang, D., Potts, C., Manning, C.D., Zou, J.Y.: Mapping the increasing use of LLMs in scientific papers. In: First Conference on Language Modeling (2024b), https://openreview.net/forum?id=YX7QnhxESU

Marchitan, T.g., Creanga, C., Dinu, L.P.: Team Unibuc - NLP at SemEval-2024 task 8: Transformer and hybrid deep learning based models for machine-generated text detection. In: Ojha, A.K., Doğruöz, A.S., Tayyar Madabushi, H., Da San Martino, G., Rosenthal, S., Rosá, A. (szerk.) Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024). pp. 403–411. Association for Computational Linguistics, Mexico City, Mexico (Jun 2024), https://aclanthology.org/2024.semeval-1.63/

Mitchell, E., Lee, Y., Khazatsky, A., Manning, C.D., Finn, C.: DetectGPT: zero-shot machine-generated text detection using probability curvature. In: Proceedings of the 40th International Conference on Machine Learning. ICML'23, JMLR.org (2023)

Nemeskey, D.M.: Natural Language Processing Methods for Language Modeling. Ph.D.-értekezés, Eötvös Loránd University (2020)

Nemeskey, D.M.: Introducing huBERT. In: XVII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2021). pp. 3–14. Szeged (2021)

Niklaus, J.: EU Wikipedias Dataset. https://huggingface.co/datasets/joelniklaus/EU_Wikipedias (2023), accessed: 2025-11-22

Sarvazyan, A.M., González, J.Á., Franco-salvador, M.: Genaios at SemEval-2024 task 8: Detecting machine-generated text by mixing language model probabilistic features. In: Ojha, A.K., Doğruöz, A.S., Tayyar Madabushi, H., Da San Martino, G., Rosenthal, S., Rosá, A. (szerk.) Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024). pp. 101–107. Association for Computational Linguistics, Mexico City, Mexico (Jun 2024), https://aclanthology.org/2024.semeval-1.17

Sarvazyan, A.M., González, J.Á., Franco-Salvador, M., Rangel, F., Chulvi, B., Rosso, P.: Overview of AuTexTification at IberLEF 2023: Detection and att-

ribution of machine-generated text in multiple domains. Proces. del Leng. Natural 71, 275–288 (2023), http://journal.sepln.org/sepln/ojs/ojs/index.php/pln/article/view/6559

Wang, Y., Mansurov, J., Ivanov, P., Su, J., Shelmanov, A., Tsvigun, A., Mohammed Afzal, O., Mahmoud, T., Puccetti, G., Arnold, T.: SemEval-2024 task 8: Multidomain, multimodel and multilingual machine-generated text detection. In: Ojha, A.K., Doğruöz, A.S., Tayyar Madabushi, H., Da San Martino, G., Rosenthal, S., Rosá, A. (szerk.) Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024). pp. 2057–2079. Association for Computational Linguistics, Mexico City, Mexico (Jun 2024), https://aclanthology.org/2024.semeval-1.279

Wang, Y., Shelmanov, A., Mansurov, J., Tsvigun, A., Mikhailov, V., Xing, R., Xie, Z., Geng, J., Puccetti, G., Artemova, E., Su, J., Ta, M.N., Abassy, M., Elozeiri, K., Ahmed, S.E.D., Goloburda, M., Mahmoud, T., Tomar, R.V., Aziz, A., Laiyk, N., Afzal, O.M., Koike, R., Kaneko, M., Aji, A.F., Habash, N., Gurevych, I., Nakov, P.: GenAI content detection task 1: English and multilingual machine-generated text detection: AI vs. human. In: Proceedings of the 1st Workshop on GenAI Content Detection (GenAIDetect). International Conference on Computational Linguistics, Abu Dhabi, UAE (January 2025)

Yang, Z.Gy., Dodé, R., Ferenczi, G., Héja, E., Jelencsik-Mátyus, K., Kőrös, A., Laki, L.J., Ligeti-Nagy, N., Vadász, N., Váradi, T.: Jönnek a nagyok! BERT-Large, GPT-2 és GPT-3 nyelvmodellek magyar nyelvre. In: XIX. Magyar Számítógépes Nyelvészeti Konferencia. pp. 247–262. Szegedi Tudományegyetem, Informatikai Intézet, Szeged (2023), https://acta.bibl.u-szeged.hu/78417/

Racka: Efficient Hungarian LLM Adaptation on Academic Infrastructure

Zsolt Csibi[2⋆], Bence György Gortka[1], Natabara Gyöngyössy[2], Kornél Nagy[1], Dávid Márk Nemeskey[1], Gábor Palkó[1], Martin Sallai[1], András Simonyi[2], András Márk Szekeres[1]

[1]ELTE Faculty of Humanities, Department of Digital Humanities
{gortka.bence, nagy.kornel, nemeskey.david, palko.gabor, sallai.martin, szekeres.andras.mark}@btk.elte.hu
[2]ELTE Faculty of Informatics, Department of Artificial Intelligence
{vrnf2j, natabara, simonyi}@inf.elte.hu

Abstract. We present Racka-4B, a lightweight, continually pretrained large language model designed to bridge the resource gap between Hungarian and high-resource languages such as English and German. Racka employs parameter-efficient continual pretraining via Low-Rank Adaptation (LoRA) on a Qwen3-4B backbone, making the recipe practical on A100 (40GB)-based HPC clusters with low inter-node bandwidth. To better match the training distribution, we replace and adapt the tokenizer. By swapping 32k tokens, we achieve 47% lower tokenization fertility and generation latency for Hungarian while maintaining competitive performance in English and German. The model is trained on 160B subword tokens (85B whitespace words) drawn from a mixture of internet and high-quality curated sources, with a composition of 44% Hungarian, 24% English, 21% German, and 11% code to mitigate catastrophic forgetting. Our evaluations indicate modest but stable results in language adaptation with similar LM-Eval-Harness results but increased overall performance on HULU and OpenHuEval compared to the original Qwen and the latest Puli models. The results also showcase that Racka-4B is capable of Hungarian chat with English reasoning even in the absence of explicit Hungarian post-training on these tasks.
Keywords: Large Language Model, Continual Pretraining, Language adaptation, Tokenizer Adaptation

1 Introduction

1.1 The Challenge of Linguistic Digital Sovereignty for Medium-Resource Languages

The proliferation of Large Language Models (LLMs) has marked a transformative era in artificial intelligence, yet this progress has been unevenly distributed across the world's languages (Ali and Pyysalo, 2024). The current landscape is

⋆ Authors in alphabetical order.

dominated by models pretrained on vast corpora composed predominantly of English and a few other high-resource languages, creating a significant performance and resource disparity for less-resourced linguistic communities (Zhong et al., 2025). For medium-resource languages such as Hungarian, a Finno-Ugric language characterized by its agglutinative nature and rich morphology, this gap is particularly pronounced. Off-the-shelf multilingual models often exhibit suboptimal performance due to insufficient representation in training data and tokenizers that are ill-suited to language-specific morphology. This is particularly the case for open-source models, which visibly struggle with Hungarian grammar.

This disparity highlights a critical challenge to linguistic digital sovereignty. Linguistic communities need the capacity to develop, deploy, and benefit from AI technologies that are not only proficient in their language but are also attuned to their unique cultural and contextual nuances. Achieving this requires the creation of models trained on nationally relevant data and, crucially, developed using methodologies that are accessible to local research communities. Likely, most research groups will not have access to the hyperscale computing infrastructure available to large industrial labs and will have to rely on shared, and often underpowered, HPC infrastructure.

1.2 Core Contributions

In response to these challenges, we introduce Racka, a 4-billion-parameter language model for Hungarian adapted trough a pragmatic and resource-efficient approach from a Qwen3-4B reasoning model. The model's name pays homage to the *Hortobágy Racka*, an indigenous Hungarian sheep breed and llama alternative. Racka was trained using publicly accessible infrastructure, Hungary's Komondor supercomputer, rather than state-of-the-art proprietary hardware. Our primary contributions are as follows:

1. **Curation of a Large-Scale Multilingual Corpus:** We describe the assembly and curation of a 160 billion subword token dataset representing state-of-the-art size in training a Hungarian-centric language model. We justify its strategic composition (44% Hungarian, 43% high-resource languages, and 11% code) as a pragmatic approach to mitigate catastrophic forgetting during the continual pretraining process. The inclusion of English and German is justified by the census report of the Hungarian Central Statistical Office[1], as these are the most common foreign languages in Hungary.
2. **Optimized Tokenization for Hungarian:** We underscore the critical role of tokenizer adaptation for morphologically rich. We document the design, training, and evaluation of a new tokenizer that dramatically improves tokenization efficiency (i.e., subword fertility) for Hungarian, while carefully preserving the performance of the base model on high-resource control languages like English and German.

[1] `https://nepszamlalas2022.ksh.hu/eredmenyek/vegleges-adatok/kiadvany/`

3. **A Resource-Efficient Training Recipe:** We detail a practical methodology for the continual pretraining of a 4B parameter model on an HPC cluster featuring NVIDIA A100 (40GB) GPUs. We provide justification for our selection of Distributed Data Parallel (DDP) over the more recent Fully Sharded Data Parallel (FSDP) paradigm, demonstrating that this choice is a deliberate optimization for our specific hardware environment.
4. **The first Hungarian reasoning LLM:** We present Racka, which to our knowledge is the first Hungarian LLM with reasoning output. The model is openly available on Hugging Face as `elte-nlp/Racka-4B`[2].

2 Related Work

2.1 Paradigms of Language Model Adaptation

Adapting existing foundation models to new languages, domains, or tasks is a central challenge in modern NLP. The prohibitively high cost of training LLMs from scratch has spurred research into more efficient adaptation strategies (Csaki et al., 2023; Khade et al., 2025). As an industry standard practice for language adaptation, continual pretraining (CPT) is widely adopted. This method focuses on further self-supervised training on large, unlabeled corpora to expand a model's foundational knowledge (Vanroy, 2024; Schuhmann et al., 2024; Ke et al., 2023).

The work on Racka falls squarely within the CPT paradigm, specifically targeting what the literature defines as "Language Expansion" and "Domain Adaptation". This framing positions our effort not as the creation of a final, task-specific model, but as general fine-tuning of a foundational model for Hungarian, which can subsequently be adapted for downstream tasks. A primary obstacle in any continual learning scenario is *catastrophic forgetting*, where a model's performance on previously learned tasks or languages degrades significantly as it learns new information (Ke et al., 2023). A simple yet effective strategy to mitigate this is data replay, which involves mixing data from the original training distribution with the new data. Our multilingual data composition, which retains a significant portion of English and German text, is a direct application of this principle. Other alternatives include careful train scheduling, intense regularizations, or the analysis of task-vector shifts. We refrain from these methods in favor of data replay, as it directly controls our training objective and does not require extensive experimentation (Li and Lee, 2024). Previous Hungarian language adaptation work also applied this approach successfully (Szentmihályi et al., 2024; Csaki et al., 2024; Yang et al., 2025d).

2.2 Parameter-Efficient Methods for Continual Pretraining

The computational burden of updating the billions of parameters in an LLM has led to the development of Parameter-Efficient Fine-Tuning (PEFT) methods. These techniques aim to adapt models by training only a small fraction

[2] `https://huggingface.co/elte-nlp/Racka-4B`

of their total parameters, drastically reducing memory and compute requirements. Among the most widely adopted PEFT methods is Low-Rank Adaptation (LoRA; Hu et al., 2021). LoRA operates on the hypothesis that the change in model weights during adaptation has a low intrinsic rank. This aligns well with our aim to train on limited-memory GPUs with low throughput internode connections. By keeping the base model frozen, it is shown to inherently aid in overcoming catastrophic forgetting of the base model's knowledge thus making it ideal for CPT (Li and Lee, 2024).

2.3 The Landscape of Hungarian Language Models

The development of Racka builds upon a growing body of work dedicated to creating capable language models for Hungarian. These efforts span both academic and industrial research and provide essential context for our contributions.

- **NYTK Models:** The Hungarian Research Centre for Linguistics (NYTK) has been a key contributor, developing a series of models within the "Puli" family. These include models in the range of 7-8B parameters, both monolingual and continually pretrained. Latest models include instruction-tuned and multilingual (English, Chinese) models (Yang et al., 2023, 2025d,c).
- **OTP Bank Models:** A major government-supported initiative, lead by OTP Bank, demonstrated the feasibility of developing high-performance models for Hungarian at a large scale (Szentmihályi et al., 2024). The resulting models were trained on an extensive dataset on custom SambaNova hardware. Unfortunately, these models are not available to the public.
- **International Initiatives:** The ecosystem also includes models adapted from multilingual open-source foundations. SambaLingo-Hungarian-Base, for instance, adapted a Llama-2 7B model by training on 59 billion tokens from the Hungarian portion of the Cultura-X dataset (Csaki et al., 2024). Similarly, projects like OpenEuroLLM have fine-tuned models like Gemma for improved Hungarian performance. Other European initiatives such as EuroLLM and Salamandra include Hungarian in their supported languages. (OpenEuroLLM, 2025; Martins et al., 2025; Gonzalez-Agirre et al., 2025)

This existing landscape highlights a clear and valuable niche for the Racka project. While previous efforts have focused on larger models (OTP-13B), monolingual training (PULI-GPT-3SX), multilingual training (PULI-GPTrio) or adaptation on smaller datasets (SambaLingo), Racka's contribution is its unique combination of a lightweight (4B) backbone, continual pretraining on an exceptionally large multilingual corpus (160B tokens), and a resource-efficient methodology explicitly designed for and validated on publicly accessible national HPC infrastructure. One key insight that can be accrued from the papers above is that for smaller languages, continual pretraining of already capable models results in better performing LLMs than training new models from scratch.

3 Training Data Curation

3.1 Corpus Design and Composition

The foundation of any successful language model is the quality and scale of its training data. For Racka, we curated a comprehensive multilingual corpus totaling approximately 160 billion subword tokens with a dataset size of 639 GB. The composition of this corpus was a strategic design choice aimed at achieving robust language adaptation for Hungarian while simultaneously mitigating the effects of catastrophic forgetting on the backbone model's existing capabilities. The final data mixture, detailed in Table 1, allocates the majority of the data to our target language, Hungarian, while retaining substantial portions of high-quality English, German, and source code data. The code component was included to maintain the model's strong reasoning and structured-text processing abilities, which are known to be beneficial for a wide range of downstream analytical tasks (Ma et al., 2024).

Table 1. Composition of the Racka Pretraining Corpus after subset weighting and tokenization of the Hungarian-adapted tokenizer.

Language	**BPE Tokens (Billions)**	**Ratio**	**Documents (Millions)**
Hungarian	∼70	44%	70.56
English	∼38	24%	26.12
German	∼34	21%	24.06
Code	∼18	11%	10.17
Total	**∼160**	**100%**	**130.91**

3.2 Data Sources

The corpus was assembled from a combination of large-scale web crawls and high-quality curated datasets to ensure both breadth and quality. The primary source for web data across all languages was the Common Crawl repository, with all data sourced from crawls conducted prior to our cutoff date of October 2024 (Common Crawl, 2025).

For English, we sample approximately 38B tokens sourced from non-web text-based subsets of The Pile (Gao et al., 2020), and the heavily filtered FineWeb corpus (Penedo et al., 2024). We sample approximately 34B tokens of German text from the similarly pre-filtered occiglot-fineweb (Occiglot Project, 2024) and 18B tokens of code from The Stackv2 (Lozhkov et al., 2024), slightly oversampling structured text formats and markups to better represent them compared to code files. 70B Hungarian tokens were collected from various sources including Common Crawl, academic repositories, court rulings, movie subtitles, Wikipedia

and digital news articles. Following Nemeskey (2020), the web and news subcorpora were heavily filtered and deduplicated on both the URL and document-level. Paragraphs that appeared frequently across the respective subcorpora were also discarded. The aggregated per-language corpus sizes are reported in Table 1; the composition of the Hungarian corpus is detailed in Appendix B. Note that the fertility of our adapted tokenizer is smaller for Hungarian, thus we produce less tokens in Hungarian compared to the other subsets.

In case of Hungarian web data, we apply standard boilerplate filtering and a strict paragraph-level 13-gram (whitespace token) deduplication resulting in our final dataset size. To ensure that the prevalence of scarce but high quality data is higher in the training dataset, we oversample the news subset (weight 2.0), the Wikipedia subset (weight 3.0), Hungarian subtitles (weight 2.0) and academic repositories (weight 1.5). We drop those documents that are less than 500 characters long but do not apply other length-based filtering.

For measuring training efficiency we hold out 10-10 thousand element validation and test sets of independent documents. Both sets are sampled in a language-stratified manner.

4 Methodology

4.1 Backbone Model Architecture

The foundation for our work is the Qwen3-4B model, an open-source transformer-based LLM developed by Alibaba Cloud accessed through Hugging Face[3]. We selected this model as our backbone due to its strong baseline performance, favorable open-source license (Apache 2.0), and an architecture well-suited for our objectives. The Qwen3-4B model features 4 billion total parameters (3.6 billion non-embedding) distributed across 36 transformer layers (Yang et al., 2025a).

Key architectural specifications include the use of Grouped Query Attention (GQA) for efficient inference, the SwiGLU activation function for improved performance, and RMSNorm for stable training (Yang et al., 2025a). The model supports a native context length of 32,768 tokens, making it capable of processing long documents without truncation, which could be extended via interpolation of the positional encoding (Su et al., 2024). We specifically chose the reasoning and instruction-tuned variant of the model, hoping that its inherent capabilities would provide a stronger starting point and ultimately benefit in downstream performance after our continual pretraining phase. To our knowledge, this makes Racka the first reasoning model to be adapted to Hungarian.

4.2 Tokenizer Adaptation for Hungarian

A key hypothesis of our work is that the default tokenizer of a multilingual model is suboptimal for a morphologically rich, agglutinative language like Hungarian.

[3] `https://huggingface.co/Qwen/Qwen3-4B`

Such tokenizers, trained on a distribution dominated by English, tend to fragment Hungarian words into numerous, less meaningful subword units, leading to longer sequence lengths, increased computational cost, and potentially poorer model performance (Csaki et al., 2023). Several previous models have also experimented with tokenizer adaptation. To address this issue, we developed a new, adapted tokenizer for Racka, slightly modifying previously used adaptation algorithms. The process involved several steps:

1. We first trained a new Byte-Pair Encoding (BPE) tokenizer from scratch, using only a small portion of our Hungarian training corpus. This produced a vocabulary optimized for the statistical properties of Hungarian.
2. We then merged this new Hungarian-specific vocabulary with the original vocabulary of the Qwen3 tokenizer. Rare, typically non-latin character-based multi-byte tokens were removed, and novel tokens were inserted resulting in an adapted vocabulary that retained all of the original latin alphabet tokens while adding 32 000 new, Hungarian-optimized tokens learned from the Hungarian-only tokenizer. See Appendix A for details.
3. The model's architecture was updated to accommodate this change: the token embedding matrix and the language-model head (the final output layer) were altered to match the new tokenizer. To provide a strong initialization for the newly added tokens in the embedding matrix, we employed the VIPI (Vocabulary Initialization with Partial Inheritance) vocabulary-transfer technique (Mosin et al., 2023). For each new token, we used the original tokenizer to break it down into its constituent subwords. The new token's initial embedding vector was then calculated as the average of the embedding vectors of these constituent subwords from the original, pretrained embedding matrix. This approach provides a semantically meaningful starting point for the new tokens, facilitating faster convergence during training.

4.3 Packed Continual Pretraining with LoRA

To enable parameter-efficient continual pretraining, we employed Low-Rank Adaptation (LoRA). LoRA modules were inserted into all linear layers of the Qwen3-4B backbone. The weights of the base model were frozen throughout the training. The embedding and language modeling head layers are tied as in Qwen3-4B.

Our choice of LoRA hyperparameters was guided by recent best practices and empirical validation. Specifically, we set the LoRA rank (r) to 64 and the scaling factor (α) to 128, following the heuristic of setting α to approximately twice the rank to ensure sufficient adaptation capacity. A dropout of 0.1 was applied to the LoRA adapters to improve generalization. With this setup we train 0.52B parameters accounting for 12.5% of the base model size. We used separate learning rates for LoRA and non-LoRA parameters: 1×10^{-4} for LoRA and 5×10^{-5} for non-LoRA parameters, both with a weight decay of 0.005. Optimization was performed using AdamW. The learning rate schedule included a linear warmup phase covering 1% of total training steps, followed by linear

decay to zero which is shown to be beneficial (Bergsma et al., 2025). We did not employ a specialized tokenizer healing phase.

To maximize training efficiency, we used input sequence packing (Ding et al., 2024), merging shorter sequences into a single context window of up to 4096 tokens. Both the beginning-of-sequence and end-of-sequence delimiters were set to `<|endoftext|>`, as defined by the Qwen tokenizer. Packing was performed using a linear scaling algorithm prior to training. For computational efficiency, we used a standard full causal attention mask (as in (DeepSeek-AI, 2025)), rather than a multi-segment lower-triangular mask (Hugging Face, 2024), which would otherwise require quadratic memory to store the mask itself, as well as the inability to use high-speed attention kernels.

4.4 Training Infrastructure and Parallelization Strategy

Training was implemented in Python 3.12 using PyTorch 2.7.1 and CUDA 12.6, with the SDPA backend for efficient attention computation on NVIDIA A100 GPUs. All experiments were conducted using the HuggingFace Transformers library version 4.52.3, Accelerate 1.7.0, and PEFT 0.15.2.

Training was performed on the Komondor HPC cluster, Hungary's national supercomputing facility at the University of Debrecen (Digital Government Development and Project Management Ltd. (DKF), 2025). We utilized the GPU-accelerated partition, comprising nodes equipped with 4×NVIDIA A100 GPUs (40 GB) and 64-core AMD EPYC 7763 CPUs, interconnected via HPE Slingshot.

Given the 4B parameter size of our model, which fits within the 40 GB memory of a single A100 GPU using gradient checkpointing, we empirically compared Fully Sharded Data Parallel (FSDP) and Distributed Data Parallel (DDP) strategies. While FSDP reduces per-GPU memory usage by sharding model states, it incurs frequent, latency-sensitive `all-gather` operations, which proved to be a bottleneck on Komondor's interconnect. In contrast, DDP replicates the model on each GPU and synchronizes gradients via a single, bulk `all-reduce` after each backward pass, resulting in less frequent communication.

We conducted performance measurements on a distributed setup consisting of two nodes, each equipped with four GPUs. In DDP configuration, the training throughput reached 2.9–3.0 seconds per iteration. In contrast, FSDP configuration delivered substantially lower performance, reaching 5.0–5.1 seconds per iteration. Mixed-precision training with the standard FSDP implementation is not possible, training with full precision resulted in OOM for our minimal training context length of 4k tokens. As DDP yielded superior scaling and GPU utilization, we adopted it for all experiments as a hardware-aware choice for parallelization on the available infrastructure.

Training was performed on 64 A100 GPUs (16 nodes) with a per-device batch size of 2 throughout 287 hours resulting in a total GPU time of 2.1 years. During training we experienced 1 unexpected (hardware failure) and 2 scheduled stops which resulted in no significant loss of compute with checkpoints in place. The estimated carbon emission for the training is 1290 kg CO_2 eq. which is a below average emission due to the environmentally friendly design of the HPC utilized.

Table 2. Tokenizer performance on the validation set: Subword fertility (lower is better) and relative change from Qwen3-4B.

Language	Qwen3-4B	Racka-4B	Change (%)
Hungarian	3.1269	1.6584	-46.96
English	1.5705	1.9387	23.44
German	2.0502	2.3090	12.62

4.5 Evaluation Protocol

We assess the model from two perspectives: intrinsic language modeling capabilities and downstream Hungarian benchmark performance.

Intrinsic Evaluation: We use perplexity (PPL) as our primary intrinsic metric. Perplexity measures the model's uncertainty in predicting the next token in a sequence; a lower perplexity score indicates a better fit to the data distribution (Jelinek et al., 2005). We calculated perplexity on the held-out validation set containing a mix of Hungarian, English, German and code to monitor learning progress and assess for catastrophic forgetting.

LM-Eval-Harness benchmark: To measure the model's practical utility without task-specific training, we conducted few-shot evaluations on a selection of Hungarian language understanding tasks (Gao et al., 2024). For this we utilized the LM-Eval-Harness with Hungarian translated benchmarks. We used the already implemented benchmarks ARC_HU, MMLU_HU, Hellaswag_HU and TruthfulQA_HU from LM-Eval-Harness, which is the work of (Lai et al., 2023). Additionally, we manually added the Hungarian translated version of GSM8K (Cobbe et al., 2021), which is a mathematical reasoning dataset and was transletd by (Thellmann et al., 2024), to check the model reasoning capabilities on mathemical problems. We ran all benchmarks separately, using the default config parameters with and without chat templates as defined in the original LM-Eval-Harness repository[4]. Table 3 shows our results.

HULU Benchmark The HULU benchmark offers a comprehensive set of supervised tasks covering a wide range of linguistic tasks, including acceptability, causal reasoning, sentiment classification, pronoun resolution, factual commitment, and reading comprehension. For the evaluation, we separately fine-tuned our model on each HULU subtask using a custom fork of the official benchmarking script[5]. Both full and LoRA mode finetuning was performed with the following key parameters: $lora\_rank = 64$, $lora\_alpha = 128$, $lora\_dropout = 0.1$, and $batch\_size = 2$. We report the best mode as detailed in Appendix D.

OpenHuEval benchmark To further evaluate the ability of the model to perform Hungarian reading comprehension and generation tasks, we conducted fine-tuned downstream evaluations using the OpenHuEval benchmark suite (Yang et al., 2025b). This collection focuses on cloze-style question an-

[4] `https://github.com/EleutherAI/lm-evaluation-harness`
[5] `https://github.com/nytud/HuLU`

Table 3. Results on the HULU, OpenHuEval and LM-Eval-Harness benchmarks. HULU results are measured on the publicly available validation set as an average of multiple runs taking the better option from the LoRA and full finetuned models. In case of Qwen and Racka models we patch the original OpenHuEval implementation to be compatible with these models. For details see Appendix C. LM-Eval-Harness was evaluated with and without chat templates and the best overall result is kept (with chat template for Racka and without it for the other models). The best result per benchmark is highlighted with **bold**.

HULU benchmark												
Dataset	**Qwen3-4B**			**Racka-4B**			**Qwen3-4B-Base**			**PULI-LlumiX-Llama-3.1 8B**		
	ACC	MCC	F1	ACC	MCC	F1	ACC	MCC	F1	ACC	MCC	F1
HuCOLA	0.811	0.348	0.784	0.862	0.566	0.856	0.825	0.404	0.803	**0.899**	**0.692**	**0.897**
HuCOPA	0.559	0.118	0.558	0.799	0.600	0.799	0.585	0.171	0.584	**0.936**	**0.872**	**0.936**
HuSST	0.752	0.502	0.743	0.760	0.514	0.751	0.754	0.508	0.751	**0.780**	**0.560**	**0.770**
HuRTE	**0.908**	**0.814**	**0.908**	0.879	0.755	0.879	0.887	0.772	0.887	0.898	0.794	0.898
HuWNLI	0.503	−0.098	0.386	**0.567**	**0.103**	**0.455**	0.537	−0.060	0.407	0.38	−0.282	0.367
HuCommitmentBank	**0.738**	**0.608**	**0.732**	0.639	0.474	0.637	0.629	0.473	0.611	0.485	0.274	0.459
Overall Performance (HULU)	0.711	0.382	0.685	**0.751**	**0.502**	**0.730**	0.702	0.378	0.673	0.729	0.485	0.721
OpenHuEval benchmark												
HuWildBench (WBScore)	**63.03**			57.17			52.59			17.77		
HuSimpleQA (Acc)	7.30			10.05			5.91			**20.03**		
HuProverbRea (Acc OE)	62.47			61.94			41.15			**75.86**		
HuProverbRea (Acc 2CQ)	74.98			**77.53**			0			77.36		
HuMatchingFIB (B Acc)	39.59			38.93			**42.3**			33.54		
HuMatchingFIB (Q Acc)	**5.94**			4.68			5.58			3.96		
HuStandardFIB (B Acc)	13.20			18.98			0			**29.16**		
HuStandardFIB (Q Acc)	1.08			**2.15**			0			**2.15**		
Overall Performance (OpenHuEval)	33.44			**33.93**			18.44			32.47		
LM-Eval-Harness benchmark												
Arc_hu (Acc)	0.320			0.345			0.379			**0.386**		
Arc_hu (Acc_norm)	0.384			0.410			0.417			**0.432**		
Hellaswag_hu (Acc)	0.337			0.366			0.361			**0.424**		
Hellaswag_hu (Acc_norm)	0.410			0.451			0.456			**0.561**		
MMLU_hu (Acc)	0.543			0.538			**0.597**			0.531		
TruthfulQA_hu_mc1 (Acc)	0.318			**0.364**			0.328			0.304		
TruthfulQA_hu_mc2 (Acc)	0.510			**0.549**			0.505			0.488		
GSM8K_hu (Strict-match)	0.633			0.530			**0.640**			0.476		
GSM8K_hu (Flexible extract)	0.629			0.533			**0.642**			0.479		
Overall Performance (LM-Eval-Harness)	0.453			0.454			**0.481**			0.455		

swering and comprehension tasks designed to capture various aspects of contextual understanding, reasoning, and lexical inference in Hungarian. OpenHuEval comprises five datasets: HuWildBench (real Hungarian user queries), HuSimpleQA (fact-based Hungarian questions), HuProverbRea (reasoning with Hungarian proverbs), HuMatchingFIB (option-based fill-in-the-blank), and HuStandardFIB (free-form fill-in-the-blank). The evaluation was made using the original repository of the benchmark[6].

5 Results and Analysis

To measure the performance of our methodology, in this section we analyze both the tokenizer and the Hungarian performance of Racka-4B. Benchmarks include tasks related to base language use, natural language inference, logic and Hungarian-specific knowledge. We compare our model to the original Qwen3-4B model, the vanilla Qwen3-4B-Base language model and a state-of-the-art Hungarian model, Puli-LlumiX-Llama-3.1 8B. The OTP models are not openly available, benchmarking them is not currently possible. Extending the benchmarks with international models, which according to our experiences tend to perform lower than models trained in Hungary, is matter of future works.

[6] `https://github.com/opendatalab/OpenHuEval`

5.1 Tokenizer Performance

The primary goal of our tokenizer adaptation was to improve the encoding efficiency for Hungarian without degrading performance on the high-resource languages present in the base model. We evaluated this using *subword fertility*, which is the average number of tokens required to represent a single whitespace delimited word (Ács et al., 2021).

The results measured on the validation documents, presented in Table 2, confirm the success of our adaptation strategy. For Hungarian, the Racka tokenizer demonstrates a substantial improvement over the original Qwen3 tokenizer. Subword fertility was reduced by over 46%, from 3.13 down to 1.66. This means that, on average, the same Hungarian text can be represented with almost half of the original Qwen tokens, leading to direct, in most cases linear improvements in inference efficiency and latency. Crucially, this improvement in Hungarian was achieved with only modest trade-offs for other languages: fertility increased by less than 13% for German and 24% for English. This indicates that our vocabulary extension successfully preserved the efficiency for high-resource languages.

5.2 Continual Pretraining Results

Following the successful tokenizer adaptation, we successfully conducted the continual pretraining phase on the full 160B token dataset. The training ran for 326 357 steps on 64 GPUs in parallel, with a per-device batch size of 2 and gradient accumulation set to 4, resulting in an effective logical batch size of 512 sequences with a context length of 4096 for each model update. The training loss curve is depicted in Appendix E. We see that even though the training loss begins to flatten after 50 000 steps, the validation perplexity continues to steadily decrease to a final value of 6.99, indicating that overfitting is not reached.

The HuLU benchmark paints a positive picture (see the top section of Table 3) with a top overall performance. Racka seems to be behind the base model(s) on HuRTE and HuCommitmentBank, but outperforms them on the other four tasks. On HuWNLI and HuCommitmentBank, it even surpasses the current state-of-the-art 8B Puli model and achieves the best result out of all tested models on the former[7].

The middle section of Table 3 presents the results of evaluating Racka on OpenHuEval. We also denote that larger (7-8 B) models in the original OpenHuEval Yang et al. (2025b) paper have an overall score of 28.77 (LLama-3.1 8B) and 25.64 (Qwen2.5-instruct 7B). The results show that Racka delivers competitive performance relative to models twice its size and to the corresponding base models with a higher overall score. Racka performs well in tasks that rely on interpretation and reasoning and a lower score on infilling and QA tasks.

Finally, we evaluate Racka on tasks from the LM-Eval-Harness Hungarian benchmark. The results are presented at the bottom of Table 3. On this benchmark the overall performance of Racka is on par with the original Qwen3-4B and

[7] All numbers are measured on the validation set.

the larger Puli model, with Qwen3-4B-Base leading the benchmarks. The results indicate that Racka exhibits reduced mathematical reasoning capabilities compared to the base model, but demonstrates improved performance on question-answering tasks. Notably, Racka outperforms PULI on TruthfulQA, contrasting with its lower performance on HuSimpleQA. We attribute this discrepancy to the broader, less culture-specific nature of TruthfulQA questions, and the limitations of Racka's culture-specific fact memorization capacity compared to a model twice its size.

6 Conclusion and Future Directions

This work introduced Racka-4B, a resource-efficient, continually pretrained language model for Hungarian, using a practical methodology specifically designed for deployment on the national Komondor HPC infrastructure. Through targeted tokenizer adaptation by expanding and initializing the vocabulary to better capture Hungarian specific vocabulary elements, we achieved a substantial reduction in subword fertility, improving both computational efficiency and the representational capacity of the model. The construction of a 160B-token multilingual corpus, with a carefully balanced data mixture, further enabled robust language adaptation while reducing catastrophic forgetting of high-resource languages.

The evaluation results confirm the effectiveness of our tokenizer adaptation and training strategy. With the generation speed improvements outlined above, the overall performance of Racka-4B is slightly higher than the base Qwen3-4B models, in some cases topping 8B models such as the latest Puli.

Future work includes systematic hyperparameter optimization, extended benchmarking and supervised fine-tuning for downstream applications as well as preference tuning. We openly publish our model and tokenizer, to support community-driven research and the advancement of digital sovereignty of Hungary.

Acknowledgement

We would like to thank Levente Szabados for the name idea and initial informal discussions.

This research was supported by the EKÖP-24 University Excellence Scholarship Program of the Ministry for Culture and Innovation, funded by the National Research, Development and Innovation Fund.

The authors acknowledge the support of the National Laboratory for Digital Heritage. Project no. 2022-2.1.1-NL-2022-00009 has been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, financed under the 2022-2.1.1-NL funding scheme.

We acknowledge the Digital Government Development and Project Management Ltd. for awarding us access to the Komondor HPC facility based in Hungary.

Bibliography

Ács, J., Lévai, D., Kornai, A.: Evaluating transferability of BERT models on Uralic languages. In: Pirinen, F.A., Arhangelskiy, T., Trosterud, T., Rießler, M. (eds.) Proceedings of the Seventh International Workshop on Computational Linguistics of Uralic Languages. pp. 8–17. Association for Computational Linguistics, Syktyvkar, Russia (Online) (Sep 2021), https://aclanthology.org/2021.iwclul-1.2/

Ali, W., Pyysalo, S.: A survey of Large Language Models for European languages (2024), https://arxiv.org/abs/2408.15040

Bergsma, S., Dey, N.S., Gosal, G., Gray, G., Soboleva, D., Hestness, J.: Straight to zero: Why linearly decaying the learning rate to zero works best for LLMs. In: The Thirteenth International Conference on Learning Representations (2025)

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., Schulman, J.: Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168 (2021)

Common Crawl: Common crawl: Get started. https://commoncrawl.org/get-started (2025), accessed: 2025-08-29

Csaki, Z., Li, B., Li, J.L., Xu, Q., Pawakapan, P., Zhang, L., Du, Y., Zhao, H., Hu, C., Thakker, U.: SambaLingo: Teaching Large Language Models new languages (2024), https://aclanthology.org/2024.mrl-1.1/

Csaki, Z., Pawakapan, P., Thakker, U., Xu, Q.: Efficiently adapting pretrained language models to new languages (2023), https://arxiv.org/abs/2311.05741

DeepSeek-AI: DeepSeek-V3 technical report (2025), https://arxiv.org/abs/2412.19437

Digital Government Development and Project Management Ltd. (DKF): Komondor HPC documentation. https://docs.hpc.dkf.hu/ (2025), the supercomputer service of the Digital Government Development and Project Management Ltd. (DKF) is used to run scientific, R&D computing tasks and to store data for scientific purposes. Accessed: 2025-08-29

Ding, H., Wang, Z., Paolini, G., Kumar, V., Deoras, A., Roth, D., Soatto, S.: Fewer truncations improve language modeling (2024), https://arxiv.org/abs/2404.10830

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., Leahy, C.: The Pile: An 800GB dataset of diverse text for language modeling (2020), https://arxiv.org/abs/2101.00027

Gao, L., Tow, J., Abbasi, B., Biderman, S., Black, S., DiPofi, A., Foster, C., Golding, L., Hsu, J., et al.: The language model evaluation harness (2024), https://zenodo.org/records/12608602

Gonzalez-Agirre, A., Pàmies, M., Llop, J., Baucells, I., Dalt, S.D., Tamayo, D., Saiz, J.J., Espuña, F., Prats, J., et al.: Salamandra technical report (2025), https://arxiv.org/abs/2502.08489

Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: LoRA: Low-Rank Adaptation of Large Language Models (2021), https://arxiv.org/abs/2106.09685

Hugging Face: 4d masks: Efficient packing for llm pretraining. https://huggingface.co/blog/poedator/4d-masks (2024), accessed: 2025-08-29

Jelinek, F., Mercer, R.L., Bahl, L.R., Baker, J.K.: Perplexity—a measure of the difficulty of speech recognition tasks. The Journal of the Acoustical Society of America 62(S1), S63–S63 (2005), https://doi.org/10.1121/1.2016299

Ke, Z., Shao, Y., Lin, H., Konishi, T., Kim, G., Liu, B.: Continual pre-training of language models (2023), https://arxiv.org/abs/2302.03241

Khade, O., Jagdale, S., Phaltankar, A., Takalikar, G., Joshi, R.: Challenges in adapting multilingual LLMs to low-resource languages using LoRA PEFT tuning (2025), https://aclanthology.org/2025.chipsal-1.22/

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C.H., Gonzalez, J., Zhang, H., Stoica, I.: Efficient memory management for large language model serving with pagedattention. In: Proceedings of the 29th Symposium on Operating Systems Principles. p. 611–626. SOSP '23, Association for Computing Machinery, New York, NY, USA (2023), https://doi.org/10.1145/3600006.3613165

Lai, V.D., Nguyen, C.V., Ngo, N.T., Nguyen, T., Dernoncourt, F., Rossi, R.A., Nguyen, T.H.: Okapi: Instruction-tuned Large Language Models in multiple languages with reinforcement learning from human feedback (2023), https://arxiv.org/abs/2307.16039

Li, C.A., Lee, H.Y.: Examining forgetting in continual pre-training of aligned Large Language Models (2024), https://arxiv.org/abs/2401.03129

Lozhkov, A., Li, R., Allal, L.B., Cassano, F., Lamy-Poirier, J., Tazi, N., Tang, A., Pykhtar, D., Liu, J., et al.: StarCoder 2 and The Stack v2: The next generation (2024), https://arxiv.org/abs/2402.19173

Ma, Y., Liu, Y., Yu, Y., Zhang, Y., Jiang, Y., Wang, C., Li, S.: At which training stage does code data help LLMs reasoning? In: The Twelfth International Conference on Learning Representations (2024), https://openreview.net/forum?id=KIPJKST4gw

Martins, P.H., Fernandes, P., Alves, J.a., Guerreiro, N.M., Rei, R., Alves, D.M., Pombal, J., Farajian, A., Faysse, M., et al.: EuroLLM: Multilingual language models for Europe. Procedia Comput. Sci. 255(C), 53–62 (2025), https://doi.org/10.1016/j.procs.2025.02.260

Mosin, V., Samenko, I., Kozlovskii, B., Tikhonov, A., Yamshchikov, I.P.: Fine-tuning transformers: Vocabulary transfer. Artificial Intelligence 317, 103860 (2023), https://www.sciencedirect.com/science/article/pii/S0004370223000061

Nemeskey, D.M.: Natural Language Processing Methods for Language Modeling. Ph.D. thesis, Eötvös Loránd University (2020)

Occiglot Project: Occiglot fineweb. https://occiglot.eu/posts/occiglot-fineweb/ (2024), accessed: 2025-08-29

OpenEuroLLM: Openeurollm-hungarian. https://ollama.com/jobautomation/OpenEuroLLM-Hungarian (2025), accessed: 2025-08-29

Penedo, G., Kydlíček, H., Allal, L.B., Lozhkov, A., Mitchell, M., Raffel, C., Von Werra, L., Wolf, T.: The FineWeb datasets: decanting the web for the finest text data at scale. In: Proceedings of the 38th International Conference on Neural Information Processing Systems. NIPS '24, Curran Associates Inc., Red Hook, NY, USA (2024)

Schuhmann, C., Beaumont, R., Assael, Y., AI, L.: LEO LM: Large Language Models for European languages. https://laion.ai/blog/leo-lm/ (2024), accessed: 2025-08-29

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: RoFormer: Enhanced transformer with rotary position embedding (2024), https://www.sciencedirect.com/science/article/pii/S0925231223011864

Szentmihályi, K., Nemeskey, D.M., Szekeres, A.M., Gortka, B.G., Indig, B., Palkó, G., Nagy, B., Vidács, L.: Pretraining GPT-style models in Hungarian. Infocommunications Journal (2024), invited paper, pre-review version

Thellmann, K., Stadler, B., Fromm, M., Schulze Buschhoff, J., Jude, A., Barth, F., Leveling, J., Flores-Herr, N., Köhler, J., Jäkel, R., Ali, M.: Towards Multilingual LLM Evaluation for European Languages. arXiv e-prints arXiv:2410.08928 (2024)

Vanroy, B.: GEITje 7B Ultra: A conversational model for Dutch (2024), https://arxiv.org/abs/2412.04092

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Gao, C., Huang, C., et al.: Qwen3 technical report (2025a), https://arxiv.org/abs/2505.09388

Yang, H., Wei, X., Wu, J., Ligeti-Nagy, N., Sun, J., Wang, Y., Yang, Gy.Z., Gao, J., Wang, J., Jiang, B., Wang, S., Yu, N., Zhang, Z., Hong, S., Liu, H., Li, W., Zhang, S., Lin, D., Wu, L., Prószéky, G., He, C.: OpenHuEval: Evaluating large language model on Hungarian specifics. In: Che, W., Nabende, J., Shutova, E., Pilehvar, M.T. (eds.) Findings of the Association for Computational Linguistics: ACL 2025. pp. 7464–7520. Association for Computational Linguistics, Vienna, Austria (2025b), https://aclanthology.org/2025.findings-acl.390/

Yang, Z.Gy., Bánfi, A., Dodé, R., Ferenczi, G., Földesi, F., Hatvani, P., Héja, E., Lengyel, M., Madarász, G., Osváth, M., Sárossy, B., Varga, K., Váradi, T., Prószéky, G., Ligeti-Nagy, N.: PULI Chat: Our first Hungarian conversational model. In: International Conference on Formal Methods and Foundations of Artificial Intelligence. pp. 1–3. Eszterházy Károly Catholic University, Eger, Hungary (2025c), https://uni-eszterhazy.hu/api/media/file/7f9158bd443acc29dbd2a211971fe8677768257c

Yang, Z.Gy., Dodé, R., Ferenczi, G., Hatvani, P., Héja, E., Lengyel, M., Ligeti-Nagy, N., Madarász, G., Sárossy, B., Varga, K., Varga, T., Váradi, T.: PULI LlumiX modell: Egy folytatólagosan előtanított nagy nyelvi modell. In: XXI. Magyar Számítógépes Nyelvészeti Konferencia. pp. 153–167. Szeged, Hungary (2025d)

Yang, Z.Gy., Laki, L.J., Váradi, T., Prószéky, G.: Mono- and multilingual GPT-3 models for hungarian. In: Text, Speech, and Dialogue. pp. 94–104. Lecture

Notes in Computer Science, Springer Nature Switzerland, Plzeň, Czech Republic (2023)

Zhong, T., Yang, Z., Liu, Z., Zhang, R., Liu, Y., Sun, H., Pan, Y., Li, Y., Zhou, Y., Jiang, H., Chen, J., Liu, T.: Opportunities and challenges of Large Language Models for low-resource languages in humanities research (2025), https://arxiv.org/abs/2412.04497

Appendix

A Tokenizer Adaptation Algorithm

Algorithm 1 Extended Tokenizer Construction for Hungarian Adaptation

Require: Original vocabulary $\mathcal{V}$ with size m; Hungarian corpus $\mathcal{C}$; target increment $n = 32{,}000$; target size $m^\star = m + n$

Ensure: Extended vocabulary $\mathcal{V}_{\text{ext}}$ of size $m^\star$

Train a BPE tokenizer of size $m^\star$ on $\mathcal{C}$ to obtain the ordered vocabulary $\mathcal{V}' = (p_0, p_1, \ldots, p_{m^\star - 1})$.
Let k be the index of the n-th token p_k in $\mathcal{V}'$ such that $p_k \notin \mathcal{V}$.
if $k = m^\star - 1$ **then**
    return $\mathcal{V}'$
else
    $\mathcal{V}'' \leftarrow$ all tokens in $\mathcal{V}$ not in $\mathcal{V}'[: k+1]$, ordered by $\mathcal{V}$'s original merge order
    return $\mathcal{V}'[: k+1]$ concatenated with $\mathcal{V}''$
end if

B Composition of the Hungarian Corpus

The table below shows the composition of the Hungarian corpus with the corresponding number of documents and BPE tokens. For Common Crawl and News, the numbers shown are after filtering and deduplication. The former represent about 30% of the pre-filtered data.

The Repositories dataset contains books and electronic journals (academic and otherwise) from the following sources:

EPA The Elektronikus Periodika Adatbázis[8] (Database of Electronic Periodicals)
MEK Magyar Elektronikus Könyvtár[9] (Hungarian Electronic Library)
OAI Open Archives Initiative repositories
OJS Open Journal System repositories

Dataset	Subset	Documents (thousands)	Tokens (millions)	Percentage
Common Crawl	`.hu`	52 775.31	40 621.05	72.71%
	`.ro`	552.19	397.70	
	`.sk`	446.77	224.60	
	`.com`	2984.91	2641.13	
Repositories	EPA	300.79	2672.38	19.48%
	Books	32.83	1338.89	
	OJS	27.70	96.36	
	OAI	349.40	7649.32	
News		5938.79	2723.87	4.5%
Court		198.30	1110.30	3.3%
HuParl		1.71	140.56	
OpenSubtitles		88.52	508.85	
Wikipedia		158.46	232.16	
Sum		63 855.67	60 357.18	100%

[8] `https://epa.oszk.hu/`
[9] `https://www.mek.oszk.hu/hu/`

C Fixing OpenHuEval for Qwen3-like Models

We have encountered several difficulties running OpenHuEval. The prompts and configurations contain minor mistakes and reasoning models, such as Qwen3, are not always evaluated correctly. Some of these problems are rooted in the OpenCompass framework itself. The most glaring issue was the omission of the OpenHuEval example scripts assumed to be present by the documentation. This was remedied by the authors after we brought it to their attention.

We have also found that the OpenHuEval tasks handled the reasoning output of models inconsistently. Some, such as HuMatchingFIB, detected the presence of the reasoning tokens and removed them before evaluation; some, such as HuSimpleQA, did not. In many cases, the presence of reasoning traces resulted in a parsing error or the model running out of the allotted context. By default, we wanted to disable reasoning in these benchmarks as small reasoning models are prone to getting stuck in thinking loops when running them with temperature 0, which is needed for reproducibility. This behavior is also outlined in the Qwen3 documentation.

Due to time constraints, we decided to fix these issues locally, with as little change as possible in the OpenCompass codebase: we modified the chat template to respect the `enable_thinking` tokenizer argument in the model configuration and disabled reasoning for the affected tasks. We made sure to leave the benchmark's datasets unchanged. The only exception was the HuWildBench task, which posed additional challenges that had to be addressed.

1. **Remove artifacts present in the Hungarian prompts only.** The HuWildBench prompt is a user prompt that is segmented into three logical units. An instruction, a question and a description. These are not delimited explicitly in the English or Chinese prompts (also provided as part of the HuWildBench dataset), however the Hungarian prompt contains the following phrases as separators attached to the end of the previous field directly without even a whitespace: "a kérdés az: ", "a leírás: ", "a válasz: " for the question, description and the indication of the answer starting point respectively. We believe that these unclear separators put directly as part of the last word erode the performance of language models, especially smaller ones. Thus we modify the prompts to keep the separators but we add two empty lines between the last character of the previous unit and the separator. We also improve the capitalization and fluency of these phrases while removing the answer start indicator as we will use the chat prompt template to denote assistant turns. This results in the following separators: "\n\nA kérdés: ", "\n\nA leírás: ".
2. **Use the proper prompt template.** The Qwen3 and Racka models use a chat prompt template with system, user and assistant message frames and are trained to follow instructions in this format. Thus, we attempted to apply the default chat template of these models in OpenCompass which setting was not respected. We opted to modify the implementation to include chat templates in HuWildBench. As Qwen3 and derivative models also need a system prompt we add a modification of the industry default "You are a helpful

assistant." adding the language constraint as "You are a helpful Hungarian assistant.". For the non-instruct Qwen3-4B-Base and PULI-LlumiX-Llama-3.1 (the latter we tested with the standard Llama-3.1 chat template which yielded slightly wores results) models we add the answer trigger prompt as "\n\nA válasz: " and do not use any special prompt template.

3. **Increasing model generation control.** As mentioned above, we make sure that the tokenizer settings are respected to avoid generating reasoning traces. We also enforce repetition penalty and frequency penalty using the definitions of vLLM Kwon et al. (2023), which are both set to 0.3 in the case of all Qwen and Racka models to avoid repetitive text generation.
4. **Fixing judge model errors.** HuWildBench is one of the tasks where the original implementation fails to remove (often English) reasoning traces from the judge model prompt. We reimplement this behavior correctly to align with the original intent present in the benchmark configuration. We also find that the judge model rarely follows the JSON schema defined in its prompts. In some cases this results in the OpenCompass implementation missing the judge score. We use a simple regex pattern to find the last "score:" in the ill-formed judge outputs and extract the first integer that follows it as a score. This method is able to correctly handle 99.91% of the failed judge prompts. When no "score:" segment is present in the judge output use a default score of 1. We enforce the usage of the single model scoring prompt to make sure OpenCompass does not automatically switch to the pairwise evaluation where a parent/base model is present.

While HuWildBench was the only subset where these errors fully prevented the original implementation from working, we believe that future work should revisit the prompts and implementation of OpenHuEval to ensure language quality and compatibility with chat and reasoning language models.

D HULU Downstream Training Details

We average results over multiple trainings to control for random factors during downstream training. The model is trained with early stopping for 10 fine tuning rounds on HuCOPA, HuWNLI and HuCommitmentBank and 5 trainings on the other subsets.

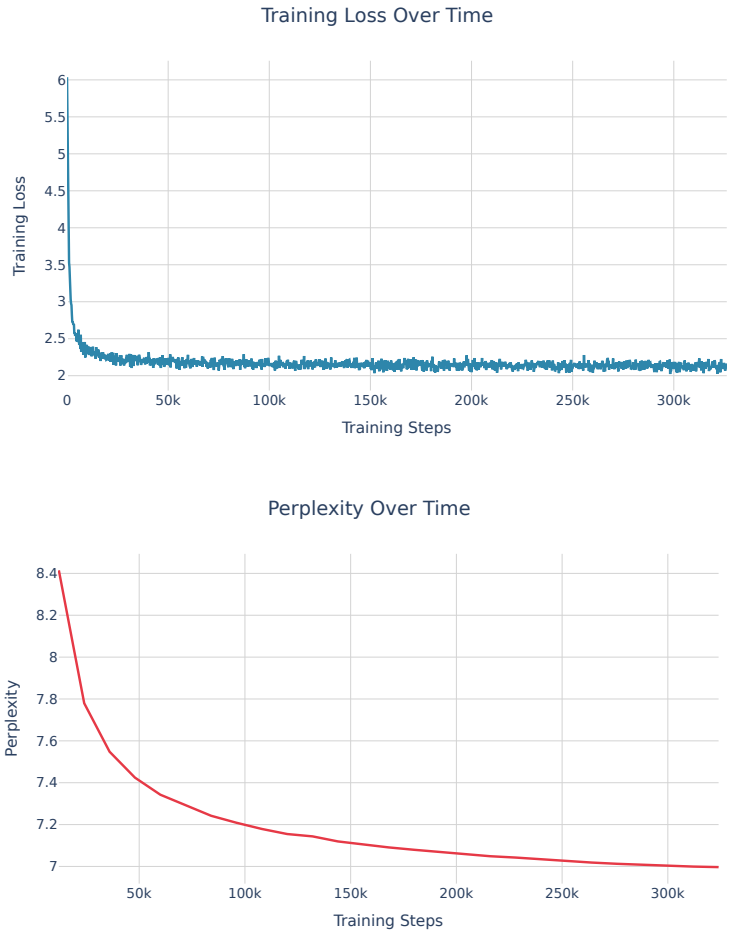
We also perform both LoRA and full fine-tuning and report the superior method in Table 3. Here we detail the best method for each model and benchmark subset in Table 4.

Table 4. Details on which fine-tuning method is superior for each model-task combination.

HULU subset	Qwen3-4B	Racka-4B	Qwen3-4B-Base	Puli-LlumiX-Llama-3.1 8B
HuCOLA	LoRA	LoRA	LoRA	LoRA
HuCOPA	LoRA	Full	LoRA	LoRA
HuSST	LoRA	LoRA	LoRA	LoRA
HuRTE	LoRA	LoRA	LoRA	LoRA
HuWNLI	Full	Full	Full	LoRA
HuCommitmentBank	Full	Full	Full	LoRA

E Loss Curves

Fig. 1 Average training Categorical-Crossentropy loss logged every 200 steps (top) and validation perplexity measured every 12 000 steps on a language-stratified held out set of documents (bottom).



Nagy nyelvmodell beszélni magyar?

Novák Attila, Novák Borbála

Pázmány Péter Katolikus Egyetem, Információs Technológiai és Bionikai Kar
Budapest, Práter u. 50/a.
{vezetéknév.keresztnév}@itk.ppke.hu

Kivonat Az érzékeny személyes adatok jelenléte számos domain esetében megakadályozza a szövegek nyilvánosságra hozatalát, és megnehezíti a hozzáférést a bennük szereplő összefüggések vizsgálatára irányuló kutatások számára. Kutatásunkban nyílt súlyú és zárt ipari nagy nyelvi modelleknek az említett probléma kezelésére hivatott magyar nyelvű pszeudonimizálási feladathoz kapcsolódó részképességeit vizsgáltuk. A névelemek adott kontextusban való nyelvileg helyes helyettesítésére való képesség felmérésére kialakított kompetenciatesztünkben a modellek egyértelműsítő minikontextusokban való névhelyettesítési képességeit mértük fel. A zárt és nem desztillált modellek jobban teljesítettek, de ezeknél is voltak meglepő hiányosságok. A nyílt súlyú modellek teljesítménye pedig kifejezetten gyengének bizonyult. Emellett a modellek magyar helyismeretét is teszteltük: képesek-e konzisztensen kezelni a címeket.

Kulcsszavak: morfológiai benchmark, pszeudonimizálás, nagy nyelvi modellek

1. Bevezetés

A ChatGPT 2022-es megjelenésével komoly paradigmaváltás következett be a nyelvtechnológiában: alapvetően emberi nyelven megfogalmazott instrukciók használatával próbálunk rávenni generikus modelleket a feladatok megoldására ahelyett, hogy példák alapján feladatspecifikus modelleket tanítanánk be. A legerősebb modellek üzemeltetése nagyon erőforrás-igényes, és csak szolgáltatásként érhetők el. Bizonyos feladattípusoknál – és ezek közé tartozik a jelen cikkben bemutatott kutatást eredetileg motiváló személyesadat-maszkolási feladat is[1] – problémát jelent azonban, hogy az adatok kezelésével megbízott adatgazda általában nem fordulhat külső szolgáltatóhoz, hogy az a feladatot elvégezze, mert ebben a felállásban nem garantálható, hogy az adatok semmilyen formában nem

[1] Jelen cikkben a terjedelmi korlátra való tekintettel nem térünk ki a pszeudonimizálás feladatának részletes bemutatására. Röviden arról van szó, hogy a szövegben minden nevet és személyes adatot konzisztensen másikra kell cserélni úgy, hogy a szöveg hibátlan maradjon, az érzékeny adatokat azonban ne lehessen visszanyerni, részletesebben l.: Novák és Novák (2024).

sziválognak ki. Ilyen esetben tehát elsősorban nyílt súlyú modellek alkalmazása jöhet szóba, amelyek általában jóval kisebb paraméterszámúak a nagy cégek egyébként többnyire ismeretlen komplexitású csúcsmodelljeinél.

Az elérhető nyelvmodellek készségeinek felmérésére és összehasonlítására rengeteg kompetenciatesztet, ú.n. benchmarkot hoztak létre, és az újonnan megjelenő modellváltozatok teljesítményét rendszeresen mérik ezeken. Emellett az emberi felhasználók (pl. az LMArenában[2]) rendszeresen mért modellpreferenciái alapján számított rangsorok is fontos referenciapontokként szolgálnak a modellpaletta folyamatos szélesedésével egyre nehezebben megválaszolható kérdésben, hogy melyik modellt válasszuk bizonyos feladatok megoldására.

A modellek tudását kvantifikálni hivatott szokásos kompetenciamérések túlnyomó többsége angol nyelven történik, esetleg kínaiul, mert a fejlesztők számára ezek a legfontosabb nyelvek. Ezek mellett még néhány más nagy nyelven nyújtott teljesítmény fontos lehet, így még a nyílt súlyú modellek egy részénél is használnak az angoltól vagy a kínaitól különböző nyelvű adatokat nemcsak a modellek nyelvi előtanításánál, hanem azoknak az elvárt viselkedések és kompetenciák elérésére hivatott finomhangolásakor is, beleértve a kisebb modellváltozatok tudásdesztillációval való létrehozásának folyamatát is. Az esetek nagy részében szinte teljes a homály ugyanakkor azzal kapcsolatban, hogy az adott modell vagy modellcsalád betanítása milyen adatokon és hogyan történt. A Google DeepMind többnyelvű, nyílt súlyú Gemma 3 modellcsaládja (Gemma Team és mtsai, 2025) esetében például annyit tudunk, hogy az előtanítás több mint 140 nyelven, a finomhangolás pedig több mint 35 nyelven történt, de semmilyen nyilvános információ nincs arról, hogy melyek ezek a nyelvek (az angolon kívül).

Az angolra leggyakrabban használt benchmarkok egy részéhez puszta gépi fordítással (Thellmann és mtsai, 2024; Dac Lai és mtsai, 2023), vagy kifinomultabb módon – részben gépi fordítást követő manuális ellenőrzéssel és javítással, vagy eleve az adott nyelvű szövegeken replikálva a benchmarkra jellemző adatannotációs folyamatot – készült számos más nyelvre, közöttük a magyarra is adaptáció (Ligeti-Nagy és mtsai, 2023, 2025; Novák és mtsai, 2023). Ezek különböző szövegértési kompetenciákat, következtetési képességeket, összefoglaló-generálási képességeket, grammatikalitási döntések meghozatalának képességét, általános common sense tudást vagy szaktudást, illetve matematikai készségeket mérnek.

Olyan jellegű kompetenciák mérésére, amelyek az angolban vagy a kínaiban nem különösebben érdekesek, ritkán áll rendelkezésre megfelelő kiértékelő tesztalmaz. Ilyen például a minket érdeklő névcsere-készség mérése, amelynek fontos összetevője, hogy a modell az adott kontextusba helyezve helyesen elragozza, és az alkalmazott névelő vagy annak hiánya szempontjából is megfelelően adaptálni tudja a behelyettesített nevet vagy egyéb adatot. Ez a feladat az angol vagy a kínai esetében nem jelent kihívást, ugyanakkor sok más nyelv esetében, ahol az esetleges jelzők és a névelők is számban, nemben és esetben egyeztetve vannak a névszói csoport fejével, még komplexebb is a feladat, mint a magyarban.

[2] `https://lmarena.ai/`

Hogy felmérjük, a jelenleg elérhető nyelvmodelleknek legalább egy töredéke hogyan teljesít egy ilyen jellegű névhelyettesítési feladaton a magyarban, az itt bemutatott kutatás keretében készítettünk egy magyar névhelyettesítési tesztalmazt. A fellelhető nagy nyelvi modellek közül nemcsak szabadon elérhető, ezért akár helyben telepíthető, hanem a nagy szolgáltatóknál üzemelő, csak távolról elérhető egyes modellek teljesítményét is megvizsgáltuk.

A pszeudonimizálási feladatban a névcserék megfelelő végrehajtása mellett fontos a cserejelöltek generálásának színvonala is. Az egyik ilyen részfeladat a helynevek és címek cseréje. Ezért egyrészt a névhelyettesítési feladatban külön figyelmet fordítottunk a helyneveknél a megfelelő helyragok használatának ellenőrzésére (pl. *Budapesten* ∼ *Győrben*), másrészt a modellek konzisztens magyarországi címgenerálási képességeit is megvizsgáltuk: ez két olyan részterület volt, amelyek megoldatlan problémát jelentettek egy korábban bemutatott magyar nyelvű pszeudonimizáló prototípusrendszerben (Novák és Novák, 2024).

2. A névhelyettesítési feladat

2.1. Korpusz, szelekció, osztályozás, annotáció

A névcsere-tesztadatbázis létrehozásakor a következő módszert követtük. Első lépésként a Webcorpus 2.0 (Nemeskey, 2020) szövegeiből kiválasztottuk azokat a mondatokat, amelyekben szerepelt valamilyen névként, mennyiségként, dátumként vagy azonosítóként azonosítható részstring, és ezeket annotáltattuk a NerKor+CARS névelem-annotáló modellel (Novák és Novák, 2022). Az annotált korpuszban szereplő nevekből gyakorisági statisztikát készítettünk azonosítva az egyes nevek szótári alakját, a névelem-annotáló modell által hozzájuk rendelt lehetséges névtípusokat, azok gyakoriságát, és a név fejének ragozott alakjait azok gyakoriságával és morfológiai elemzésével.

Az egyes neveket – amellett, hogy milyen típusú entitást jelölnek – a morfológiai és helyesírási jellemzőik szempontjából is annotáltuk. A figyelembe vett jellemzők a következők voltak:

- magánhangzóharmónia-típus,
- a tárgyrag és az eszközhatározós eset ragjának alakja (amit a szózáró mássalhangzó és a nyitótőség határoz meg),
- a tővégi magánhangzónyúlás,
- hogy kötőjellel kapcsolódnak-e hozzá a toldalékok,
- a helynévként funkcionáló nevek esetében az, hogy az irányhármasságot kifejező ragcsoportok közül melyiknek a használata jellemző az adott név esetében.

A neveket csökkenő gyakorisági sorrendben dolgoztuk fel, és azokat a neveket tartottuk meg, amelyek

- legalább százszor szerepeltek a korpuszban, és legalább hárombetűsek,

– legalább tíz különböző ragozott alakjuk előfordult a korpuszban, és az emMorph (Novák és mtsai, 2016) a korpuszból kiválasztott, kézzel ellenőrzött nevekkel kibővített változata elfogadta őket, és
– nem utalt jel masszív ingadozó viselkedésre (pl. a magánhangzó-harmónia vagy bizonyos toldalékok kapcsolásának szempontjából).

A névelem-felismerő által az adott névhez rendelt címkék közül első körben a leggyakoribbat rendeltük a névhez. Az azonosított névtípusok közül az alábbiakat tartottuk meg:

– autók (CAR),
– létesítmények (FAC) – épület, reptér, közterület, híd stb.,
– közigazgatási egységek nevei (GPE) – országok, tartományok, régiók, megyék nevei, és a település- és településrész-nevek,
– egyéb helynevek (LOC) – földrajzi nevek (víz, hegy, völgy, terület) és csillagászati objektumok nevei,
– sajtó (MEDIA) – újság, tévé, rádió, portál,
– szervezetek (ORG) – cégek, zenekarok, csapatok, közigazgatási, oktatási, katonai, pénzügyi szervezetek stb.,
– személynevek (PER) – keresztnevek, vezetéknevek, fiktív személyek nevei,
– terméknevek (PROD) – elektronikai cikkek, szoftverek, járművek, stb.,
– alkotások címei (WORK_OF_ART) – könyvek, filmek, zenék, sorozatok, színdarabok, műalkotások, stb.

Minden névelem- és ragozási típusra maximum négy példát tartottunk meg a korpuszból. Ez volt a kiinduló adatbázisunk, amelyet tovább javítottunk, finomítottunk és szűrtünk.

Az eredeti névelem-felismerő modell nem különböztet meg a fenti kategóriákon belül altípusokat. Hogy realisztikusan modellezzük a névcseréket, és ellenőrizhessük a helynevekre jellemző helytoldalék- és névelőhasználat helyességét, az egyes fő kategóriák elemeit a fenti altípusokba soroltuk. Erre a Gemini 2.5 Flash modellt használtuk, majd a kimenetet kézzel ellenőriztük. Az eredetileg rossz főkategóriába sorolt nevek besorolását javítottuk, vagy a nevet kihagytuk. Ugyancsak kihagytuk a fenti felsorolásban "stb."-vel jelölt heterogén entitáshalmazokat jelölő neveket.

Osztályozni kellett a neveket aszerint is, hogy tipikusan határozott névelővel vagy anélkül jelennek-e meg. Az altípusokon belül általában egységes a viselkedés, de vannak kivételek, például az országneveknél (*az USA, a Fülöp-szigetek, a Vatikán* ∼ *Anglia, Hollandia*), illetve a földrajzi területi egységek neveinél (*a Hanság, az Antarktisz* ∼ *Gemenc, Ázsia*). Birtokos szerkezetekben az egyébként nem névelős nevek esetében is áll(hat) határozott névelő, ugyancsak jellemző beszélt nyelvi kontextusokban az egyébként névelő nélkül használt személynevek határozott névelős használata (*a Feri*). A szinte mindig határozott névelővel álló nevek is névelő nélkül állnak számos szerkezetben (*FTC-Újpest 2:2, a folyó neve: Duna*). A tesztanyagban ezeket a kivételes kontextusokat elkerültük vagy egyértelmű környezetet teremtettünk (a birtokos szerkezeteknél), hogy megfelelően tesztelni tudjuk a modellek névelőhasználattal kapcsolatos lexikai tudását.

Ezután ellenőriztük a helynevekre jellemző helytoldalék-besorolás helyességét. Első körben a korpuszgyakorisági adatokból indultunk ki, de ez sok esetben nem adott helyes besorolást. Bizonyos névtípusoknál rendszeres többértelműségek vannak a helyragok használatával kapcsolatban (pl. a vízneveknél az „ottlét" jellegétől függően mindhárom helyviszonyrag-csoport használatos: *a Balatonban fürdünk, a Balatonon vitorlázunk,* és *a Balatonnál/on nyaralunk.*) Végül a lokatív raghasználat ellenőrzését csak azoknál a helynévtípusoknál tartottuk meg, ahol teljesen egyértelmű preferencia van, és ez az inesszívusz (*-bAn*) és társai (*-bA, -bÓl*), illetve a szuperesszívusz (*-On*) és társai (*-rA, -rÓl*) közötti szembenállás mentén rendeződik el. Ez a teljesen egyértelmű szembenállás például a közigazgatási egységek neveire, a földrajzi nevek közül a nagyobb területek vagy élőhelyek neveire, a hegy- és hegységnevekre, a bolygónevekre és asztronómiai övezetek/rendszerek neveire jellemző. A fenti osztályozásoknál szintén a Gemini 2.5 Flashsel való előannotálás + alapos kézi ellenőrzés volt az annotáció menete.

2.2. Névcsoportok és helyettesítési párok létrehozása

A szavakat a fent leírt név-alkategóriákon belül ragozási osztályokba soroltuk. Ehhez a már említett magánhangzóharmónia-típust, a toldalékolási típust (amelyet a tárgyrag és az eszközhatározós eset ragjának alakja, a tővégi magánhangzónyúlás, és a ragok kötőjeles kapcsolására vonatkozó információ együttesen határozott meg), és a (fent leírt helynévtípusoknál) a helyragosztályt használtuk. A ragozási típusokat hét nagyobb osztályba (tővégnyúlásos, egyéb magánhangzóvégű, kötőjelesen toldalékolt, digráfvégű -t ragos, digráfvégű -Vt ragos, egyéb mássalhangzóvégű -t ragos, egyéb mássalhangzóvégű -Vt ragos), majd ezeket két partícióba soroltuk úgy, hogy az egymástól legjobban különböző toldalékolási minták kerüljenek a két partícióba.

Ezek együtt (két magánhangzóharmónia-osztály, két ragozási partíció, két helynévragozási osztály) minden névtípuson belül $2 \cdot 2$ vagy a megfelelő helynévtípusoknál $2 \cdot 2 \cdot 2$ alosztályt határoztak meg. A tesztanyagban szereplő névhelyettesítési párokat úgy hoztuk létre, hogy minden pár két tagja a fenti 2-3 partíció mindegyike szempontjából ellenkező csoportba tartozik.

A tesztszavaknál nem harmonikus toldalékokat (*-é, -ék, -ként*) csak komplex toldalékszekvenciákban hagytunk meg. Mindezeknek az lett az eredménye, hogy minden kiválasztott névpár mindkét tagja a tesztanyagban szereplő minden toldalékot garantáltan más alakban vesz fel, így a ragokat nem érintő puszta stringcsere csak a teljesen ragozatlan alapalak esetében ad helyes eredményt.

Az anaforikus birtokos *-é* végződést csak biztosan egyértelmű esetekben (ahol más toldalék megelőzi: pl. *Péteremét*) hagytuk meg, ahol nem téveszthető össze a birtokos személyjellel (a *Péterét* alak önmagában többértelmű).

2.3. A teszt-szövegkörnyezetek létrehozása

A modellek névhelyettesítési/ragozási/helyesírási képességeinek vizsgálatához miniszövegkörnyezeteket hoztunk létre, amelyekben a szóalak értelmezése egyértelmű. Ezt azért csináltuk így, hogy a modellek viselkedését az eredeti feladathoz

hasonló körülmények között, de egyértelműen kiértékelhető módon vizsgálhassuk. Az elemi szövegkörnyezeteket az eredeti korpuszban szereplő morfológiailag elemzett és ellenőrzötten hibátlan szóalakokhoz hoztuk létre úgy, hogy az előálló elemi frázisban a környezet és a szóalak együttese egyértelműen meghatározza, hogy milyen szóalakot kell behelyettesíteni. A helyragos alakok vizsgálatakor kontrollként olyan kontextusok is szerepelnek, amelyekben az adott helyrag nem lokatív vagy direkcionális jelentésű, hanem az ige oblikvuszi vonzata: *eljöttünk Szenegálból* ∼ *elege volt Szenegálból.*

A modellek feladata az volt, hogy a szövegkörnyezetben azt a szóalakot hozzák létre, ami a Webkorpusz 2.0-ban eredetileg szerepelt. Az egyes célszóalakokhoz tartozó elemi feladatokat úgy hoztuk létre, hogy a névtípus és a szóalakra jellemző morfoszintaktikai címkehalmaz függvényeként előálló szövegkörnyezetbe behelyettesítettük az adott névnek az előző szakaszban leírt módon kiválasztott párját a megfelelően ragozott alakban. Ezt a ragozott alakot a név fejének lemmája és a morfoszintaktikai címkék alapján az emMorphot morfológiai generátorként használva állítottuk elő.

Az egyes névtípus + morfoszintaktikaicímke-halmaz párokhoz tartozó minikontextusok előállításához szintén megpróbáltuk a Geminit használni, de sokat kellett csiszolni a kimeneten, hogy viszonylag változatos, egyértelmű és a névelőbeillesztés szempontjából is jól kezelhető szövegeket kapjunk.

Az elemi teszt-szövegkörnyezetek generálásakor biztosítani kellett a megfelelő névelőhasználatot is. Ehhez egyrészt a korábban előállított név-névelő adatbázist használtuk, másrészt a birtokos szerkezeteknél reguláris kifejezésekkel adaptáltuk az előállt szövegeket.

Egy adott névhez maximum 20 mini-szövegkörnyezetet generáltunk. A tesztelendő szóalakokat véletlenszerűen mintavételeztük a korpuszban talált ragozott alakok közül, és véletlenszerű sorrendben adtuk vissza. Az esetek nagy részében (mintegy 80%-ában) a ragozatlan alak is bekerült a tesztalakok közé.

Hogy a kiértékelést lehetővé tegyük, minden minikörnyezethez generáltunk egy egyedi azonosítót, amely alapján a modellek által generált outputban azonosítani tudtuk, hogy melyik feladat megoldásaként hozta létre az adott modell az adott szövegrészletet (hacsak át nem írta az azonosítót is – néha előfordult ilyen). A kontextusazonosító két részből állt, az első az adott lexikai elemet (nevet), a második a konkrét alakot azonosította. Az azonosítókra azért is szükség volt, mert a modellek időnként a kontextust is átírták. Emellett a kontextusokhoz hozzárendeltük a névalak gyakoriságára és komplexitására (hány toldalékmorfémát tartalmaz) vonatkozó információt is.

A folyamat illusztrációja a függelék az 1–4 ábráin látható. Összesen 896 névpárhoz 14307 kontextus jött létre.

2.4. Vizsgált modellek

Négy olyan ipari modellt vizsgáltunk meg, amelyek csak távoli szolgáltatásként elérhetőek: Gemini 2.5 Flash, Gemini 2.5 Flash-Lite, Grok 4 Fast, Grok 4.1. Ezek mellett a következő nyílt súlyú modelleket teszteltük: Gemma 3 27B it,

Llama-3.3-70B-it, Mistral-Small-3.1-24B, Qwen3-14B. A fentiek mellett szerettük volna legalább egy magyarcentrikus modell teljesítményét is megvizsgálni. Ezek közül a chatPuli a legfrissebb modell, amelyet az NYTK által üzemeltetett online chatfelületen lehet elérni.

Az említett ipari modellekről lényegében semmit sem lehet tudni. A nyílt súlyú modellek paraméterszáma értelemszerűen a B előtti szám. Az egyes modellek által maximálisan generálható szöveg hossza 1K és 128K token közötti (részletesen l. a 3. táblázatban a függelékben). Ezek közül a Gemma kimenetitoken-limitje (8K) jelentett korlátozást abból a szempontból, hogy a népesebb névelemosztályokba tartozó neveket kisebb részadagokban kellett feldolgoztatni vele. A chatPulié pedig annyira alacsony (kb. 1000 token), hogy egyszerre két név 20-20 ragozott alakját lehetett vele előállíttatni. Így aztán a chatPuli tesztelésével hamar felhagytunk, és jóval kevesebb adatunk van róla, mint a többi modellről.

A névbeillesztési feladat, amelyet a modelleknek adtunk, nem igényel különösebb kreativitást, így az volt az intuíciónk, hogy semmi értelme a mostanában divatos töprengő modelleket használni, így a Gemini modelleket is zéró gondolkodásitoken-limittel hívtuk meg, és első körben a hőmérséklet paramétert is nullára állítottuk.

2.5. Eredmények

A modelleknek a névhelyettesítési feladaton nyújtott teljesítményét az 1. táblázatban foglaltuk össze. A mellékletben szereplő 4.–6. táblázatokban látható részletesebb hibatípus-elemzés.

Első ránézésre nem tűnnek nagyon jónak az eredmények. A legjobb eredményt a Gemini 2.5 Flash adta 16,4%-os hibaaránnyal. Ha a névelőhibáktól és a felesleges kötőjelezéstől (ami messze a leggyakoribb hibatípus[3]) eltekintve helyes a generált névalak, akkor 11,48%-os hibaarányt kapunk. Ez még mindig nem nagyon jó. Ha a becsült tokenszintű hibaarányt[4] tekintjük, akkor már kevésbé tűnik tragikusnak a helyzet – legalábbis a zárt modellek mezőnyében. Itt a Grok 4 Fast modell jobb eredményt ért el, mint a Gemini 2.5, mert ritkább szóalakokat rontott el.

A teszthalmaz fedésére vonatkozó adatoknál azért szerepel Google modelljeinél közel 300%-os érték, mert ezeken háromszor futtattuk le a tesztet, első alkalommal T=0 a második két alkalommal T=1 hőmérséklet-beállítással. A Gemini

[3] Felmerülhet magyarázatként a kötőjeles ragkapcsolás durva túlalkalmazására, hogy talán a korpuszban is sok ilyen alak szerepel, mert nagyon sok ember is azt gondolja, hogy minden idegen szóhoz minden ragot kötőjellel kell kapcsolni, azonban a Webkorpusz 2.0-ban látott alakok statisztikája csak részben támasztja alá ezt a hipotézist. A korpuszban nagyságrendekkel ritkább ez a jelenség, mint amit a modellek kimenetében tapasztalunk.

[4] Mikor tokenszintű hibaarányról beszélünk, akkor token alatt nem az LLM-zsargonban használatos szódarabtokenekre kell gondolni, hanem a klasszikus type–token szembeállításról van szó: a szóalakok hibáit olyan súllyal vesszük figyelembe, amilyen gyakori az adott alak, hogy becslést kapjunk arra, hogy valós szövegekben milyen hibaaránnyal találkoznánk.

Modell	**hiba**	**szóhiba**	**kont.-hiba**	**hiba /token**	**szóhiba /token**	**Lev. név**	**Lev. kont.**	**korp.-fedés**
Qwen3 14B	79,98	68,72	0,06	16,38	12,46	2,46	6,00	48,08
Mistral Small 3.1 24B	82,37	59,65	0,03	25,61	19,68	2,05	17,00	99,17
chatPuli	75,85	49,30	0,80	22,99	10,60	2,17	8,00	14,27
Llama 3.3 70B it	44,40	33,88	0,35	13,78	11,78	1,93	4,00	97,69
Gemma 3 27B it	42,13	32,53	0,11	19,83	18,28	1,63	14,75	298,78
Gemini 2.5 Flash-Lite	54,26	28,53	0,36	14,22	6,78	1,54	4,16	254,06
Gemini 2.5 Flash	**16,42**	**11,48**	-	3,62	1,78	1,42	-	298,78
Grok 4 Fast	18,69	15,21	0,14	**1,90**	**1,43**	1,38	4,40	98,89
Grok 4.1	19,16	16,07	-	3,55	2,95	1,80	-	90,37

1. táblázat. A modellek teljesítménye a névhelyettesítési feladaton. A táblázatban hibaarányértékek szerepelnek a generált type-ok/tokenek százalékában, az alacsonyabb érték a jobb. A szóhibaértékek a névhibák a névelőhibák és a fölösleges kötőjelezés nélkül. Kontextushiba (kont.-hiba): azok az esetek, amikor a modell módosította a név körüli szövegkörnyezetet. A tokenszintű hibaarányokat a névalakok gyakorisági adatai alapján becsültük. A Lev. értékek a hibás alakok átlagos Levenshtein-távolságát adják meg a helyes alaktól azokra az esetekre, ahol a név vagy a kontextus hibás volt. A korpuszfedésértékek azt mutatják, hogy a tesztkészletet a futtatások során mennyire sikerült lefedni (az outputból sikeresen kinyert kontextusok aránya). A Google modelleket háromszor teszteltük, innen a 300% körüli értékek. Részletesebb hibaelemzés a függelékben.

modellek háromszori tesztelésénél egyes névtípusokon viszonylag nagy szórást (akár 2-3-szoros, a becsült tokenszintű hibaarányban akár hússzoros hibaarányváltozást) mutattak az eredmények. A Gemma modellnél ugyanakkor nem tapasztaltunk számottevő ingadozást a teljesítményben.

A kisebb méretű nyílt súlyú modellek esetében (most ideértjük az egyébként nem letölthető, elvileg magyarra külön finomhangolt chatPulit is) jóval súlyosabbnak tűnik a helyzet, mint a zárt modelleknél, és ha megnézzük, hogy konkrétan tömegesen milyen szóalakokat generáltak, akkor végképp elkeserítő az eredmény. Ráadásul a teszteseteink nem voltak kifejezetten ritka nevek. Mi lenne, ha még kisebb paraméterszámú modellekkel próbálkoznánk? A chatPuli egyébként üdítően különbözik az elődeitől legalább abban, hogy érti, és végrehajtja az utasításokat, és kapunk outputot. A Gemini 3 esetében pedig valószínűleg levonhatjuk a következtetést, hogy a magyar nemcsak nem volt benne abban a bizonyos 35 nyelvben, amelyiken a sima generatív nyelvmodellezési feladaton túlmutató készségekre is tanították a modellt, hanem egyébként is túlságosan alulreprezentált volt a magyar a tudásdesztillációhoz használt transzferanyagban.

Egészében véve azonban a tokenszintű hibaarányra tett becslésünk mindenképp pesszimista felső becslésnek tekinthető, mert annak érdekében, hogy minél többféle ragozási paradigma (nagyjából egyenlően) reprezentálva legyen az anyagban, felülmintavételeztük a ritkább nehéz szavakat, és mindenképpen alul-

mintavételeztük a triviálisan megoldódó toldalékolatlan, vagy szerencsés esetben az eredeti szóalakkal azonosan toldalékolódó eseteket. A párok összeállításának folyamata és aztán a párok listájának további szűrése tovább erősítette ezt a hatást. A toldalékolatlan alakok egyébként gyakoriak a szövegekben: a nevek nagyjából 50%-ban toldalékolatlan formában szerepelnek a Webkorpuszban, a tesztanyagban ugyanakkor a toldalékolatlan alak nem szerepel a tesztben az esetek 20%-ában (és így a tokenszintű hibaarányt ezeknél a szavaknál durván felülbecsültük). Emellett a tesztgyártási folyamatban az alapvetően helyesírás-ellenőrző eredetű emMorph szűrőként alkalmazása – amit a Webkorpusz 2.0-ban szereplő viszonylag sok elemzési hiba motivált eredetileg – bevezetett egyfajta preskriptív elfogultságot az anyagba. Ezt azonban végül is vállaljuk. Ez egy nehéz benchmark, ami a helyesírási normáknak való megfelelést is méri. De mostanában egyébként is emberfeletti teljesítményt várunk a modellektől. Amilyen teljesítményt – különösen a nyílt súlyú modellek – ebben a tesztben nyújtottak, az egyelőre viszont meglehetősen emberalatti.

Emellett – különösen a Gemini modelleknél – azt tapasztaltuk, hogy egy-egy futtatásnál viszonylag konzisztensen viselkednek egy-egy név toldalékolása szempontjából, így egy-egy rossz döntéssel legtöbbször a teljes paradigma hibás lesz. Sok különálló szövegre való alkalmazás esetében ugyanakkor a különböző szövegek között nem feltétlenül ugyanazt a viselkedést mutatná a modell. A többszöri futtatásra kapott eredmények mikro-átlagolása ugyanakkor valamelyest enyhítette ezt a hatást.

Ami a konkrét hibákat illeti, az egyik legmeglepőbb tapasztalat az volt, hogy a legtöbb modell (beleértve a Gemini 2.5 Flash két és a Flash-Lite mindhárom futtatását) egyáltalán nem tudta elragozni a gyakori és egyáltalán nem rendhagyó *Anna, Éva, Mária, Chile* neveket (egyetlen helyes alakot sem sikerült generálni), hanem az *a/e* végződéshez kötőjellel kapcsolták a ragokat (a Gemini 2.5 Flashnél gyengébb modellek ráadásul teljesen rosszakat). Az *e*-végű szavak esetében ez még talán magyarázható lenne azzal, hogy viszonylag sok *e*-végű idegen név valóban kötőjellel toldalékolódik, de az *a*-végűek esetében mindenképp meglepőnek találtuk ezt a viselkedést.[5] Tény, hogy ezekben a példákban kötőjelesen ragozott nevet kellett *a*-végűvel helyettesíteni, és ez nagyon nem sikerült a legtöbb modellnek.

A Gemini 2.5 Flash és a 2.5 Flash-Lite teljesítménye között egyébként kifejezetten nagy a különbség. Az utóbbi egy erősen csökkentett paraméterszámú desztillált modell. Ha a Gemma modellsorozat története a korábbiakhoz hasonlóan folytatódik, akkor sajnos arra számíthatunk, hogy a Gemma 4 sem fog jobban teljesíteni, mint a mostani Gemini 2.5 Flash-Lite.

[5] Az *a*-végű szavak kötőjeles toldalékolása a magánhangzónyúlás jelölése nélkül a mozaikszók toldalékolására jellemző, azok azonban többnyire csupa nagybetűsek, kivéve bizonyos adónemek közkeletű rövid elnevezéseit (*áfa, szja, eva, kata*). Ezek toldalékolt alakjainak korpuszbeli előfordulása is viszonylag jelentős, illetve a *hektár* rövidített *ha* alakjának toldalékolt alakjaié is.

3. A címgenerálási feladat

A címgenerálási feladat előkészítése valamivel egyszerűbb volt. Ennek az volt a feltétele, hogy találjunk olyan adatbázist, amely alapján ellenőrizni tudjuk a generált címek helyességét. A jelenleg már egyre elterjedtebben elérhető eszközhasználatra képes modellek esetében egyébként a megfelelő erőforrás rendelkezésre állása esetén delegálhatjuk is a helyes címek generálásának feladatát egy akár determinisztikusan működő komponensnek. Ebben a feladatban azonban arra voltunk kíváncsiak, hogy mennyire tudnak egy ilyen feladatot a vizsgált modellek fejből vagy önálló eszközhasználattal (kereséssel) megoldani.

A feladathoz 21 nagyobb települést, illetve ezek között négy budapesti kerületet sorsoltunk ki, amelyek mindegyikében több mint 200 különböző közterület van, valamint 20 kisebbet, amelyek mindegyikében legalább 10. A tesztelt modellek azt a feladatot kapták, hogy legalább 10-10 különböző érvényes címet generáljanak ezeken a településeken irányítószámmal együtt. A prompt a függelékben található az 5. ábrán látható módon nézett ki.

3.1. A tesztelt modellek

Ebben a feladatban a nyílt súlyú modellek közül csak a Gemma 3 szerepelt, azonban az időközben megjelent Gemini 3 Pro és a Gemini 2.5 Pro, valamint a Gemini 2.5 Flash gondolkodó üzemmódját is teszteltük a sima 2.5 Flash és Flash-Lite valamint a két Grok modell mellett. A címek házszám részének érvényességét (megfelelő erőforrás híján) nem ellenőriztük.[6]

A 2. táblázatban látható eredményeket kaptuk az irányítószám, az közterületnév és a közterület jellege tekintetében is helyes generált címek arányát tekintve.

Az eredmény érdekessége, hogy a messze legjobban teljesítő Gemini 3 Pro a levezetési folyamattal kapcsolatban visszaadott visszajelzése szerint nem alkalmazott keresést vagy más eszközt a címek generálásakor, hanem a belső tudására támaszkodott. Ha ez igaz, akkor mindenképpen impresszívnek tekinthető, hogy a generált címek kétharmada az irányítószám tekintetében is hibátlan volt (ez a nagyobb településeken levő címek esetében nem triviális, mert ezeknél több postai körzet van). A hibás címek szinte mindegyikében is csak apró hiba csúszott be: 1 számjegy nem jó az irányítószámból, a közterület jellege nem stimmel, Petőfi helyett Petőfi Sándor utca, stb.

Ebből a feladatból azt is megtudhattuk, hogy jó eséllyel beválik, ha az alábbi utcanevekre tippelünk: *Ady Endre utca, Kossuth Lajos* vagy *Kossuth utca, Dózsa György utca, Arany János utca, Petőfi Sándor* vagy *Petőfi utca, József*

[6] A Grok 4.1 egyébként elvérzett volna a házszám-ellenőrzésen, mert kettesével növekvő páratlan házszámokat generált a teljes címsorozatban, így a kicsi településeken már sokszázas házszámok szerepeltek minden utcánál. Debrecenben és a budapesti kerületekben az irányítószámokat is sorban generálta az egyes címekhez, így a budapesti kerületeknél az utolsó címhez (Erzsébet esetében az utolsó öthöz) már másik kerülethez tartozó irányítószám jött létre.

Modell	pontosság	helyes cím
Gemma 3 27B it	0.16	64/410
Gemini 2.5 Flash lite	0.19	78/410
Gemini 2.5 Flash	0.36	148/410
Gemini 2.5 Flash Thinking	0.37	152/410
Gemini 2.5 Pro Thinking	0.40	166/410
Gemini 3 Pro Thinking	**0.65**	268/410
Grok 4 Fast	0.33	137/410
Grok 4.1	0.29	117/410

2. táblázat. A címgenerálási feladat eredményei. A teljesen helyes generált címek aránya és száma.

Attila utca, Rákóczi utca, Béke utca. Egy olyan baseline rendszer, amely a település/kerület leggyakoribb irányítószámát és az előbbi 10 leggyakoribb utcanevet használva generál címeket, 41%-os pontosságot ér el, ami a Gemini 3 Pro-n kívül minden rendszert ver (azokat is, amelyek keresést használtak).

4. Előzmények

Az itt bemutatott morfológiai generálási kompetenciateszt különbözik a legtöbb jelenleg népszerű teszttől abban, hogy azok – közöttük a bevezetésben említett többek között magyarra is adaptált benchmarkok nagy része is – általában a modellek szövegértési, illetve magas szintű kognitív képességeit mérik. Ezeknél a forrásnyelv inkább csak proxy, aminek a nüanszai sokszor korlátozott mértékben érintik az adott tesztben nyújtott teljesítményt (kivéve azokat az eseteket, amikor az adott nyelv jellegzetességeiből adódóan sokkal könnyebb lesz a teszt a fordítás következtében, és megszűnik azt mérni, amire eredetileg hivatott, mint például a Winograd-sémák nagy részének lefordításakor olyan nyelvekre, amelyekben van grammatikai nem). Vannak ezek mellett grammatikus-agrammatikus minimálpárokon alapuló tesztkészletek sok nyelvre (pl. Warstadt és mtsai (2020); Başar és mtsai (2025)) – és a HuLU (Ligeti-Nagy és mtsai, 2023) is tartalmaz grammatikalitási döntési feladatokat –, amelyek ugyan közvetlenül nyelvspecifikus grammatikai tudás (és nem valamilyen magasabb kognitív készség vagy tudás) mérésére szolgálnak, de megint csak perceptív és nem generatív fókuszúak.

A generatív képességek objektív mérése általában elég nehéz, mert nagyon változatos szövegek jönnek elő a modellekből, amelyeknek az automatikus kiértékelése nehéz. Erre „nagy nyelvmodell(ek) mint bírá(k)” jellegű rendszereket lett szokás használni, de ezzel is vannak mindenféle problémák, egyrészt például a bírák modelltársaikkal szembeni elfogultságai, másrészt azok gyors elavulása, amikor a kicsit újabb keletű jelölt képességei már meghaladják a kicsit régebbi keletű bírák képességeit.

Itt mi viszonylag szigorúan megszorított generálási feladatokat adunk a modelleknek, és nagyon konkrét morfológiai és helyesírási készségeket mérünk, ami

könnyebb objektív kiértékelést tesz lehetővé. Ennek természetesen vannak előzményei. Kettőt említünk ezek közül: a SIGMORPHON Shared Task sorozat 2016 és 2023 közötti kiadásaiban szerepelt morfológiai ragozási feladat sok nyelven. Azonban ezekben a shared taskokban adatalapú morfológiai generátort kellett építeni, amely egy lemma és morfoszintaktikai jegyhalmaz alapján legyártja a megfelelő szóalakot. Ilyen jellegű feladatot emberek csak megfelelő felkészítés után tudnak megoldani (valószínűleg jóval több hibával, mint amennyivel egy minta alapján elragoznának egy másik szót). A ChatGPT megjelenése után bekövetkezett paradigmaváltás azután teljesen marginalizálta ezt a kutatási irányt (is). A másik kutatásban, amit meg szeretnénk említeni, azt a feladatot adták nagy nyelvmodelleknek, hogy sokmorfémás finn és török szavakat állítsanak helyre összekevert toldalékmorféma-sorozatok sorbarendezésével (Ismayilzada és mtsai, 2025). Ez a feladat szintén viszonylag mesterkélt. Mi ezzel szemben természetes, hétköznapi emberek által is megoldható, valódi alkalmazás által motivált feladatot adtunk a modelleknek, ahol izolált generálás helyett kontextusban helyettesítés a feladat, amelyhez egyszerre van szükség értelmezési és generálási készségre, de mindkettő külön finomhangolás nélkül rendelkezésre kellene, hogy álljon a modellek számára. Tény, hogy ehhez megfelelő helyesírási kompetencia is kell, és nehéz idegen nevek is szerepelnek az anyagban, amelyeknek helyes elragozásához a kiejtést jól meg kell tippelni.

5. Összefoglalás

Cikkünkben nagy nyelvmodellek két, a pszeudonimizálási feladathoz köthető magyar nyelvű generatív részképességének mérésére szolgáló feladatot és számos kurrens modell ezeken a teszteken nyújtott teljesítményét mutattuk be. A kontextusban való névhelyettesítési feladatot hiánypótlónak érezzük, amely talán tényleg a modellek generatív nyelvtudásának egy aspektusát méri. A zárt és nem desztillált modellek jobban teljesítettek, de ezeknél is voltak meglepő hiányosságok. A nyílt súlyú modellek teljesítménye pedig kifejezetten gyengének bizonyult, bár az ebben kategóriában nagy paraméterszámúnak számító modellváltozatokat teszteltünk. Ugyanakkor az eredmények alapján becsült tokenszintű névcsere-hibaarány biztosan pesszimista felső becslés, mert a teszt összeállítása során felülmintavételeztük a nehéz idegen neveket, és alulmintavételeztük a triviálisan megoldódó toldalékolatlan eseteket. A másik, címgenerálással kapcsolatos tesztben a legfrissebb Gemini 3 Pro modell kifejezetten jól szerepelt.

Köszönetnyilvánítás

A kutatás az Európai Unió támogatásával valósult meg az RRF-2.3.1-21- 2022-00004 azonosítójú Mesterséges Intelligencia Nemzeti Laboratórium projekt keretében.

Hivatkozások

Başar, E., Padovani, F., Jumelet, J., Bisazza, A.: TurBLiMP: A Turkish benchmark of linguistic minimal pairs. In: Christodoulopoulos, C., Chakraborty, T., Rose, C., Peng, V. (szerk.) Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing. pp. 16506–16521. Association for Computational Linguistics, Suzhou, China (Nov 2025), https://aclanthology.org/2025.emnlp-main.834/

Dac Lai, V., Van Nguyen, C., Ngo, N.T., Nguyen, T., Dernoncourt, F., Rossi, R.A., Nguyen, T.H.: Okapi: Instruction-tuned large language models in multiple languages with reinforcement learning from human feedback. arXiv e-prints pp. arXiv–2307 (2023)

Gemma Team, és mtsai: Gemma 3 technical report (2025), https://arxiv.org/abs/2503.19786

Ismayilzada, M., Circi, D., Sälevä, J., Sirin, H., Köksal, A., Dhingra, B., Bosselut, A., Ataman, D., Plas, L.V.D.: Evaluating morphological compositional generalization in large language models. In: Chiruzzo, L., Ritter, A., Wang, L. (szerk.) Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers). pp. 1270–1305. Association for Computational Linguistics, Albuquerque, New Mexico (Apr 2025), https://aclanthology.org/2025.naacl-long.59/

Ligeti-Nagy, N., Madarasz, G., Foldesi, F., Lengyel, M., Osvath, M., Sarossy, B., Varga, K., Yang, G.Z., Héja, E., Váradi, T., Prószéky, G.: HuGME: A benchmark system for evaluating Hungarian generative LLMs. In: Arviv, O., Clinciu, M., Dhole, K., Dror, R., Gehrmann, S., Habba, E., Itzhak, I., Mille, S., Perlitz, Y., Santus, E., Sedoc, J., Shmueli Scheuer, M., Stanovsky, G., Tafjord, O. (szerk.) Proceedings of the Fourth Workshop on Generation, Evaluation and Metrics (GEM²). pp. 385–403. Association for Computational Linguistics, Vienna, Austria and virtual meeting (Jul 2025), https://aclanthology.org/2025.gem-1.32/

Ligeti-Nagy, N., Héja, E., Laki, L.J., Takács, D., Yang, Z.G., Váradi, T.: Hát te mekkorát nőttél! - a HuLU első életéve új adatbázisokkal és webszolgáltatással. In: Gábor, B., Gergely, G., Veronika, V. (szerk.) XIX. Magyar Számítógépes Nyelvészeti Konferencia. pp. 217–230. Szegedi Tudományegyetem, Szeged (2023), http://acta.bibl.u-szeged.hu/78415

Nemeskey, D.M.: Natural Language Processing Methods for Language Modeling. Ph.D.-értekezés, Eötvös Loránd University (2020)

Novák, A., Novák, B.: NerKor+Cars-OntoNotes++. In: Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC 2022). pp. 1907–1916. European Language Resources Association, Marseille, France (Jun 2022), https://aclanthology.org/2022.lrec-1.203

Novák, A., Novák, B., Zombori, T., Szabó, G., Szántó, Z., Farkas, R.: A question answering benchmark database for Hungarian. In: Prange, J., Friedrich, A. (szerk.) Proceedings of the 17th Linguistic Annotation Workshop (LAW-

XVII). pp. 188–198. Association for Computational Linguistics, Toronto, Canada (Jul 2023), https://aclanthology.org/2023.law-1.19/

Novák, A., Novák, B.: Személyes adatok azonosítása és automatikus lecserélése magyar nyelvű szövegekben. In: Berend, G., Gosztolya, G., Vincze, V. (szerk.) XX. Magyar Számítógépes Nyelvészeti Konferencia. pp. 117–129. Szegedi Tudományegyetem (2024), 267 p., 13 p.

Novák, A., Siklósi, B., Oravecz, C.: A new integrated open-source morphological analyzer for Hungarian. In: Calzolari, N., Choukri, K., Declerck, T., Goggi, S., Grobelnik, M., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S. (szerk.) Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016). pp. 1315–1322. European Language Resources Association (ELRA), Portorož, Slovenia (May 2016), https://aclanthology.org/L16-1209

Thellmann, K., Stadler, B., Fromm, M., Schulze Buschhoff, J., Jude, A., Barth, F., Leveling, J., Flores-Herr, N., Köhler, J., Jäkel, R., Ali, M.: Towards multilingual LLM evaluation for European languages (2024), arxiv.org

Warstadt, A., Parrish, A., Liu, H., Mohananey, A., Peng, W., Wang, S.F., Bowman, S.R.: BLiMP: The benchmark of linguistic minimal pairs for English. Transactions of the Association for Computational Linguistics 8, 377–392 (2020), https://aclanthology.org/2020.tacl-1.25/

6. Függelék – Illusztrációk, táblázatok, promptok

A függelékben illusztráljuk a névcsere-benchmark létrehozásának folyamatát a *Szenegál* országnév ragozásához tartozó minikontextusok előállításának lépéseivel. Itt szerepel emellett a tesztelt modellek kimenetitoken-limitjeit összefoglaló táblázat, a névhelyettesítési feladatban szereplő modellek részletes hibatípus-elemzését bemutató táblázatok és a címgenerálási feladatban használt prompt.

```
GPE_CNTRY,B,Ine,,t/l.C$-t.0 4753 Szenegál !!GPE 392 LOC 168 ORG 84 PER
4 WORK_OF_ART 3 EVENT 2 PROJ 1 MISC 1 LANGUAGE
3065 Szenegál[/N][Nom] 876 Szenegálban[/N][Ine] 288 Szenegálnak[/N][Dat]
255 Szenegálba[/N][Ill] 227 Szenegálból[/N][Ela] 206 Szenegált[/N][Acc]
172 Szenegáltól[/N][Abl] 115 Szenegállal[/N][Ins] 63 Szenegálig[/N][Ter]
44 Szenegálra[/N][Subl] 41 Szenegálról[/N][Del] 31 Szenegálon[/N][Supe]
9 Szenegálhoz[/N][All] 5 Szenegálnál[/N][Ade]
```

1. ábra. Azonosított névaltípussal és toldalékolási osztállyal kiegészített korpuszstatsztika a Webkorpusz 2.0-ból a *Szenegál* országnévhez. Név-altípus: GPE_CNTRY, harmóniatípus: B, helyragosztály: Ine, tárgy/eszk.hat. t/l, mássalhangzóvégű -t ragos, 0-s ragozási partíció. Korpuszstatisztika: névgyakoriság: 4753+392+..., szóalakok gyakoriséggal, elemzéssel)

```
Szenegál: 113606 172/1 féltem <Szenegáltól>; 113607 115/1 nincs bajom
<Szenegállal>; 113600 3065/0 <Szenegál> gyönyörű; 113602 288/1 köszönöm
<Szenegálnak>; 113612 9/1 <Szenegálhoz> képest szép; 113613 5/1 szebb
<Szenegálnál>; 113605 206/1 láttam <Szenegált>
L: Szenegál: 113601 876/1 éltem <Szenegálban>; 113603 255/1 elutaztam
<Szenegálba>; 113604 227/1 <Szenegálból> jövök; 113609 44/1 hasonlít
<Szenegálra>; 113610 41/1 meséltem <Szenegálról>; 113611 31/1 végigvonult
<Szenegálon>
```

2. ábra. A *Szenegál* országnévhez generált minikontextusok az eredeti szóalakokkal. Ez a névhelyettesítési feladat helyes megoldása (a kontextus és a fókusz jelölésével).

```
Azori-szigetek > Szenegál
Azori-szigetek: 113606 172/1 féltem <az Azori-szigetek[/N][Abl]>;
113607 115/1 nincs bajom <az Azori-szigetek[/N][Ins]>; ...
113605 206/1 láttam <az Azori-szigetek[/N][Acc]>;
L: Azori-szigetek: 113601 876/1 éltem <az Azori-szigetek[/N][Supe]>; ...
113611 31/1 végigvonult <az Azori-szigetek[/N][Supe]>;
```

3. ábra. A *Szenegál* országnévhez (B,Ine,0 toldalékolási osztály) generált minikontextusok a helyettesítő névvel (*Azori-szigetek*, F,Supe,1 toldalékolási osztály) a morfológiai generálás előtt.

```
Take the following pairs of named entites. Replace all occurrences of the
first entity with the second one in all of the following short contexts
and at the beginning of the line properly reinflecting the substitute
name (and possibly adding/removing a definite article) according to the
context. Remove these instructions from the output.

Azori-szigetek > Szenegál
Azori-szigetek: 113606 172/1 féltem az Azori-szigetektől; 113607 115/1
nincs bajom az Azori-szigetekkel; 113600 3065/0 az Azori-szigetek gyönyörű; ...
L: Azori-szigetek: 113601 876/1 éltem az Azori-szigeteken; 113603 255/1
elutaztam az Azori-szigetekre; 113604 227/1 az Azori-szigetekről jövök; ...
```

4. ábra. A *Szenegál* országnévhez generált minikontextusok a helyettesítő névvel (*Azori-szigetek*) a morfológiai generálás után. Így néz ki a névhelyettesítési feladat a modellek számára. A példa az alkalmazott promptot is tartalmazza.

Modell	kimenet max. hossza
Gemini 2.5 Flash/Flash-Lite	64K
Grok 4 Fast/4.1	30K
Gemma 3 27B it	8K[1]
Mistral-Small-3.1-24B	128K
Qwen3-14B	41K[2]
chatPuli	1K?

3. táblázat. A névcsere-feladatban vizsgált modellek kimenetitokenszám-limitjei. [1]Tapasztalati érték a `https://generativelanguage.googleapis.com` API-n keresztül való elérés esetén. A Gemini 2.0 modellcsaládra is érvényes, amelyből a Gemma 3 modellcsaládot desztillálták. [2]bemenet+kimenet

modell	Qwen3-14B		Mistral-S-3.1-24B		chatPuli	
hibatípus	type	token	type	token	type	token
hiba	79,98	16,38	82,37	25,61	75,85	22,99
hiba a névben	79,98	16,38	82,37	25,61	75,85	22,99
hibás szóalak	68,72	12,46	59,65	19,68	49,30	10,60
extra kötőjel	52,25	7,62	58,67	21,52	66,07	21,26
csak extra kötőjel	6,93	1,35	20,96	5,47	20,76	10,99
rossz helyrag	6,87	6,06	11,20	13,13	1,60	0,22
névelőhiba	15,70	3,03	4,94	0,53	10,38	1,50
csak névelőhiba	4,32	2,57	1,46	0,45	5,79	1,41
nem cserélte le	2,96	0,59	3,91	0,51	-	-
kötőjelhiány	0,59	0,01	1,15	0,09	-	-
csak kötőjelhiány	-	-	0,43	0,01	-	-
kontextushiba	0,06	-	0,03	-	0,80	-
balkontextus-hiba	0,06	-	0,03	-	0,60	-
jobbkontextus-hiba	-	-	-	-	0,20	-
hibás kontextus ID	-	-	-	-	-	-
névtávolság	2,46	2,08	2,05	1,71	2,17	1,50
kontextustávolság	6,00	-	17,00	-	8,00	-
tesztkészlet lefedése	48,08	-	99,17	-	14,27	-

4. táblázat. Részletes hibaelemzés – névhelyettesítési feladat: a leggyengébb teljesítményt nyújtó nyílt súlyú modellek. A hibatípusok gyakoriság szerint csökkenő sorrendben szerepelnek az összes modell aggregált hibáit alapul véve. Az értékek az mutatják, hogy a generált név type-ok hány százaléka hibás. A tokenszintű hibaarányokat a korpuszgyakorisági értékek alapján becsültük. A távolságok átlagos Levenshtein-távolságok a helyes alaktól név- vagy kontextushiba esetén.

modell	Llama-33-70B-it		Gemma-3-27b-it		Gemini-2.5-flash-lite	
hibatípus	type	token	type	token	type	token
hiba	44,40	13,78	42,13	19,83	54,26	14,22
hiba a névben	44,31	13,77	42,13	19,83	53,67	12,48
hibás szóalak	33,88	11,78	32,53	18,28	28,53	6,78
extra kötőjel	15,19	3,25	8,06	1,35	32,42	7,21
csak extra kötőjel	7,11	1,43	2,28	0,13	23,62	5,01
rossz helyrag	1,75	0,75	10,17	14,84	6,50	3,25
névelőhiba	2,59	0,50	7,16	1,49	1,73	0,72
csak névelőhiba	1,92	0,41	4,66	1,26	1,30	0,65
nem cserélte le	2,86	1,67	1,54	1,02	1,40	0,71
kötőjelhiány	3,06	0,25	3,68	0,21	0,55	0,05
csak kötőjelhiány	1,40	0,16	2,66	0,17	0,21	0,04
kontextushiba	0,35	-	0,11	-	0,36	-
balkontextus-hiba	0,23	-	0,10	-	0,34	-
jobbkontextus-hiba	0,06	-	0,10	-	0,02	-
hibás kontextus ID	0,06	-	-	-	0,02	-
névtávolság	1,93	2,23	1,63	1,42	1,54	1,75
kontextustávolság	4,00	-	14,75	-	4,16	-
tesztkészlet lefedése	97,69	-	298,78	-	254,06	-

5. táblázat. Részletes hibaelemzés – névhelyettesítési feladat: a középmezőny – nyílt súlyú és egyéb desztillált modellek.

modell	Gemini-2.5-flash		Grok4-fast		Grok4.1	
hibatípus	type	token	type	token	type	token
hiba	16,42	3,62	18,69	1,90	19,16	3,55
hiba a névben	16,39	3,62	18,66	1,90	19,16	3,55
hibás szóalak	11,48	1,78	15,21	1,43	16,07	2,95
extra kötőjel	6,13	2,12	3,28	0,14	1,39	0,10
csak extra kötőjel	3,49	1,22	0,84	0,04	0,72	0,05
rossz helyrag	4,15	0,72	6,71	1,06	5,42	1,64
névelőhiba	1,14	0,48	1,15	0,48	1,20	0,53
csak névelőhiba	1,03	0,47	0,95	0,39	0,91	0,51
nem cserélte le	0,20	0,21	-	-	0,88	0,85
kötőjelhiány	0,75	0,16	2,56	0,08	2,52	0,07
csak kötőjelhiány	0,39	0,14	1,67	0,04	1,45	0,04
kontextushiba	-	-	0,14	-	-	-
balkontextus-hiba	-	-	0,03	-	-	-
jobbkontextus-hiba	-	-	0,12	-	-	-
hibás kontextus ID	-	-	-	-	-	-
névtávolság	1,42	1,54	1,38	1,61	1,80	3,26
kontextustávolság	-	-	4,40	-	-	-
tesztkészlet lefedése	298,78	-	98,89	-	90,37	-

6. táblázat. Részletes hibaelemzés – névhelyettesítési feladat: zárt súlyú modellek.

For all of the following settlements, generate 10 valid addresses each with zip code, each in a different street, and output them as a yaml map with the settlement name as key, and a list of address strings as value in yaml block format. The format of the addresses should be: zip settlement street address house number.
Vác; Békés; Budapest 12. kerület; Szigetszentmiklós; Debrecen; Budapest 16. kerület; Érd; Balatonfüred; Tiszafüred; Cegléd; Budapest 20. kerület; Budapest 03. kerület; Biatorbágy; Dunakeszi; Karcag; Gyomaendrőd; Törökszentmiklós; Dunaharaszti; Fót; Veresegyház; Baja;
Ábrahámhegy; Söjtör; Hajdúböszörmény; Rezi; Tenk; Halimba; Bük; Egyek; Nemti; Bojt; Csengersima; Keszü; Mátramindszent; Bácsalmás; Bükkszentkereszt; Nagyveleg; Nyírtura; Balatonboglár; Vilonya; Szamossályi

5. ábra. A címgenerálási feladatban alkalmazott prompt.

Assessing Retrieval Performance and Chunking Strategies in Hungarian RAG Systems

Edis Hasaj[1], András Simonyi[2], Dávid Márk Nemeskey[3]

[1]Eötvös Loránd University (ELTE)
edis@student.elte.hu
[2]ELTE Faculty of Informatics, Department of Artificial Intelligence
simonyi@inf.elte.hu
[3]ELTE Faculty of Humanities, Department of Digital Humanities
nemeskey.david@btk.elte.hu

Abstract. This paper presents a systematic evaluation of Retrieval-Augmented Generation (RAG) components for Hungarian question answering, focusing on retrieval effectiveness and chunking strategies on long-form Hungarian historical texts. Using the MILQA benchmark and a custom corpus of full-length books, we compare sparse, dense, hybrid, and reranking approaches under uniform metrics. Among dense models, E5-Large performs best, clearly surpassing other multilingual encoders and the BM25 baseline. Query rewriting shows mixed effects: it harms dense retrieval but Query2Doc notably improves BM25. The strongest overall results come from a hybrid BM25 + Query2Doc + E5-Large pipeline fused with Reciprocal Rank Fusion. For chunking, recursive segmentation at 512 tokens outperforms several semantic methods, which tend to over-fragment long documents. Multilingual cross-encoder reranking, especially BGE-rr, provides additional gains. Overall, the study offers practical guidance for building high-accuracy Hungarian RAG systems and underscores the importance of optimized retrieval and chunking in low-resource settings.
Keywords: information retrieval, RAG, embeddings

1 Introduction

Large-scale multilingual language models have made significant advancements in recent years, allowing for improved natural language understanding and generation across multiple languages. However, many languages, including Hungarian, remain low-resource, meaning they have significantly less training data available compared to high-resource languages like English or French. This results in lower performance in both retrieval and generation tasks, raising concerns about the generalization abilities of state-of-the-art multilingual models.

One promising approach to improving language model outputs is Retrieval-Augmented Generation (RAG) (Lewis et al., 2021), which enhances text generation by incorporating external knowledge retrieval. While RAG has shown impressive results in high-resource settings, its effectiveness for Hungarian remains underexplored, at least academically. Additionally, improving retrieval

mechanisms, particularly through query rewriting techniques leveraging multilingual Large Language Models (LLMs), could potentially enhance retrieval performance, leading to better overall responses in Hungarian.

Beyond these general motivations, this research is directly driven by a practical application: the development of an interactive Hungarian-language chatbot designed for the 200th anniversary exhibition of the Hungarian Academy of Sciences. The chatbot's purpose is to provide accurate, verifiable information about the Academy's history, its founding figures, and its institutional evolution. Achieving this requires reliable retrieval from long, complex Hungarian historical texts, many of which exhibit dense syntax, long sentences, and substantial morphological variation. Since factual correctness is paramount in this public-facing setting, the quality of retrieval—and the chunking and reranking decisions that support it—plays a decisive role. This context further underscores the necessity of systematically evaluating RAG components for Hungarian, particularly when working with domain-specific corpora derived from historical books and archival materials related to the Academy.

While retrieval-augmented generation has seen rapid progress in recent years, most RAG systems and evaluation benchmarks are heavily centered on English. Even models described as multilingual are typically optimized and assessed using English-dominant datasets, and performance parity across languages cannot be assumed(Zhang et al., 2023). Hungarian, in particular, presents a challenging testbed due to its agglutinative morphology, free word order, and limited presence in widely used evaluation benchmarks(Orosz et al., 2023). From a model perspective, transformer-based RAG systems introduce additional challenges for Hungarian: subword tokenization schemes interact poorly with agglutinative morphology, resulting in longer and more fragmented token sequences than in English, which in turn increases pressure on fixed context windows and reduces the number of retrieved passages that can be jointly processed(Csaki et al., 2023). Although robust retrieval can be achieved through language-specific preprocessing and hybrid retrieval strategies, doing so requires additional engineering effort compared to English pipelines, and downstream generation quality remains constrained by English-dominant instruction tuning and evaluation practices. Together, these factors make Hungarian RAG an informative case for understanding the limits of current multilingual RAG approaches.

2 Related work

Research on Hungarian-language information retrieval (IR) has historically been shaped by the challenges posed by the language's rich morphology. Traditional sparse retrieval approaches—such as TF–IDF and BM25 (Robertson et al., 1994), represent documents through surface-level lexical features: words or stems. This approach can achieve high precision, but it often leads to reduced effectiveness in recall when inflected word forms or compound structures vary widely (Manning et al., 2008). Sparse retrieval methods also struggle to capture semantic similarity or relatedness, since they rely purely on exact or near-exact lexical matching.

This becomes a particular challenge in Hungarian, where extensive inflection and derivation greatly reduce the chance that a query and a relevant document will share the same word forms (Savoy, 2008). Although lemmatization helps normalize some of this variation, it cannot fully resolve the vocabulary mismatch introduced by the language's rich morphology.

The introduction of contextualized embeddings and transformer-based models has significantly improved retrieval quality in multilingual and low-resource languages. Dense vector models such as BERT (Devlin et al., 2019) produce contextual embeddings that better capture semantic similarity across different word forms and contexts. When fine-tuned for information retrieval (Khattab and Zaharia, 2020; Chen et al., 2024b), dense models overcome the vocabulary mismatch that exists between documents and queries (Qiao et al., 2019).

The effectiveness of dense models notwithstanding, often the best performance is achieved by combining sparse and dense signals in a hybrid retrieval setting (Chen et al., 2024b). Reciprocal Rank Fusion (RRF) is frequently employed to merge rankings, benefiting from the complementary strengths of lexical precision and semantic generalization (Cormack et al., 2009).

Query rewriting has traditionally been used to enhance retrieval performance by modifying user queries through pseudo-relevance feedback (Attar and Fraenkel, 1977; Xu and Croft, 1996) or external lexical resources such as WordNet (Miller, 1995), expanding them with terms more likely to retrieve relevant documents (Wang et al., 2023). Subsequent relevance-modeling approaches shifted the focus from explicit term matching toward modeling query generation itself, treating queries as samples drawn from latent document models and ranking documents according to the probability of generating the observed query (Lavrenko and Croft, 2001). More recent rewriting techniques make use of large language models to generate richer semantic representations of user intent. For example, Query2Doc creates pseudo-documents through few-shot prompting of LLMs and concatenates them with the original query, helping both sparse and dense retrievers better capture the underlying information need (Wang et al., 2023). HyDE follows a similar philosophy by generating synthetic passages that hypothetically answer the query, embedding these documents using a contrastively trained encoder to form the query representation (Gao et al., 2022). Such rewriting methods have shown promise in multilingual and low-resource retrieval settings, though their effectiveness depends strongly on the underlying retrieval model.

In modern retrieval pipelines, first-stage retrieval is often followed by reranking using cross-encoders, which jointly encode the query and candidate passages within a single transformer model (Humeau et al., 2020). Unlike the bi-encoders used in the first stage (Khattab and Zaharia, 2020), which produce independent embeddings for queries and documents, cross-encoders allow deep bidirectional attention between the two sequences, enabling much finer interaction modeling. This often leads to substantially higher reranking accuracy, albeit at a significant computational cost due to the need to jointly encode each query–document pair. As a result, cross-encoders are typically used as a second-stage reranking module to refine a small set of candidates retrieved by faster sparse or dense methods.

Finally, text chunking plays an essential role in retrieval-augmented systems, especially when long documents must be segmented into manageable units. Token-constrained, recursive chunking methods—such as those implemented in the `semchunk`[1] library—aim to preserve semantic coherence by splitting text along meaningful structural boundaries while enforcing strict token limits (Isaacus, 2025). Semantic chunking techniques further leverage embeddings to detect shifts in meaning between adjacent sentence groups, selecting breakpoints where semantic distances exceed a learned or statistically derived threshold (McCormick, 2024). Both approaches provide more coherent and contextually stable segments than naive fixed-length splitting, improving both retrieval precision and the quality of downstream generation in RAG pipelines.

Recent work on Hungarian RAG systems has addressed the technical challenges of processing complex, unstructured documents. A recent advancement is the development a pipeline for an educational chatbot, focusing on the ingestion of scanned PDF textbooks containing images and discontinuous text. They demonstrated that vision-enabled large language models (LLMs), particularly closed models like GPT-4o, significantly outperform traditional OCR and rule-based parsers, achieving near-perfect accuracy in text extraction and structural parsing. For the retrieval and ranking stage, they found that a hybrid approach combining BM25 and dense vector search (FAISS), fused with Reciprocal Rank Fusion (RRF), yielded the best performance. Evaluating on 69 textbook questions (43 multiple-choice, 16 true/false, 10 essay), they reported that GPT-4-turbo with this hybrid RAG method achieved 98.55% accuracy when essay answers were judged by GPT-4o with a correctness threshold of greater than 0.8. (Csáki and Vándor, 2025a)

In their companion paper focused on cost-performance analysis, (Csáki and Vándor, 2025b) compared closed and open-weight LLMs within the same RAG framework. They confirmed that for the initial OCR and document processing phase, the higher cost of closed models like GPT-4o was justified by their 100% accuracy and robustness. However, for the final question-answering task, open-weight models like Qwen2-72B provided a highly competitive, cost-effective alternative, achieving 95.65% accuracy on the same test set. Their work underscores that a performant Hungarian RAG system can be built by leveraging closed models for high-stakes preprocessing while utilizing open models for generation to optimize long-term operational costs.

3 Data and Preprocessing

The primary knowledge source for the RAG system is a collection of 165 historical books related to the Hungarian Academy of Sciences. The full corpus contains approximately 13.7 million tokens, measured using the tokenizer of `hut5-base`[2] (Madarász et al., 2025). Because the documents were provided as PDFs with noisy, inconsistent OCR layers, substantial preprocessing was required before

[1] `https://github.com/isaacus-dev/semchunk`

[2] `https://huggingface.co/GaborMadarasz/hut5-base`

retrieval experiments could be conducted. Due to the copyrighted nature of this data, the dataset and the generated questions can not be made publicly available, but some examples of the data will be shown in appendix A.

3.1 OCR Correction

The OCR text exhibited frequent orthographic errors, irregular spacing, and broken word forms, all of which significantly degraded baseline retrieval quality. To address this, an AWQ-quantized Llama 3.3 70B model[3] (Touvron et al., 2023) was used to normalize and correct the text on a page-level basis. This correction pass produced a measurable improvement in linguistic quality: perplexity under `NYTK/PULI-GPT-2`[4] (Győző et al., 2023) decreased from 41.51 to 26.85 (approximately 35%). Page-number artifacts introduced by imperfect OCR were also removed using pattern-based detection; for documents that could not be segmented consistently, a fallback static token-based splitting strategy was applied.

3.2 Initial Evaluation Resources

To determine appropriate retrieval techniques for Hungarian, the MILQA dataset (Novák et al., 2023)—a manually annotated Hungarian extractive question–answering benchmark—was used as an initial evaluation resource. MILQA provided a high-quality testbed for comparing sparse, dense, hybrid retrieval models as well as query rewriting methods and rerankers prior to evaluating performance on the Academy corpus.

3.3 Synthetic Benchmark Construction

Because no supervised QA dataset exists for the Academy materials, a synthetic benchmark was constructed to rigorously evaluate retrieval performance on the target domain. The pipeline operates on semantically coherent text chunks generated using the `semchunk` recursive chunker (Isaacus, 2025), with 50-token overlaps. For each chunk, a Hungarian T5-based question generation model—fine-tuned on MILQA—produced candidate questions. These were answered by an extractive QA model[5] operating on the same chunk. Unanswerable or low-quality questions were removed through a combination of linguistic validity rules and a cross-encoder reranker (`bge-reranker-v2-m3`) (Chen et al., 2024a).

Across the 165 books, the pipeline generated roughly 70,000 question–answer pairs, of which 15,800 high-confidence examples were retained. This synthetic dataset was subsequently used to compare retrieval pipelines and evaluate the impact of different chunking strategies.

[3] `https://huggingface.co/casperhansen/llama-3.3-70b-instruct-awq`
[4] `https://huggingface.co/NYTK/PULI-GPT-2`
[5] `https://huggingface.co/ZTamas/xlm-roberta-large-squad2-qa-milqa-impossible`

3.4 Chunking Strategies

Chunking plays a pivotal role in retrieval performance due to the long, narrative structure typical of the documents in the corpus. Two approaches were evaluated:

1. Recursive token-aware chunking using `semchunk`.
2. Semantic chunking based on embedding-driven breakpoint detection (McCormick, 2024).

After selecting the best retrieval and query rewriting configurations using MILQA, these chunking methods were compared on the synthetic benchmark. Recursive chunking with a maximum size of 512 tokens yielded the highest NDCG@10 scores after reranking, and was therefore selected for the final system.

4 System Design

This section describes the retrieval–augmented generation (RAG) pipeline used in this work. Instead of detailing individual software services or deployment choices, we focus on the core sequence of transformations that a user query undergoes: conversational rewriting, hybrid retrieval, reranking, and answer generation. The emphasis is on the algorithmic flow that produces grounded, contextually relevant responses in Hungarian.

4.1 Pipeline Overview

When a user submits a query, the system processes it together with its conversation history. An instruction-tuned LLM, `gpt-4o-mini` (OpenAI, 2024) is first used to determine whether the latest user message is a follow-up to a prior question. If so, the system generates a history-aware reformulation of the query, producing a standalone version that fully expresses the user's intent.

Next, the rewritten (or original) query is transformed into two parallel representations for hybrid retrieval. For lexical retrieval, we apply the Query2Doc method, prompting an LLM to generate a pseudo-document describing the information need. This expanded query is used for BM25 search. For dense retrieval, the query is encoded using the `multilingual-e5-large` encoder (Wang et al., 2024) and used to query the vector index.

Each retrieval branch returns the top 100 candidate passages. These results are merged using reciprocal rank fusion (RRF) with $k = 60$, producing a unified ranking that combines the strengths of sparse and dense search. A cross-encoder reranker (`bge-reranker-v2-m3`) (Chen et al., 2024a) then scores each query–passage pair, and the top 10 passages are selected for grounded answer generation.

In the final stage, another LLM (originally also `gpt-4o-mini`) is prompted, conditioned on both the retrieved passages and a short summary of the history of the Hungarian Academy of Sciences. The model is instructed to answer using only the provided context. Responses are streamed to reduce latency.

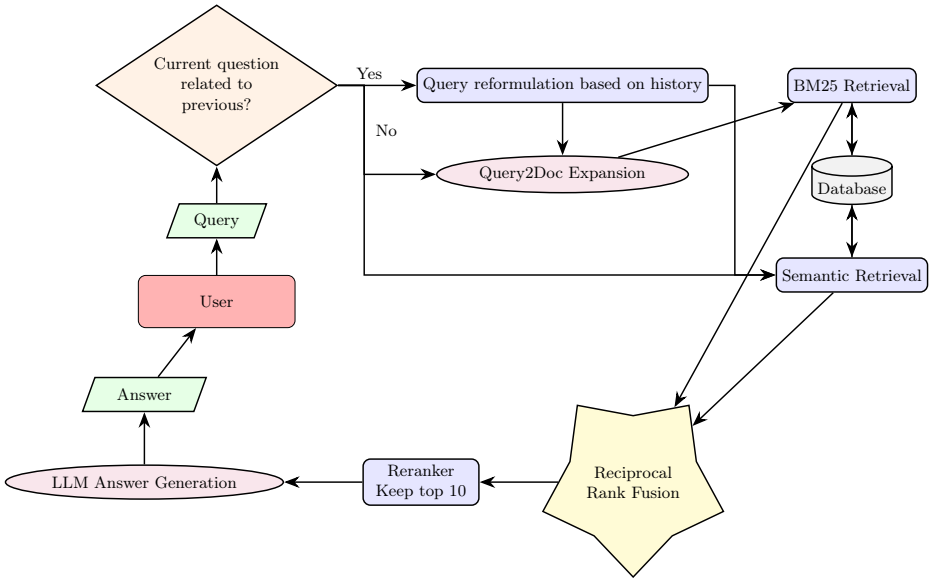


Fig. 1: Flowchart of the RAG system

5 Experiments

We evaluate our system in three stages. First, we compare sparse, dense, and hybrid retrieval methods on the MILQA benchmark, including the effect of query rewriting and reranking. Second, we evaluate chunking strategies on the Hungarian Academy of Sciences corpus using a BEIR-style protocol (Thakur et al., 2021). Finally, we assess end-to-end RAG answer quality using LLM-as-judge metrics. The code for all experiments can be found on GitHub[6].

5.1 Retrieval Evaluation

We compare BM25, several multilingual dense encoders, and their hybrid combinations. The evaluation uses MILQA (train+test, excluding unanswerable items), measuring Recall@k, NDCG@10, MRR@100, and Precision@k.

Sparse vs. Dense Retrieval. Among dense encoders, `multilingual-e5-large` achieves the best retrieval performance, surpassing `BGE-M3`, `OpenAI-TE3L`, and smaller multilingual encoders. BM25 performs strongly for Hungarian but remains below the best dense models.

[6] `https://github.com/ELTE-DH/mta200_rag`

Table 1. Retrieval performance of sparse and dense models (no query rewriting).

Model	R@1	R@10	NDCG@10	MRR@100
BM25	0.640	0.913	0.776	0.736
BGE-M3[7]	0.683	0.957	0.826	0.785
E5-Large[8]	**0.715**	**0.969**	**0.857**	**0.785**
OpenAI-TE3L[9]	0.629	0.943	0.791	0.744

Query Rewriting. We test Query2Doc and HyDE. Query2Doc substantially improves BM25, but rewriting tends to degrade dense retrieval quality.

Table 2. Effect of query rewriting on retrieval.

Base Model	Rewriting	R@1	R@10	NDCG@10	MRR@100
BM25	Query2Doc	**0.694**	**0.950**	**0.825**	**0.786**
E5-Large	HyDE	0.670	0.965	0.670	0.780
E5-Large	Query2Doc	0.697	0.965	0.670	0.799

Hybrid Retrieval. Hybrid retrieval uses Reciprocal Rank Fusion (RRF) with BM25 and E5-Large. Applying Query2Doc only to BM25 yields the strongest overall performance.

Table 3. Hybrid retrieval performance (RRF with BM25 + E5-Large).

Hybrid Setting	R@1	R@10	NDCG@10	MRR@100
BM25(Q2D) + E5-Large	**0.744**	**0.981**	0.865	**0.829**
BM25(Q2D) + E5-Large(Q2D)	0.735	0.979	**0.873**	0.825

Dense retrieval with `multilingual-e5-large` outperforms other encoders, BM25 benefits substantially from Query2Doc, and the best overall performance comes from a hybrid BM25(Q2D)+E5-Large setup with RRF.

Reranking. We evaluate three multilingual cross-encoders using the best hybrid retrieval configuration.

[7] `https://huggingface.co/BAAI/bge-m3`
[8] `https://huggingface.co/intfloat/multilingual-e5-large`
[9] `https://ai.azure.com/catalog/models/text-embedding-3-large`

Table 4. Reranker performance on hybrid retrieval candidates.

Reranker	R@1	R@10	NDCG@10	MRR@100
GTE[10]	0.816	0.992	0.912	0.887
Jina v2[11]	0.845	0.994	0.927	0.905
BGE-reranker-v2-m3[12]	**0.871**	**0.996**	**0.942**	**0.924**

Among cross-encoders, `bge-reranker-v2-m3` provides the strongest reordering of hybrid candidates and is selected for downstream use.

5.2 Chunking Evaluation

We evaluate chunking on the Hungarian Academy of Sciences corpus using the best-performing hybrid retrieval and the best reranker. Two families of chunkers are compared:

- **Recursive token splitter** with maximum chunk sizes of 512, 1024, and 2048 tokens.
- **Modified semantic chunker** with flexible ranges (128–512, 256–1024, 512–2048 tokens).

Recursive 512-token chunks perform best prior to reranking. After reranking, 512- and 1024-token recursive chunks both perform strongly, with 512 tokens retaining a slight advantage.

Table 5. Chunking strategies (post-reranking) on the Hungarian Academy corpus.

Chunker	NDCG@10	R@10	P@1	MRR@100
Recursive 512 tokens	**0.857**	**0.944**	0.754	0.833
Recursive 1024 tokens	0.848	0.929	**0.764**	**0.834**
Semantic 256–1024 tokens	0.835	0.918	0.741	0.814

5.3 Generation Evaluation

We evaluate end-to-end RAG answer quality by measuring the faithfulness, response relevancy with RAGAS (Es et al., 2025). For this purpose, we constructed an evaluation set of 164 refined questions, supplemented by 60 human-annotated QA pairs.

[10] `https://huggingface.co/Alibaba-NLP/gte-multilingual-reranker-base`
[11] `https://huggingface.co/jinaai/jina-reranker-v2-base-multilingual`
[12] `https://huggingface.co/BAAI/bge-reranker-v2-m3`

– **Synthetic QA set:** Faithfulness = 0.754, Relevancy = 0.782
– **Human-annotated set:** Faithfulness = 0.739, Relevancy = 0.699

These results indicate that the system maintains good factual grounding and produces contextually appropriate answers across both synthetic and human-authored test sets, with some slight degradation on the human-authored set.

5.4 Error Analysis

To provide qualitative insight into the faithfulness scores reported in the previous section, we conducted a manual analysis of incorrect responses generated by the system on the human-annotated set.

Interpretation of Results. While faithfulness and relevance scores of approximately 70% are below state-of-the-art results for high-resource languages (which often exceed 90%) (Es et al., 2025), they represent a strong baseline for the low-resource, domain-specific setting of historical Hungarian text. The gap in performance is less attributable to residual OCR errors—which were mitigated to a certain extent by the Llama-3 correction pass—and more to the linguistic complexity of the corpus and the difficulty of the synthesis tasks required.
We identified five primary error categories that persist within the pipeline:

Temporal and Chronological Reasoning. The model occasionally fails to cross-reference dates within retrieved contexts to validate the feasibility of events. It tends to treat entities as contemporary if they appear in the same context window, ignoring specific biographical timelines. For example, the system failed to reject a hypothetical meeting between József Eötvös and János Arany in 1872, ignoring retrieved evidence of Eötvös's death in 1871.

Logical and Role Confusion. We observed errors where the model misinterpreted the modality of a sentence or conflated hierarchical roles. In one instance, the model incorrectly classified John von Neumann as an elected member based on a text stating that a member *attempted* to elect him. This suggests limitations in the multilingual model's ability to resolve subtle Hungarian morphological cues regarding modality.

Hallucination of Facts and Relationships. When explicit links were missing from the retrieved chunks, the model showed a tendency to fabricate connections to satisfy the user's prompt. This was most prevalent in queries regarding kinship, where the model asserted sibling relationships between unrelated historical figures solely based on shared surnames (e.g., the "Balogh" family).

Synthesis Incompleteness. While the system performs well on targeted fact extraction, it struggles with synthesis tasks that require aggregating data scattered across multiple segments. This resulted in incomplete outputs when the user requested exhaustive lists, such as enumerating all Academy Presidents over a specific multi-decade period.

Context Incompleteness. About 30% of the error cases caught by the LLM-as-judge metrics were instances when the answer to the test question was not in the model's context at all, and the model responded with a fallback "I don't know" answer, which led to a faithfulness and response relevancy score of 0.0.

5.5 Final System Configuration Derived from Experiments

Based on the experimental findings, the following configuration is used in the final RAG pipeline:

- **Retriever:** Hybrid search combining BM25 with Query2Doc and `E5-Large`, fused with RRF.
- **Reranker:** `bge-reranker-v2-m3`.
- **Chunking:** Recursive 512-token chunks with 10% overlap.

This combination provides the strongest balance of retrieval quality, ranking accuracy, and generation consistency for Hungarian historical texts.

6 Limitations and Future Work

Although the proposed system provides a strong baseline for retrieval-augmented generation in Hungarian, several limitations remain. These arise primarily from the scarcity of high-quality Hungarian data, limited computational resources, and the restricted scope of evaluated models.

Model Adaptation. The system relies on general-purpose multilingual LLMs without any domain- or task-specific fine-tuning. Training a Hungarian-focused RAG model—either through continued pretraining or supervised fine-tuning—could significantly improve factual grounding, fluency, and robustness for historical or scientific domains.

Question Generation Quality. The synthetic benchmark enables scalable evaluation but is constrained by the quality of the underlying question generation model. Pretraining or fine-tuning a Hungarian T5-style model on large monolingual corpora, or using a larger model with good Hungarian capabilities, could improve question quality and consistency, resulting in a more reliable retrieval evaluation dataset.

Evaluation Scope. Only a limited number of multilingual and open-source models were evaluated due to cost and compute constraints. A broader comparison—including newly emerging Hungarian LLMs—would provide a clearer understanding of language-specific strengths and weaknesses in retrieval-augmented settings.

Conversational Capabilities. The current system focuses on single-turn question answering. Supporting multi-turn retrieval with explicit contextual tracking and reformulation remains an open challenge. Evaluating such systems using multi-turn RAG benchmarks or conversational faithfulness metrics would be a valuable next step.

7 Conclusion

This work presents a comprehensive empirical study of retrieval, reranking, and chunking strategies for building an effective Hungarian RAG pipeline. Hybrid retrieval—combining BM25 with Query2Doc and multilingual dense embeddings—provides the strongest overall performance, while cross-encoder reranking further improves ranking quality. Smaller recursive chunks (512 tokens) offer the best retrieval effectiveness and computational efficiency.

End-to-end evaluation on both synthetic and human-authored questions shows that the resulting system generates answers that are contextually grounded and reasonably faithful, even in a low-resource linguistic setting. These findings highlight the practicality of combining robust retrieval methods with modern multilingual LLMs for high-quality information access in Hungarian.

Acknowledgements

The authors acknowledge the support of the National Laboratory for Digital Heritage. Project no. 2022-2.1.1-NL-2022-00009 has been implemented with the support provided by the Ministry of Culture and Innovation of Hungary from the National Research, Development and Innovation Fund, financed under the 2022-2.1.1-NL funding scheme.

Bibliography

Attar, R., Fraenkel, A.S.: Local Feedback in Full-Text Retrieval Systems. J. ACM 24(3), 397–417 (Jul 1977), https://doi.org/10.1145/322017.322021, place: New York, NY, USA Publisher: Association for Computing Machinery

Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., Liu, Z.: BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation (Jun 2024a), http://arxiv.org/abs/2402.03216, arXiv:2402.03216 [cs]

Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., Liu, Z.: M3-Embedding: Multi-Linguality, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. In: Ku, L.W., Martins, A., Srikumar, V. (eds.) Findings of the Association for Computational Linguistics: ACL 2024. pp. 2318–2335. Association for Computational Linguistics, Bangkok, Thailand (Aug 2024b), https://aclanthology.org/2024.findings-acl.137/

Cormack, G.V., Clarke, C.L.A., Buettcher, S.: Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. pp. 758–759. ACM, Boston MA USA (Jul 2009), https://dl.acm.org/doi/10.1145/1571941.1572114

Csaki, Z., Pawakapan, P., Thakker, U., Xu, Q.: Efficiently Adapting Pretrained Language Models To New Languages (Dec 2023), http://arxiv.org/abs/2311.05741, arXiv:2311.05741 [cs]

Csáki, C., Vándor, P.: RAG módszerek implementálásának kihívásai egy célorientált chatbot rendszer kontextusában. In: XXI. Magyar Számítógépes Nyelvészeti Konferencia. pp. 97–110. Szeged, Hungary (Feb 2025a)

Csáki, C., Vándor, P.: Zárt és nyílt súlyú LLM-ek teljesítményértékelése RAG-alapú felsőoktatási chatbot alkalmazás példáján. In: XXI. Magyar Számítógépes Nyelvészeti Konferencia. pp. 233–246. Szeged, Hungary (Feb 2025b)

Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (May 2019), http://arxiv.org/abs/1810.04805, arXiv:1810.04805 [cs]

Es, S., James, J., Espinosa-Anke, L., Schockaert, S.: Ragas: Automated Evaluation of Retrieval Augmented Generation (Apr 2025), http://arxiv.org/abs/2309.15217, arXiv:2309.15217 [cs]

Gao, L., Ma, X., Lin, J., Callan, J.: Precise Zero-Shot Dense Retrieval without Relevance Labels (Dec 2022), http://arxiv.org/abs/2212.10496, arXiv:2212.10496 [cs]

Győző, Y.Z., Réka, D., Gergő, F., Enikő, H., Ádám, K., János, L.L., Noémi, L.N., Noémi, V., Tamás, V.: Jönnek a nagyok! BERT-Large, GPT-2 és GPT-3 nyelvmodellek magyar nyelvre. In: XIX. Magyar Számítógépes Nyelvészeti Konferencia. pp. 247–262. Szeged, Hungary (Jan 2023)

Humeau, S., Shuster, K., Lachaux, M.A., Weston, J.: Poly-encoders: Transformer Architectures and Pre-training Strategies for Fast and Accurate Multi-sentence Scoring (Mar 2020), http://arxiv.org/abs/1905.01969, arXiv:1905.01969 [cs]

Isaacus: isaacus-dev/semchunk (Apr 2025), https://github.com/isaacus-dev/semchunk, original-date: 2023-11-05T03:47:20Z

Khattab, O., Zaharia, M.: ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 39–48. SIGIR '20, Association for Computing Machinery, New York, NY, USA (2020), https://doi.org/10.1145/3397271.3401075, event-place: Virtual Event, China

Lavrenko, V., Croft, W.B.: Relevance based language models. In: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 120–127. SIGIR '01, Association for Computing Machinery, New York, NY, USA (Sep 2001), https://doi.org/10.1145/383952.383972

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., Riedel, S., Kiela, D.: Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks (Apr 2021), http://arxiv.org/abs/2005.11401, arXiv:2005.11401 [cs]

Madarász, G., Holl, A., Ligeti-Nagy, N., Yang, Z.G., Váradi, T.: Text cleaning with transformer language models for Hungarian. In: XXI. Magyar Számítógépes Nyelvészeti Konferencia. pp. 123–135. Szegedi Tudományegyetem TTIK, Informatikai Intézet, online kiadás (2025), https://real.mtak.hu/216688/, num Pages: 13

Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York, illustrated edition edn. (Jul 2008)

McCormick, Z.: Solving the out-of-context chunk problem for RAG (Jul 2024), https://d-star.ai/solving-the-out-of-context-chunk-problem-for-rag/

Miller, G.A.: WordNet: a lexical database for English. Commun. ACM 38(11), 39–41 (Nov 1995), https://doi.org/10.1145/219717.219748, place: New York, NY, USA Publisher: Association for Computing Machinery

Novák, A., Novák, B., Zombori, T., Szabó, G., Szántó, Z., Farkas, R.: A Question Answering Benchmark Database for Hungarian. In: Prange, J., Friedrich, A. (eds.) Proceedings of the 17th Linguistic Annotation Workshop (LAW-XVII). pp. 188–198. Association for Computational Linguistics, Toronto, Canada (Jul 2023), https://aclanthology.org/2023.law-1.19/

OpenAI: GPT-4 Technical Report (Mar 2024), http://arxiv.org/abs/2303.08774, arXiv:2303.08774 [cs]

Orosz, G., Szabó, G., Berkecz, P., Szántó, Z., Farkas, R.: Advancing Hungarian Text Processing with HuSpaCy: Efficient and Accurate NLP Pipelines. In: Ekštein, K., Pártl, F., Konopík, M. (eds.) Text, Speech, and Dialogue. pp. 58–69. Springer Nature Switzerland, Cham (2023)

Qiao, Y., Xiong, C., Liu, Z., Liu, Z.: Understanding the Behaviors of BERT in Ranking (2019), https://arxiv.org/abs/1904.07531, _eprint: 1904.07531

Robertson, S., Walker, S., Jones, S., Hancock-Beaulieu, M., Gatford, M.: Okapi at TREC-3. In: Text Retrieval Conference (1994), https://www.semanticscholar.org/paper/Okapi-at-TREC-3-Robertson-Walker/fa8ffa9f8d8367b987845ae88f2adcac2689f110

Savoy, J.: Searching strategies for the Hungarian language. Information Processing & Management 44(1), 310–324 (Jan 2008), https://www.sciencedirect.com/science/article/pii/S0306457307000520

Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., Gurevych, I.: BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models. In: Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2) (2021), https://openreview.net/forum?id=wCu6T5xFjeJ

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., Lample, G.: LLaMA: Open and Efficient Foundation Language

Models (Feb 2023), http://arxiv.org/abs/2302.13971, arXiv:2302.13971 [cs]

Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R., Wei, F.: Multilingual E5 Text Embeddings: A Technical Report (Feb 2024), http://arxiv.org/abs/2402.05672, arXiv:2402.05672 [cs]

Wang, L., Yang, N., Wei, F.: Query2doc: Query Expansion with Large Language Models (Oct 2023), http://arxiv.org/abs/2303.07678, arXiv:2303.07678 [cs]

Xu, J., Croft, W.B.: Query expansion using local and global document analysis. In: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 4–11. SIGIR '96, Association for Computing Machinery, New York, NY, USA (1996), https://doi.org/10.1145/243199.243202, event-place: Zurich, Switzerland

Zhang, X., Thakur, N., Ogundepo, O., Kamalloo, E., Alfonso-Hermelo, D., Li, X., Liu, Q., Rezagholizadeh, M., Lin, J.: **MIRACL** : A Multilingual Retrieval Dataset Covering 18 Diverse Languages. Transactions of the Association for Computational Linguistics 11, 1114–1131 (Sep 2023), https://direct.mit.edu/tacl/article/doi/10.1162/tacl_a_00595/117438/MIRACL-A-Multilingual-Retrieval-Dataset-Covering

A Qualitative samples of OCR repair and question generation

Original text (pre-fixing)

Értékes munka, melynek megírására a M. T. Akadémia még 1872-ben szólította fel először Than Károlyt : A kísér leti kémia elemei című mű, mely 1896-tól 1906-ig jelent meg. E munka méltatásához nem kell kommentár. Mindannyian még diák korunkban belőle tanultunk és ma is, ha hirtelen valamilyen adatra van szükségünk, gyakran fordulunk hozzá. Than Károly egyik legkiválóbb tanítványa a ma élő vegyészek nagy tiszteletben álló nesztora : Ilosvay Lajos.

Fixed text (post-fixing)

Értékes munka, melynek megírására a M. T. Akadémia még 1872-ben szólította fel először Than Károlyt: "A kísérleti kémia elemei" című mű, mely 1896-tól 1906-ig jelent meg. E munka méltatásához nem kell kommentár. Mindannyian még diák korunkban belőle tanultunk, és ma is, ha hirtelen valamilyen adatra van szükségünk, gyakran fordulunk hozzá. Than Károly egyik legkiválóbb tanítványa a ma élő vegyészek nagy tiszteletben álló nesztora: Ilosvay Lajos.

Human-created question

Melyik mű elkészítésére kapott felkérést Than Károly az Akadémiától 1872-ben?

T5-generated questions (unfiltered)

1. Mikor jelent meg a ,,A kísérletű kémiai elemei'' című műve?
3. Milyen tudományos tárgyakat írt a Magyar Tudományis Akadémia 1906-ban?
4. Ki volt Ilosvay Lajos?
5. Mikor jelent meg az első magyar kémiai kémiai megírása?

Fig. 2: Example of orthographic normalization with aligned human-annotated and model-generated questions.

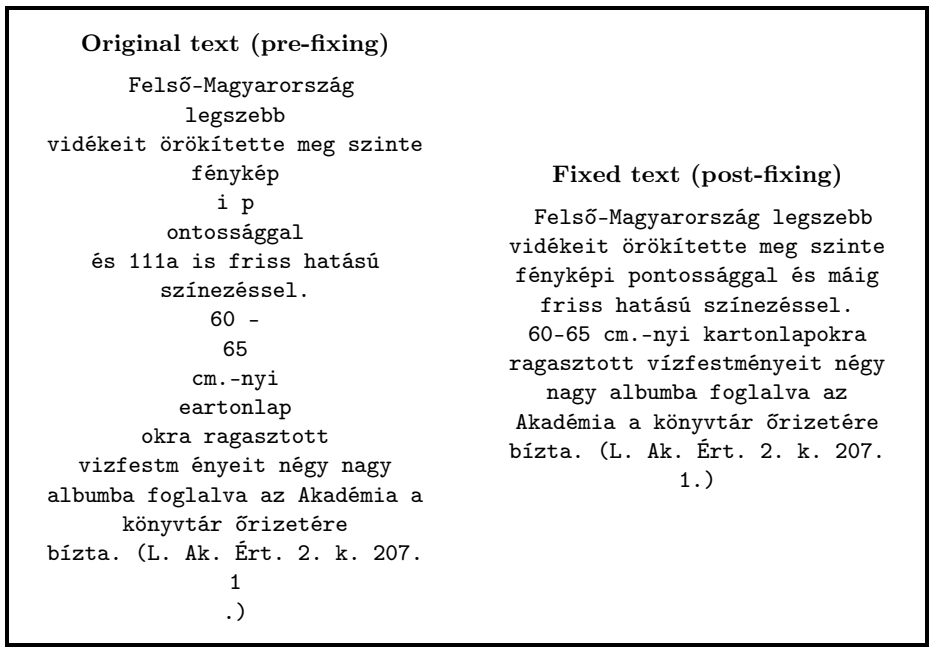


Fig. 3: Qualitative comparison of a part of the dataset, before and after the fixing of the orthography.

B Prompts used for the Large Language Model-based techniques

System Prompt

Improve the clarity and grammar of the following question and generate an abstractive answer based on the provided context.
Keep all the JSON fields in Hungarian, except for the keys.
If the question can't be answered using the provided context, simply leave both fields of the JSON object empty.
Return a JSON object following the provided JSON schema.

User Input (Document Title + Question + Context)

```
Document title: {TITLE}
Question: {QUESTION}
Context: {EXTRACTIVE-ANSWER-WITH-ADDITIONAL-SURROUNDING-TOKENS}
```

Fig. 4: Prompt for question refinement and abstractive answer generation. The system defines the task, and the user provides the document title, question, and context.

```
You're a chatbot working in an exhibition of the Hungarian Academy of
↪  Sciences (MTA).You will hold conversations only related to the
↪  exhibition and history of the academy. The history of the academy and
↪  additional exhibition info will be given to you in the CONTEXT below.
↪  Respond in Hungarian only, and don't mention the context given to you
↪  in the system prompt to the user, respond naturally. Answer the
↪  user's questions based on the CONTEXT. Make sure to ONLY use the
↪  knowledge given in the CONTEXT, if the CONTEXT doesn't contain
↪  information relevant to the question, just answer with 'Nem tudom.'.
CONTEXT:
{EXHIBITION_GENERAL_INFORMATION_DOCUMENT}
{RETRIEVED_DOCUMENTS}
```

Fig. 5: System prompt for answer generation in the final step of the pipeline.

```
You are a Hungarian document orthography repair model. You will be given
↪ documents with irregular orthography, typically caused by extraction
↪ from a PDF's text layer. Your task is to rewrite the text to standard
↪ Hungarian orthography. Do not process or modify text in any language
↪ other than Hungarian. Return only the corrected Hungarian text,
↪ preserving paragraphs separated by empty lines. Do not include
↪ comments, explanations, or additional text in your output.

Examples:
Input: 'A köv etke z ő sza vak hibá sak: pld. szókö z ek.'
Output: 'A következő szavak hibásak: például szóközök.'

Input: 'E z e g y pró b a szö veg.'
Output: 'Ez egy próbaszöveg.'
Input: Gleichzeitig ordnete die Akademie die Ordnung und Katalogisierung,
↪ sowie die Ausgabe eines gedruckten Verzeichnisses der Sammlung an.
↪ vallás-
erkölcsi, t árs ad almi, kulturális és politi kai szereplé sérő l.
Output: Gleichzeitig ordnete die Akademie die Ordnung und
↪ Katalogisierung, sowie die Ausgabe eines gedruckten Verzeichnisses
↪ der Sammlung an.Valláserkölcsi, társ adalmi, kulturális és politikai
↪ szerepléséről.
```

Fig. 6: System prompt for the OCR document orthography fixing.

```
Generate a passage in Hungarian as an answer for the question below:
{PROMPT}
```

Fig. 7: Prompt for HyDE and Query2Doc.

System Prompt

Determine if the following two questions are related. Answer with 'yes' or 'no' only.

User Prompt

Question 1: {QUESTION1}
Question 2: {QUESTION2}
Are these questions related?

Fig. 8: Prompt structure used to assess question relatedness. The system sets up the task, while the user provides two questions to be compared.

System Prompt

Given a chat history and the latest user question
which might reference context in the chat history, formulate a standalone question
which can be understood without the chat history.
Do **NOT** answer the question — just reformulate it if needed, otherwise return it as is.

User Input (Chat History + Question)

```
{chat_history}
```

Fig. 9: Prompt for generating a standalone version of the user's latest question based on chat history. The system defines the task, and the user provides the chat context.

System Prompt

Improve the clarity and grammar of the following question and generate an abstractive answer based on the provided context.
Keep all the JSON fields in Hungarian, except for the keys.
If the question can't be answered using the provided context, simply leave both fields of the JSON object empty.
Return a JSON object following the provided JSON schema.

User Input (Document Title + Question + Context)

```
Document Title: {TITLE}
Question: {QUESTION}
Context: {CONTEXT}
```

Example JSON Output

```
{
"fixed_question": {FIXED_QUESTION},
"abs_answer": {ABSTRACTIVE_ANSWER}
}
```

Fig. 10: Prompt for improving question clarity and generating an abstractive answer based on a given document context.

Párbeszéd

200 óra lett, maradhat? A BEA-Dialogue+ korpusz

Gedeon Máté[1,2], Barta Piroska Zsófia[1,2], Mihajlik Péter[1,3], Mády Katalin[3]

[1] Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Mesterséges Intelligencia Tanszék
[2] Speechtex Kft.
[3] ELTE Nyelvtudományi Kutatóközpont
gedeonm@edu.bme.hu, mihajlik@tmit.bme.hu

Kivonat A BEA-Dialogue korpusz, amely párbeszédes beszédfelismerési feladatokra készült, szigorú beszélőfüggetlen felosztása révén ugyan objektív értékelést tesz lehetővé, de jelentősen korlátozza a rendelkezésre álló tanítóadat mennyiségét. Munkánk célja feltárni, hogy a függetlenségi feltételek részleges enyhítése – a beszélgetőpartnerek és kísérletvezetők személyeinek átfedését megengedve a halmazok között – milyen mértékben növelheti a felhasználható adatmennyiséget és ez hogyan befolyásolja a párbeszédes leiratozó modellek teljesítményét. Ennek eredménye a BEA-Dialogue+ korpusz, amely a korábbi 85 órához képest lényegesen több, összesen 200 órányi lejegyzett természetes beszélgetést tartalmaz, miközben megőrzi a főbeszélők teljes szeparációját. Az összehasonlító kísérletek során több Whisper- és Fast Conformer-alapú modellt kiértékeltünk, utóbbiakat finomhangolva is Serialized Output Training (SOT) módszerrel. Eredményeink azt mutatják, hogy a finomhangolás nélküli modellek teljesítménye a korpusz komplexitásának növekedésével romlik, míg a SOT-alapú finomhangolás négy metrikában (WER, CER, cpWER, cpCER) is jelentősen javítja a teljesítményt. A BEA-Dialogue+ méretéből adódóan értékes új erőforrás a magyar dialógusleiratozó rendszerek fejlesztéséhez és lehetőséget ad a magyar rendszerek egységes kiértékelésére.
Kulcsszavak: dialóguskorpusz, gépi beszédfelismerés, több-beszélős ASR

1. Bevezetés

A magyar nyelvű gépi beszédfelismerés (ASR, Automatic Speech Recognition) – követve a világtrendeket – jelentős fejlődésen ment keresztül az elmúlt években, mind az elérhető adatmennyiség, mind pedig a felhasznált modellek kifinomultsága tekintetében (Mihajlik és mtsai, 2024; Dobsinszki és mtsai, 2026). A folyamatosan bővülő, kutatási célra publikált spontán beszédet tartalmazó adatbázisok között megemlítendő a BEA-Large és annak dialógusalapú változata, a BEA-Dialogue (Gedeon és mtsai, 2025). Ezek az adatbázisok fontos eszközöket jelentenek a magyar ASR-rendszerek tanításához és kiértékeléséhez, különösen a spontán beszéd, és természetes párbeszédek leiratozásában.

A BEA-Dialogue korpusz a BEA-Large felvételeiből lett származtatva, a célja pedig az, hogy párbeszédszerű egységekbe szervezett, természetes kommunikációs helyzeteket biztosítson a beszédfelismerő rendszerek számára. A korpusz létrehozásakor azonban alapvető kritérium volt a teljes beszélőfüggetlenség megőrzése a tanító-, validációs- és teszthalmazok között, hogy a kiértékelés valóban objektív képet adjon a modellek általánosító képességéről. Ez a szigorú feltétel ugyanakkor azzal a következménnyel járt, hogy a BEA-Dialogue effektív adatmennyisége jelentősen csökkent: a publikált változat mindössze 85 órányi beszélgetést tartalmaz, szemben a BEA-Large eredeti 255 órás terjedelmével.

Ez a csökkenés különösen hátrányos lehet a modern, neurális hálózatokon alapuló beszédfelismerő rendszerek számára, amelyek teljesítménye erősen függ a rendelkezésre álló adatmennyiségtől és annak sokféleségétől (Roger és mtsai, 2020). Ebből kiindulva, jelen tanulmányban azt vizsgáljuk, hogy a szétosztási feltételek enyhítésével – azaz a beszélőfüggetlenség részleges feloldásával – mennyivel nagyobb adathalmaz állítható elő, és hogy az így létrejövő kiterjesztett korpuszon tanított modellek teljesítménye hogyan viszonyul a szigorúan független bontású változatokon végzett kísérletekhez.

A függetlenség feloldása viszont automatikusan adatszivárgási (data leakage) problémákat jelent. Noha egy konkrét felvétel természetesen nem kerülhet egynél több halmazba (mint például tanító, validáló vagy kiértékelő halmazba) és a főbeszélők átfedését is elkerültük a halmazok között (főbeszélőnek az elsődleges adatközlőt nevezzük), de a kísérletvezetők és beszélgetőpartnerek több halmazban is előfordulnak. Így az eredmények általánosíthatóságát óvatosan kell kezelni (Kapoor és Narayanan, 2023), viszont megjegyezzük, hogy a beszélők átfedésének megengedése (nem kiszűrése) bevett gyakorlat olyan adatok esetén, ahol ennek kiküszöbölése szinte lehetetlen (pl. broadcast news) (Arısoy és mtsai, 2007).

A kísérletek során használt BEA-Dialogue+ korpusz kutatási célra elérhető[1].

2. Adatok

A BEA-Dialogue korpusz kifejezetten párbeszédes beszédfeldolgozási kutatások céljára készült. A korpusz alapját a BEA adatbázis (Neuberger és mtsai, 2014) 242 olyan beszélőjének Mády és mtsai (2024) szerint lejegyzett felvételei adják, akik korábban nem szerepeltek a BEA-Base adatbázisban (Mihajlik és mtsai, 2022).

Az új adatállomány létrehozása során a felvételekben szereplő megszólalásokat időbélyegeikkel és beszélőazonosítóikkal együtt (SPK – főbeszélő, EXP – kísérletvezető, DP – beszélgetőpartner) gyűjtötték ki. A párbeszédeket a felvételekben található szünetek mentén vágták szegmensekre, így jöttek létre a természetes határokkal rendelkező, koherens dialógusrészletek. Ezeket a rövidebb egységeket később nagyobb, körülbelül 30 másodperces szegmensekké egyesítették, amelyek jól használhatóak automatikus beszédfeldolgozási és nyelvtechnológiai feladatokhoz.

[1] https://phon.nytud.hu/bea/

A korpuszban a főbeszélőkön túl több női és férfi kísérletvezető, illetve beszélgetőpartner is szerepel, ami változatos beszédhelyzeteket és interakciós mintákat biztosít. A BEA-Dialogue esetén a tanító (train), validációs (dev) és kiértékelési (eval) részhalmazok minden beszélőben diszjunktak. Ennek érdekében a felosztás három kísérletvezető személyhez kötődően történt, úgy, hogy az egyes halmazok között sem a főbeszélők, sem a partnerek ne ismétlődjenek. Ennek következtében például a *discourse* modul – az egyedüli modul amelyekben a beszélgetőpartnerek is szerepelnek – néhány beszélő esetében nem került bele az adathalmazba.

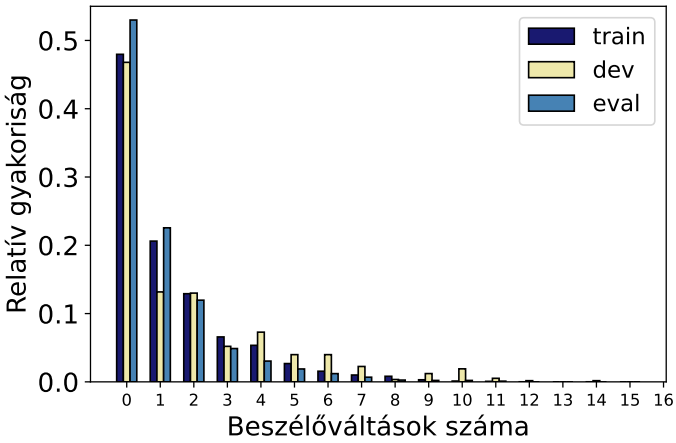
Az így létrejött BEA-Dialogue a BEA-adatbázis eddigi legnagyobb párbeszédes részgyűjteménye, amely biztosítja a beszélőfüggetlen szétosztást. Viszont ez a függetlenség komolyan korlátozta a létrehozható korpusz méretét.

A BEA-Dialogue+ esetén a feltételt úgy enyhítettük, hogy csak a főbeszélő esetén követeltük meg a függetlenséget. Az eredményt a 1. táblázat foglalja össze. Mint látható, így 85 óráról 200 órára növekedett az adatmennyiség. A *dev* és *eval* halmazok méretösszege hasonló mindkét verziónál, de a BEA-Dialogue+ esetén kiegyensúlyozottabb a kettő.

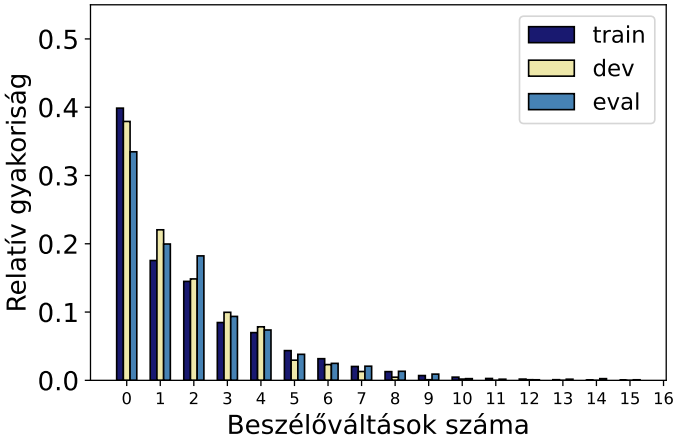
	BEA-Dialogue			**BEA-Dialogue+**		
	Train	**Dev**	**Eval**	**Train**	**Dev**	**Eval**
# Speakers [f|m]	121 | 67	3 | 6	29 | 16	179 | 126	11 | 4	15 | 2
# Segments	9,179	577	1,906	25,193	1,084	1,207
# Words	532,732	34,056	105,472	1,587,977	65,399	75,119
# Characters	3,217,617	206,740	641,628	9,402,372	387,251	445,382
# Nonlexical units	44,013	1,662	7,041	113,361	3,482	5,286
Avg. # Speakers / Segment	1.77	1.92	1.61	1.97	1.81	1.94
Avg. # Utterances / Segment	10.99	8.68	9.74	10.39	8.48	9.91
Avg. Segment Duration [s]	26.23	26.31	26.09	26.21	26.0	25.95
SPK Duration [h]	46.41	3.18	9.64	111.66	4.60	4.99
EXP Duration [h]	15.40	0.61	2.13	54.03	2.27	2.83
DP Duration [h]	1.39	0.38	0.76	16.82	0.76	0.81
Liter. Overlap Duration [h]	1.60	0.21	0.26	8.58	0.31	0.40
Total Overlap Duration [h]	3.28	0.29	0.40	14.54	0.44	0.65
Total Duration [h]	66.87	4.22	13.81	183.41	7.83	8.70

1. táblázat. Metaadatok a két BEA-Dialogue változathoz.

Az adatmennyiségen kívül érdekes változás, hogy a fél perces szegmensekre jutó beszélőváltások száma is megváltozott a feltétel enyhítésével (1., 2. ábrák). Mint látható, az új korpusz kevesebb esetben tartalmaz nulla vagy egy váltást, így a feladat várhatóan nehezedik, hiszen több esély van a rábeszélésre, ami különösen megnehezíti a leiratozást. Ezt igazolja az 1. táblázat "Total Overlap Duration" sora, ami minden, a beszélők által keltett hangátfedést összesít, de különösen a "Literal Overlap Duration" ami a szövegesen lejegyezett, ill. a jelen tanulmányban beszédfelismerési kísérletekben kiértékelt átfedő beszédrészeket összesíti.



1. ábra: Beszélőváltások számának eloszlása (szegmensenként) a BEA-Dialogue esetén.



2. ábra: Beszélőváltások számának eloszlása (szegmensenként) a BEA-Dialogue+ esetén.

3. Kísérletek

A BEA-Dialogue+ adathalmaz esetében a BEA-Dialogue eredeti cikkében (Gedeon és mtsai, 2025) bemutatott modellekkel megegyezőeket tanítottunk, amiket saját fejlesztésű modellekkel is kiegészítettünk. A tanítás során a Serialized Output Training (SOT) Kanda és mtsai (2020) módszert alkalmaztuk, amelyben a beszélőváltásokat egy `<sc>` (speaker change) token jelzi. Ezeket a tokeneket úgy illesztettük be, hogy minden beszélő megszólalásai egyben maradjanak, még akkor is, ha egy másik beszélő közbeszólt. Ezzel a modellt a beszélőhatárok felis-

merésére tanítjuk, miközben megőrizzük az egyes megszólalások nyelvi egységét. Például:

> *Szia, megkaptad a levelet?* `<sc>` *Igen, épp most olvasom.* `<sc>` *Mit gondolsz róla?* `<sc>` *Elég érdekes, szerintem megéri megbeszélni.*

A tanításhoz az adatbázisból eltávolítottuk a hezitációkat, a nevetést és minden egyéb, annotált nem-szöveges eseményt (Mády és mtsai, 2024). A modelleket kizárólag a megtisztított, verbális tartalmon tanítottuk és ezen az adaton értékeltük ki. Továbbá a hatékonyabb tanítás érdekében a tanítóhalmaz kiugróan hosszú szegmenseit nem használtuk fel a tanításhoz.

Az értékelést *WER* (Word Error Rate, szószintű hibaarány) és *CER* (Character Error Rate, karakterszintű hibaarány) metrikák alapján végeztük, illetve ezeknek a dialógusra finomított (cpWER, cpCER) változataival. Utóbbiak minimalizálják a hibákat az összes lehetséges permutáció között, ahol az egységeket a `<sc>` tokenek zárják közre. Mivel egyes szegmensek több mint tíz beszélőváltást tartalmaztak, az összes permutáció kiértékelése nem volt kivitelezhető. Ennek megoldására hibrid stratégiát alkalmaztunk: legfeljebb hét beszélőváltásig minden lehetséges permutációt kiszámítottunk, míg a nagyobb számú váltások esetében *beam search* algoritmust használtunk a közel optimális eredmények elérésére.

A finomhangolt modellek esetében továbbá beszámolunk a beszélőváltások meghatározásának pontosságának (*scAcc*) értékéről is, amely azt mutatja meg, hogy a modell az esetek hány százalékában jósolta meg helyesen a `<sc>` tokenek előfordulásait. A tanításokhoz az NVIDIA NeMo eszköztárát (Kuchaiev és mtsai, 2019) használtuk.

4. Eredmények

A 2. táblázat a BEA-Dialogue korpuszon elért eredményeket mutatja, míg a 3. táblázat a kiterjesztett BEA-Dialogue+ adathalmazon kapott értékeket tartalmazza. Mindkét esetben azonos modelleket és egységes kiértékelési protokollt alkalmaztunk, így az eredmények közvetlenül összehasonlíthatóak. Whisper (Radford és mtsai, 2022) és Fast Conformer(Rekesh és mtsai, 2023) modelleket értékeltünk ki.

A 2. táblázat a Gedeon és mtsai (2025) által közölt baseline értékeket tartalmazza (a finomhangolt modellt `fc_en_l (ft)`-ként jelölve), amelyeket két további, magyar adatokon előtanított FastConformer CTC Large és XLarge modellel egészítettünk ki, amik Dobsinszki és mtsai (2026) receptúráját követik. Az egyik ilyen modell (`fc_hu_l`) architekturálisan megegyezik az angol súlyokból finomhangolt Fast Conformer CTC Large modellel, míg a másik (`fc_hu_xl`) nagyobb méretű Fast Conformer CTC XL modell. Ezek a modellek 2700 órányi magyar beszédanyagon voltak előtanítva, aminek eredményeképpen finomhangolás nélkül is jelentősen jobb teljesítményt értek el, mint a Whisper modellek, sőt még az angol súlyokról finomhangolt Fast Conformer CTC modellnél is kedvezőbb eredményeket mutattak. A táblázatban a finomhangolt modellek `(ft)`

Model	dev					eval				
	WER	**cpWER**	**CER**	**cpCER**	**scAcc**	**WER**	**cpWER**	**CER**	**cpCER**	**scAcc**
`whisper-medium`	25.45	25.27	12.61	12.42	–	29.21	29.12	14.71	14.63	–
`whisper-large-v2`	19.65	19.42	9.84	9.58	–	24.50	24.42	13.13	13.05	–
`whisper-large-v3`	21.19	21.04	12.74	12.56	–	22.21	22.13	12.27	12.18	–
`fc_hu_l (zs)`	14.75	14.60	6.37	6.23	–	16.33	16.27	7.76	7.71	–
`fc_hu_xl (zs)`	13.27	13.17	5.89	5.76	–	15.48	15.43	7.47	7.42	–
`fc_en_l (ft)`	19.69	19.53	7.95	7.78	69.32	20.56	20.44	9.11	9.00	**82.16**
`fc_hu_l (ft)`	12.19	11.96	5.63	5.45	67.42	13.90	13.80	7.03	6.93	79.20
`fc_hu_xl (ft)`	**11.43**	**11.21**	**5.32**	**5.12**	**70.71**	**13.03**	**12.92**	**6.65**	**6.55**	80.94

2. táblázat. BEA-Dialogue beszédfelismerési eredmények (%).

Model	BEA-Dialogue+-dev					BEA-Dialogue+-eval				
	WER	**cpWER**	**CER**	**cpCER**	**scAcc**	**WER**	**cpWER**	**CER**	**cpCER**	**scAcc**
`whisper-medium`	30.83	30.73	15.96	15.83	–	30.19	30.06	16.00	15.86	–
`whisper-large-v2`	24.82	24.70	13.46	13.30	–	25.48	25.36	14.68	14.52	–
`whisper-large-v3`	23.28	23.15	13.15	12.99	–	23.27	23.17	13.34	13.22	–
`fc_hu_l (zs)`	18.07	17.99	8.11	8.02	–	18.91	18.80	9.50	9.40	–
`fc_hu_xl (zs)`	16.76	16.68	7.68	7.59	–	17.32	17.19	9.00	8.88	–
`fc_en_l (ft)`	16.30	16.11	7.42	7.23	**73.05**	16.49	16.28	8.29	8.11	**69.11**
`fc_hu_l (ft)`	14.97	14.78	7.12	6.94	70.64	15.19	14.99	7.95	7.79	67.99
`fc_hu_xl (ft)`	**12.84**	**12.67**	**6.31**	**6.15**	**73.05**	**13.59**	**13.42**	**7.34**	**7.18**	67.56

3. táblázat. BEA-Dialogue+ beszédfelismerési eredmények (%).

jelzéssel vannak ellátva, míg a finomhangolás nélküliek (zero-shot) `(zs)`-sel. A finomhangolás utáni eredmények tovább erősítik a korpusz relevanciáját: még az alapból magyarul tanult modellek is profitáltak a párbeszéd-specifikus továbbtanításból, ami kiemeli a BEA-Dialogue korpusz értékét a dialógusleiratozó modellek fejlesztésében.

A BEA-Dialogue+ eredményei alapján (ld. 3. táblázat) jól látható, hogy a finomhangolás nélküli modellek minden esetben gyengébben teljesítenek a BEA-Dialogue-hoz képest, jellemzően legalább 10% relatív romlást mutatva. Ennek egyik magyarázata a 2. ábrán látható beszélőváltás-eloszlás: míg a BEA-Dialogue-ban nagy számban fordulnak elő olyan szegmensek, amelyekben nulla vagy egyetlen váltás történik, addig a BEA-Dialogue+ esetében több a komplex, több váltást tartalmazó szegmens (különösen az *eval* halmaz esetén). Ez növeli a modell számára a feladat nehézségét, különösen akkor, ha a modell nem kap explicit információt a beszélőváltások határairól.

Ezzel szemben a finomhangolás jelentősebb javulást idéz elő ezen a korpuszon, aminek több oka is lehet. Egyrészt a jóval nagyobb mennyiségű tanítóadat miatt a finomhangolt modellek hatékonyabban képesek alkalmazkodni a BEA-Dialogue+ sajátosságaihoz, így a komplexebb, több beszélőváltással jellemezhető szegmenseket is jobban kezelik. Másrészt nem zárható ki, hogy a BEA-Dialogue

bővítése során létrejött beszélőbeli átfedések (adatszivárgás) is hozzájárultak ehhez a javuláshoz, ami a modell számára könnyebbé teheti egy-egy beszélő hangjának felismerését. Az előtanított modellek finomhangolás után ezen a korpuszon különösen nagy relatív javulást értek el, de így is elmaradtak a BEA-Dialogue-on mért eredményektől, tehát a feladat nem lett könnyebb a nagyobb adatmennyiség ellenére, így a korpusz hasonlóan hasznos rendszerek összehasonlítására.

Mindkét korpusz esetében megfigyelhető, hogy a cpWER és cpCER értékek a WER és CER értékekhez képest minden esetben alacsonyabbak. A javulás mértéke változó, de jellemzően a finomhangolt modellek mutatnak nagyobb javulást.

Az scAcc mutató alakulása összefüggést mutat a hibaarányokkal: a nehezebb szegmensekben (több beszélőváltással) a WER növekedésével párhuzamosan romlik a beszélőváltások felismerésének pontossága is. Ez azt sugallja, hogy az scAcc nemcsak a beszélőhatárok felismerésének sikerességét tükrözi, hanem közvetetten a feladat nehézségét is indikálja. A jobb scAcc érték viszont nem feltétlen eredményez jobb WER értéket, ahogy azt az `fc_en_l (ft)` illetve `fc_hu_l (ft)` modellek mutatják, előbbi hiába jobb az beszélőváltások meghatározásában, a leiratozásban utóbbi jobb.

A BEA-Dialogue+ esetében a korpuszbővítés során létrejött bizonyos mértékű beszélőátfedés kedvezően hathat a finomhangolt modellek teljesítményére. Ha a modell ugyanazon beszélő korábbi előfordulásaival már találkozott a tanítóadatokban, könnyebben képes adaptálódni a hanghoz és annak akusztikai jellemzőihez. Ennek a hatásnak a kvantitatív vizsgálata további elemzést igényel; ugyanakkor az eredmények alapján valószínűsíthető, hogy a javulás csak részben magyarázható adatszivárgással, mivel a legnagyobb teljesítménynövekedés a komplex szegmensekben mutatkozik, ahol inkább a párbeszédstruktúra modellezése lehet a döntő tényező.

A 4. táblázat egy konkrét példát illusztrál (a BEA-Dialogue+ *eval* halmazából), az egyes modellek kimeneteivel. Egy négy másodperc hosszú hangfájlról van szó, ami egy beszélgetés végén van. A hosszhoz képest sok szó elhangzik, ezek többsége egymásra beszélve, amit láthatóan csak a finomhangolt modellek képesek kezelni.

5. Összegzés

Ebben a tanulmányban bemutattuk a BEA-Dialogue+ korpuszt, amely a magyar nyelvű párbeszédes beszédfelismerési kutatások számára jelentős mennyiségi növekedést jelent a korábbi BEA-Dialogue adatbázishoz képest. A korpusz létrehozásának alapvető motivációja az volt, hogy a szigorú beszélőfüggetlenségi feltételek részleges enyhítésével – a főbeszélők teljes szeparációjának megőrzése mellett – jelentősen növeljük a rendelkezésre álló adatmennyiséget. Az eredmény egy 200 órás korpusz, amely közel két és félszeresére bővíti a korábbi 85 órás változatot.

A kísérleti eredmények szerint a korpuszbővítés kettős hatással jár. A finomhangolás nélküli, előtanított modellek teljesítménye a BEA-Dialogue+-on mintegy 10%-kal romlik a BEA-Dialogue-hoz képest, amit nagyrészt a megnövekedett

4. táblázat: Összehasonlítás konkrét példán keresztül (a BEA-Dialogue+ eval halmazából)

Referencia:
hát igen jó köszönöm szépen szerintem elég lesz ennyi <sc> köszönjük <sc> kösz

Modell	Leirat	WER	CER
`whisper-medium`	*jó köszönöm szépen szépen szerep*	72.73	60.29
`whisper-large-v2`	*igen jó köszönöm szépen köszönjük*	54.55	51.47
`whisper-large-v3`	*hát igen jó okosan szépen szívesen köszönjük*	54.55	44.12
`fc_hu_l (zs)`	*hát igen jó köszönöm szépen szépen*	54.55	51.47
`fc_hu_xl (zs)`	*hát igen jó köszönöm szépen*	54.55	60.29
`fc_en_l (ft)`	*hát igen jó köszönöm szépen szerintem elég lesz*	27.27	30.88
`fc_hu_l (ft)`	*hát igen jó köszönöm szépen szerintem elég lesz ennyi <sc> kösz*	**9.09**	**14.71**
`fc_hu_xl (ft)`	*hát igen jó köszönöm szépen szerintem elég lesz ennyi <sc> kösz*	**9.09**	**14.71**

beszélőváltások száma magyarázhat. Ezzel szemben a Serialized Output Training (SOT) módszerrel finomhangolt FastConformer modellek nagyobb relatív javulást mutatnak, mint a BEA-Dialogue-on.

A korpusz legnagyobb előnye a jelentősen megnövekedett adatmennyiség, amely lehetővé teszi a párbeszédes leiratozó rendszerek hatékonyabb tanítását. Emellett az egységes szegmensekre bontott korpusz (törekedve a 30 másodperces szegmensekre) ideális lehet rendszerek kiértékelésére, összehasonlítására. A beszélőátfedésből eredő potenciális adatszivárgás hatásának pontos mértéke további vizsgálatot igényel, például az eddig használtaktól független halmazon va-

ló kiértékeléssel, ami pontosabb képet adna az adatmennyiség növelésének és a beszélőátfedések hatásáról is. Mind a BEA-Dialogue mind a BEA-Dialogue+ regisztrált kutatók számára letölthetővé és így "Conversational AI" kutatások számára hasznosíthatóvá válik a közeli jövőben.

Köszönetnyilvánítás

A munka a Kulturális és Innovációs Minisztérium Nemzeti Kutatási Fejlesztési és Innovációs alapból nyújtott támogatásával, az EKÖP_KDP-25-1-BME-21 pályázati program finanszírozásában, a 2025-2.1.2-EKÖP-KDP-2025-00005 számú projekt, valamint az NKFIH K143075 és K135038 projektek támogatásával valósult meg.

Hivatkozások

Arısoy, E., Sak, H., Saraçlar, M.: Language modeling for automatic turkish broadcast news transcription. In: Interspeech 2007. pp. 2381–2384 (2007)

Dobsinszki, G., Mihajlik, P., Kádár, M.S., Fegyó, T., Mády, K.: Best data is more supervised data – even for hungarian asr. In: Karpov, A., Gosztolya, G. (szerk.) Speech and Computer. pp. 60–69. Springer Nature Switzerland, Cham (2026)

Gedeon, M., Barta, P.Z., Mihajlik, P., Gráczi, T.E., Kohári, A., Mády, K.: Toward conversational hungarian speech recognition: Introducing the BEA-Large and BEA-Dialogue datasets (2025), https://arxiv.org/abs/2511.13529

Kanda, N., Gaur, Y., Wang, X., Meng, Z., Yoshioka, T.: Serialized output training for end-to-end overlapped speech recognition. In: Interspeech (2020), https://api.semanticscholar.org/CorpusID:214714409

Kapoor, S., Narayanan, A.: Leakage and the reproducibility crisis in machine-learning-based science. Patterns 4(9), 100804 (2023), https://www.sciencedirect.com/science/article/pii/S2666389923001599

Kuchaiev, O., Li, J., Nguyen, H., Hrinchuk, O., Leary, R., Ginsburg, B., Kriman, S., Beliaev, S., Lavrukhin, V., Cook, J., Castonguay, P., Popova, M., Huang, J., Cohen, J.M.: Nemo: a toolkit for building ai applications using neural modules (2019), https://arxiv.org/abs/1909.09577

Mády, K., Etelka, G.T., Kohári, A., Mihajlik, P.: Revised annotation conventions in hungarian speech corpora. Beszédtudomány / Speech Science 4(1), 185–202 (2024), 18 p.

Mihajlik, P., Balog, A., Graczi, T.E., Kohari, A., Tarján, B., Mady, K.: BEA-base: A benchmark for ASR of spontaneous Hungarian. In: Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Odijk, J., Piperidis, S. (szerk.) Proceedings of the Thirteenth Language Resources and Evaluation Conference. pp. 1970–1977. European Language Resources Association, Marseille, France (Jun 2022), https://aclanthology.org/2022.lrec-1.211/

Mihajlik, P., Mády, K., Kohári, A., Fruzsina, F.S., Kiss, G., Gráczi, T.E., Doğruöz, A.S.: Is spoken Hungarian low-resource?: A quantitative survey of Hungarian speech data sets. In: Calzolari, N., Kan, M.Y., Hoste, V., Lenci, A., Sakti, S., Xue, N. (szerk.) Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024). pp. 9382–9388. ELRA and ICCL, Torino, Italia (May 2024), https://aclanthology.org/2024.lrec-main.820/

Neuberger, T., Gyarmathy, D., Gráczi, T.E., Horváth, V., Gósy, M., Beke, A.: Development of a large spontaneous speech database of agglutinative hungarian language. In: Sojka, P., Horák, A., Kopeček, I., Pala, K. (szerk.) Text, Speech and Dialogue. pp. 424–431. Springer International Publishing, Cham (2014)

Radford, A., Kim, J.W., Xu, T., Brockman, G., McLeavey, C., Sutskever, I.: Robust speech recognition via large-scale weak supervision (2022), https://arxiv.org/abs/2212.04356

Rekesh, D., Koluguri, N.R., Kriman, S., Majumdar, S., Noroozi, V., Huang, H., Hrinchuk, O., Puvvada, K., Kumar, A., Balam, J., Ginsburg, B.: Fast conformer with linearly scalable attention for efficient speech recognition (2023), https://arxiv.org/abs/2305.05084

Roger, V., Farinas, J., Pinquier, J.: Deep neural networks for automatic speech processing: A survey from large corpora to limited data (2020), https://arxiv.org/abs/2003.04241

Local LLM Deployment for Scalable and Real-Time AI-based interviewer

Benedek Huba Máth[1], Zita Kovács-Ördög[1]

[1]Clementine
{bmath, zordog}@clementine.hu

Abstract. Large Language Models (LLMs) have rapidly become central to natural language processing applications; however, their reliance on external APIs raises concerns about privacy, cost, and latency. In this work, we explore the feasibility of deploying Small Language Models (SLMs) locally for use in Minerva, an AI-based interviewer. We evaluated several open-source frameworks—Hugging Face Transformers, Ollama, and vLLM—on an NVIDIA GeForce RTX 4060 Ti, focusing on inference speed, stability, and accuracy. Using a dataset collected from Minerva's polling sessions, we benchmarked multiple SLMs against GPT-4o mini, currently used in production. Our findings show that vLLM combined with the Llama 3.1 8B model (GPTQ quantization) achieves substantial reductions in response latency while maintaining acceptable classification accuracy across multiple question types, although not all question types. These results demonstrate that locally hosted SLMs can already partially replace cloud based APIs in real-time conversational systems, paving the way for more private, scalable, and efficient AI voice assistants.

1 Introduction

The appearance of LLMs is arguably one of the biggest technological leaps in recent years. Their ability to understand context (to some degree) and generate human-like text in real time has sparked interest among both the public and professionals. Their capabilities also allow businesses to integrate them into existing workflows or build new applications around them. However, major issues remain. While widely available services such as OpenAI's ChatGPT API offer convenient access to LLMs, they require sending sensitive data to third parties, raising clear privacy concerns. Systems relying on external vendors are also vulnerable to changes, e.g.: an application built on a specific model can fail if the vendor discontinues it. Moreover, in some applications, time is critical, and the added latency can render the entire system unusable due to poor user experience.

The solution for these problems is running an LLM on local servers; however, this raises another set of challenges. The LLM has to fit the available hardware, which for many businesses is consumer-grade, since the expensive, high-capacity GPUs that are vital for training models like ChatGPT are often financially infeasible. Therefore, the model has to fit in the available amount of VRAM

while maintaining the required level of response quality and speed. Due to this, one of the most important parameter to optimize is the model size. Models that can fit the consumer-grade GPUs' with $\approx$ 16 GB VRAM are usually in the order of a few billion parameters. These are considerably smaller than the large-scale LLMs with more than 100 billion parameters, therefore in this article we will call them SLMs (Small Language Models). Due to the wide range of different use-cases the different frameworks and models have already received significant attention in the literature, however, these works usually compared metrics such as throughput (tokens/s) and latency using state-of-the-art GPUs like the A100, which offer little help to developers in optimizing their system for hardware constraints and focusing on user experience. (OpenAI et al., 2025) , (Kwon et al., 2023) , (Chitty-Venkata et al., 2024)

In this article, we investigate the open-source solutions available for running LLMs locally to switch the current LLM component, openAI's API, in our AI Voice Assistant, Minerva using an NVIDIA GeForce RTX 4060 Ti.

Minerva's task is to carry out a full public opinion poll automatically via phone in Hungarian. The system initiates a call to a telephone number and poses a predefined question using a text-to-speech (TTS) component. After the respondent answers, the audio signal is passed to an automatic speech recognition (ASR) component, and the resulting raw ASR output is provided to an LLM for automated response interpretation, that is, for assigning the response to a predefined answer category. For closed-ended questions, the answer category is represented as a numerically encoded value (e.g., 1 = male; 2 = female), whereas for open-ended questions the verbatim response (i.e., the ASR output) is returned; in this case, the LLM's task is limited to determining whether the received response is relevant with respect to the posed question. Based on the answer category assigned by the LLM, the subsequent step of the human–machine dialogue is determined. This step may involve terminating the call – if the respondent has completed the questionnaire or does not wish to participate – issuing a clarification or repetition of the previously asked question, or proceeding to the next question. Depending on the response to a given question, Minerva may automatically skip certain follow-up questions; for instance, if the respondent indicates that they do not intend to participate in the next parliamentary election, the system does not inquire about party preference. The raw ASR outputs, together with the answer categories assigned by the LLM, are logged during the call and subsequently processed after the completion of the campaign, including data cleaning, database structuring, and automated analysis. Our goal for this article is to significantly reduce the time it takes the LLM to process the answer, speeding up Minerva. This is essential, because Minerva's user experience highly depends on this factor, and if it is too slow we can lose participants. In the current version of it, the LLM that processes the answers is OpenAI's GPT-4o mini, however, we found that its response time is relatively slow, around $1 - 1.2$ s, while according to the literature in natural human conversation the response times are between $300 - 500$ms. Note, the total response time for Minerva is determined by the combined latency of the ASR and TTS models alongside the

LLM's inference time. These systems should also be improved, however in this article we focus on improving the LLM. It is important to note that it is not simply enough to speed up the process for one call, we require fast, concurrent, real-time responses from the SLM for multiple calls at the same time. (Meyer, 2023), (Stivers et al., 2009)

2 Data

We conducted our research on available local SLM frameworks and models, using a dataset we collected in May from the 12th to the 16th. The aim of the research, conducted using Minerva, was to map the general well-being, problem awareness, sense of security, and political attitudes of the Hungarian adult population. 50 robots—copies of Minerva—called at the same time completely automatically without any human intervention. Our data showed, that somewhere between $20 - 30$ calls was the average number of active calls at the same time, the remaining robots were waiting for the receiver to answer the phone. The telephone numbers were randomly selected. In the version of Minerva used during this data collection the operational LLM was GPT-4o mini. The questions we asked are shown in Table 1. From this point on, we will reference the questions with their IDs.

ID	Text	Category
Q1	Can we start?	3 categories
Q2	Are you a man or a woman?	2 categories
Q3	What do you think is the public mood in our country these days? Do most people feel good or bad?	3 categories
Q4	Who or what is primarily responsible for this mood?	open
Q5	What do you think is the biggest problem in the country today?	open
Q6	Which country or organization do you think poses the greatest threat to the world?	open
Q7	The first statement is that the cost of daily shopping has increased significantly. To what extent do you agree with this?	5 categories
Q8	The next statement is that the cost of living is a problem. Do you completely disagree with this, or do you tend to disagree, tend to agree, or completely agree?	5 categories
Q9	And how much do you agree that we have enough savings for the unexpected?	5 categories

ID	Text	Category
Q10	The following statement: If I get sick, I know I will be in good hands with public healthcare. To what extent do you agree with this?	5 categories
Q11	The following statement: I find the prospects of the young generation for the future worrying.	5 categories
Q12	And finally, I believe that there will be no war in Hungary in the next five years.	5 categories
Q13	Now I'm going to list political leaders, and I'd like to know if you had to describe them in one or two words, what would you say about them? So, how would you describe Viktor Orbán in one word?	open
Q14	What about Péter Magyar?	open
Q15	And Donald Trump?	open
Q16	And Vladimir Putin?	open
Q17	My next question: if parliamentary elections were held this Sunday, would you go to vote? Would you definitely go, or probably go, or probably not go, or definitely not go?	4 categories
Q18	If there were parliamentary elections this Sunday, which party would you vote for?	7 categories
Q19	Finally, I would like to ask what your highest level of education is? Primary, secondary or higher education?	4 categories

Table 1. The questions we asked during our data collection campaign in May. Some of the questions have a given number of categories, this means, that the SLM had to select the correct category for the given answer from that many. In the case of the open-ended questions, the SLM had to decide whether the answer received make sense, and if it does, it had return it back as it received. In the closed questions with a given number of categories we introduced another, which is the "I can not interpret the answer" category for the cases where the SLM is unable to determine the correct category. It is important to note, that this is different from the case where the responder explicitly says, they do not wish to answer.

We used 1000 of the questions' answers along with their output coming from GPT-4o mini as Minerva proceeded with those outputs. However, in order to determine how precise is that output, we manually evaluated a subset of the data. This hand-checked dataset consisted of 200 answers for the questions: Q2, Q17, Q18, Q3, Q9. Due to GDPR reasons, we cannot publish the raw answers coming from the responders, however a processed, cleaned dataset of the answers is available on https://minervaintezet.hu/ for research purposes.

The reasons why we selected these questions for hand-checking are the following:

– Q2 is one of the simplest question, if a model is not up to our expectations on this question, we do not investigate it further.

– Q17 and Q18 are among the most important questions in the field of political public opinion polls, therefore any models we might select to replace GPT-4o mini have to perform very well on these questions.
– Q3 is a binary categorical question (plus the "Did not answer"), which is also a common type of questions in public opinion polls.
– Q9 is more challenging since the differences between the categories are much narrower than in the other questions' cases. Any model that performs really well on this is a very good candidate for the replacement.

At this point we had not worked with the open questions yet. The reason for this decision, is that the purpose of this smaller, manually-checked dataset is to filter out the acceptable SLMs among the thousands of available, open-weight models for further investigations and we believed that the open questions are easier to evaluate—since the logic for them is that the SLM has to determine whether the responder answered the question and if they did, return the original text—than those we selected. In Table 2. we compared the hand-checked categories to the GPT-4o mini categories. For Q2, Q18 and Q3 the differences are effectively negligible, however for Q17 and Q9 they are significant. Upon further analysis, the reasons for this are the following; the model has a slight lack of ability to make a distinction between the similar intents like: tend to agree, completely agree; in addition the model can not handle the double negation that is quite common in Hungarian e.g.: 'Nem lehet nem rá szavazni'-'It is impossible not to vote for him'.

ID	Q2	Q17	Q18	Q3	Q9
Number of matching answers	200 (100%)	184 (92%)	198 (98%)	196 (96%)	176 (88%)

Table 2. The number of matching answers between the hand-checked and GPT-4o mini versions, for clarity, relative agreement rates (percentages) are reported alongside absolute counts. In the case of the two 4-category questions, Q17 and Q9, the differences are significant, the others are negligible. This is due to two reasons: one, the slight inability of the model to make a distinction between the similar intents like: completely agree or tend to agree. Second, the double negation which is relatively common in Hungarian, but tends to mislead the model.

3 Frameworks

We experimented with several frameworks for running SLMs locally. While user-friendly applications like GPT4ALL exist, our use case required precise parameter control and integration within a 16GB VRAM (NVIDIA RTX 4060 Ti) hardware constraint. We initially utilized Hugging Face Transformers and Ollama but migrated to vLLM to address performance bottlenecks and lack of well-developed concurrent processing. All selected frameworks and models are free and compatible with commercial deployment.

3.1 Hugging Face Transformers

The Hugging Face Transformers library provides a unified API for state-of-the-art NLP models across PyTorch and TensorFlow. Its primary advantage is the extensive Model Hub, which facilitates local execution for privacy-sensitive or resource-constrained environments. While highly modular and supported by optimization tools like Accelerate and Safetensors, it can be less efficient than specialized inference engines for high-throughput tasks. (Wolf et al., 2020)

3.2 Ollama

Ollama is a lightweight framework that simplifies local LLM deployment via a command-line interface. Built upon `llama.cpp`, it supports GGUF quantization, allowing the deployment of larger models that might exceed VRAM limits when using standard Transformers formats. While Ollama excels in ease of use and local memory management, it proved insufficient for our specific requirements regarding high-concurrency workloads.

3.3 vLLM

vLLM is a high-throughput library optimized for memory-efficient inference. Its core innovation, **PagedAttention**, minimizes memory fragmentation and maximizes GPU utilization, enabling larger batch sizes and lower latency. We specifically leveraged **Automatic Prefix Caching (APC)**, which caches the KV cache for shared prompt prefixes to reduce inference time. This is particularly effective for our Minerva implementation, as detailed in Section 4. Additionally, vLLM supports the Marlin kernel for GPTQ quantization, which our testing confirmed significantly accelerates inference on consumer-grade hardware.

4 Methodology

In this section, we describe the tests we conducted on the available data and how we evaluated the results. In the first test, we fed the answers to the SLM, measured run time and compared the output to the manually verified version. As mentioned in Section 1. the main goal is to reduce the response processing time while maintaining the required quality of the output. In the second test, we focused on the concurrent inferences speed.

4.1 First test

In the first test, we worked with Hugging Face Transformers and Ollama. We used the manually verified categories as a comparison, and knowing that the SLM is prone to misclassifying the Q17 and Q9 questions, we accepted the SLM's output if it differed the manually verified class by one category, e.g.: If the *real* output was "completely agree" and the SLM's output was "tend to

agree", we accepted it, this is because the path of the questionnaire changes only if the answer is 'completely disagree'—e.g.: if the receiver does not wish to vote in the next election, we will not ask who would they vote for—while in Minerva's operation the differences like the one between 'tend to agree' and 'completely agree' are negligible. However, the errors due to double negation, where the SLM's output was "completely disagree", while the *real* output was "completely agree" or "tend to agree" remained, these could affect the poll's course, leaving out some questions. Apart from the number of matching outputs, other important statistics are the number of outputs in the correct form. This is due to the fact that the SLM has to integrate seamlessly into Minerva, and a key part of this is producing the output in the form that is compatible with the other blocks. And last but certainly not least, we measured the run time. In this case, when we speak about run time, we mean the time that elapsed between the SLM receiving the prompt until it produces an output.

4.2 Second test

In the second test, we worked with vLLM. We used the same methodology as we did in the first test for scoring, however one major difference is that since the hand-categorized answers are not enough for this test we used the whole dataset and we accepted the GPT 4o mini's categories as the *correct* ones, as Minerva successfully already ran multiple poll campaigns using it. In order to test the framework capabilities of concurrent inference, we loaded the available prompts into channels separated by random breaks with uniform distributions representing the time it takes the responder to answer Minerva's next question. Although mathematically an exponential distribution should be used in this case, we wanted to make sure that these breaks remain between controlled bounds. We also shuffled the prompts in order to avoid abusing the APC mechanics, although in the log messages after the first run we always saw 90%+ prefix cache hit rate. During the test, we changed breaks' distribution. The model of this test is shown in Figure 1. By concurrent inference we mean that the framework should be able to process different responses simultaneously and it can happen that while one channel is processing one, another is waiting for a prompt.

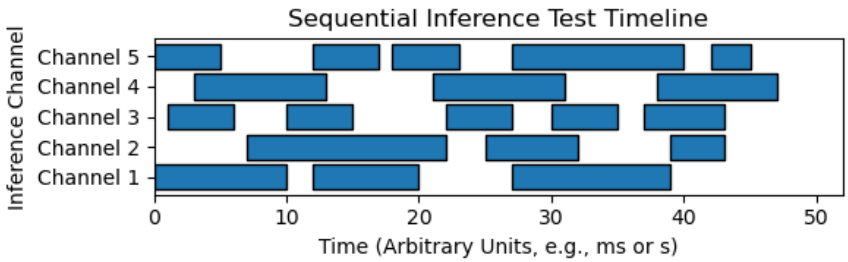


Fig. 1: The model of the vLLM test. The prompts are processed across a specified number of channels. The blue blocks represent the time it takes the SLM to process the prompt and the white blocks represent the breaks between the prompts.

4.3 Prompt Engineering

During baseline testing we made slight changes to the production prompt. In the original version we required a given format where the model had to produce specific Hungarian characters (éáűúőóüö). We realized, that the smaller, faster models are prone to miss these e.g.: write "o" instead of "ó". Therefore we made sure to switch the output's language completely to English, since this does not affect the user experience on the other side of the phone, Minerva still speaks Hungarian. Another key takeaway was that, in order to speed up the process, we had to significantly decrease the number of output tokens. This meant that we changed our encodings, instead of requiring complete words such as: "tend to agree", we switched to numeric labels, for example 1 for completely agreeing and 4 for completely disagreeing. In some questions, especially where the border between the categories is blurred, e.g.: the border between "completely agree" and "tend to agree" is much more blurred than in the case of gender identification, we require the output to have the form:

$$\{'category' : 1,' confidence' : 60\}, \{'category' : 2,' confidence' : 40\}$$

, a Python list with dictionaries in it, where 'confidence' shows how sure the model is in its categorization. In the second test we tested how does the run time change if we only require the *'category'* in the output.

Another important lesson is that in text classification the SLMs heavily rely on the given examples and system prompts. By listing more examples to each category, the accuracy increased. Also in case of the vLLM the APC mechanism required the prompts to be in the form: PREFIX PART + DYNAMIC PART. In one of the baseline tests we managed to get the inference time from 1.95 s to 1.24 s just by taking advantage of the APC. The system prompt we used is relatively simple: *"You are a helpful assistant. You give back the answers exactly as requested! You don't write anything else!"*. Apart from these changes we did not perform further prompt optimization for any model beyond evaluation-driven

adjustments. These modifications were not validated on a separate development set, which may affect generalization and is left for future work.

5 Results and Discussion

In this section, we present our results with the 3 different frameworks and models. We report only the results for the models that managed to pass the initial baseline tests. One important lesson from the baseline tests was that the smaller models—typically in the range of 1-4B parameters—lack sufficient capability to process Hungarian text. These include being able to load the model into the limited VRAM and with a small amount of prompt engineering achieving an acceptable output on the simplest question, Q2.

5.1 First test results

Our main concern is the run time; therefore, we present the run times first in Table 3. Both the mean times and the standard deviations are calculated from the 200 answers we worked with. In the table the "Mean" refers to the average run time of the 200 answers measured in seconds, the "Deviation" is the standard deviation of these times. We highlighted the best model in this case, which is the Llama 3.1 8B with Ollama framework. It was expected that Ollama would generally be faster, as it worked with more quantized versions of the models, in fact comparing the Hugging Face's Llama 3.1 results we can see, that the Hugging Face version is taking approximately twice as much time than the Ollama version. It is also no surprise that the gpt-oss-20b is the slowest, as it is the largest model, this trend is further proved by the second largest model, Gemma3 12B, it is the second slowest model in the Ollama framework. Due to their sizes we failed to load them into the Hugging Face framework. The standard deviations are used to determine how stable the model is in terms of run time. There are some cases, where the deviation is larger than the mean value, which indicates that these models' tend to produce extreme values, therefore they are less stable than others. Even in this statistics, the Llama 3.1 model proves to be the best, since its maximum deviation is 0.087 which is acceptable for our goal.

Question		Q2		Q17		Q18		Q3		Q9	
Framework	Model	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.	Mean	Std.
HF	Llama 3.1 8B	0.86	0.004	1.02	0.139	0.90	0.009	0.90	0.006	1.09	0.292
HF	Mistral 7B	1.00	0.052	1.62	2.136	1.09	0.682	1.41	1.742	1.36	1.104
HF	Qwen 2.5 7B	0.89	0.025	1.20	0.28	0.95	0.047	0.90	0.018	1.27	0.326
Ollama	gpt-oss-20b	3.15	1.316	3.17	1.989	3.20	1.774	3.34	1.856	4.33	3.234
Ollama	**Llama 3.1 8B**	**0.43**	**0.025**	**0.57**	**0.087**	**0.47**	**0.023**	**0.48**	**0.014**	**0.54**	**0.082**
Ollama	Gemma 3 12B Q4	0.95	0.078	1.52	0.142	1.37	0.065	1.11	0.064	1.50	0.094
Ollama	Mistral 7B	0.37	0.153	0.61	0.876	0.45	0.139	0.64	0.666	0.59	0.422

Table 3. The run times and their deviation for the different models. In this table HF stands for Hugging Face, Ollama for Ollama server and Std. for standard deviation. All of the models were 'Instruct' versions. The Llama 3.1 model with Ollama framework is highlighted as it performed the best in terms of Mean run times for Q17, Q3, Q9. In Q2 and Q18 Mistral is slightly faster in Mean, however in these cases Llama has a much smaller Deviation, which means its run time is more predictable, which better suits our system.

In Table 4. we present our results for accuracy. The "NoCFA"—**N**umber **o**f **C**orrectly **F**ormatted **A**nswers—columns show, how many of the inputs got an output in the correct form, that is compatible with other parts of Minerva. Their maximum value is 200. The "Score" columns show the ratio of the correctly-formed answers that matched with the manually verified categories. This means that if for a model the "NoCFA" is 100, and their score is 0.5, then the number of matching outputs is 50. If any model produces somewhere around $90+\%$, we consider it acceptable. In this table we highlighted the fastest model, Llama 3.1 and even in terms of accuracy it is acceptable for us.

Question		Q2		Q17		Q18		Q3		Q9	
Framework	Model	Score	NoCFA	Score	NoCFA	Score	NoCFA	Score	NoCFA	Score	NoCFA
HF	Llama 3.1 8B	0.99	200	0.755	200	0.90	200	0.885	200	0.845	200
HF	Mistral 7B	0.985	200	0.838	191	0.905	199	0.937	190	0.706	197
HF	Qwen 2.5 7B	0.99	200	0.835	200	0.81	200	0.915	200	0.75	200
Ollama	gpt-oss-20b	1	198	0.91	200	0.90	200	0.93	200	0.810	200
Ollama	**Llama 3.1 8B**	**0.99**	**200**	**0.85**	**200**	**0.89**	**198**	**0.89**	**200**	**0.81**	**200**
Ollama	Gemma 3 12B Q4	1	148	0.90	198	0.75	24	0.92	38	0.93	177
Ollama	Mistral 7B	0.99	195	0.89	194	0.88	196	0.92	157	0.82	159

Table 4. The different frameworks and models' results in terms of accuracy. The Llama 3.1 model in the Ollama framework is highlighted, since it is the best in terms of run times. We consider this accuracy result to be acceptable.

5.2 Second test results

In this test, we worked with the entire available dataset, 1000 answers for each question. In this case, we only had the categories from GPT 4o mini from the campaign. Using the first test's result, we only worked with the Llama 3.1

model, since by far it was the best combination of speed and accuracy. We tested two quantization types, one called 'AWQ', the other 'GPTQ'. For this test, we fixed the number of channels to 20. During this, we noticed that the accuracy for the open-ended questions is extremely poor. Our investigation showed, that even the GPT-4o mini performs poorly at returning text exactly as received often introducing changes. Therefore we made some changes, instead of getting back the original response, if it was an answer, we only wanted the model to decide, if the question has been answered. Even with this change we got relatively poor results, which indicates that additional improvements will be necessary in future work. We present a part of our results in Table 5, the whole test result is in the Appendix. The 'With confidence' and 'Without confidence' means, whether we require the 'confidence' key in the LLM's output or not, e.g.:

$$[\{'category' : 1,' confidence' : 60\}, \{'category' : 2,' confidence' : 40\}]$$

or

$$[\{'category' : 1\}, \{'category' : 2\}]$$

The results show, that the GPTQ quantization is much faster and they have very similar accuracy. It is also clear that despite the changes we made in the open-ended question's case the accuracy is still not at an acceptable level. The mean run time is again reported in seconds, the score and the NoCFA are the same as in the Ollama's case.

Questions	AWQ						GPTQ					
	With confidence			Without confidence			With confidence			Without confidence		
	score	NoCFA	Mean	score	NoCFA	Mean	score	NoCFA	Mean	score	NoCFA	Mean
Q2	0.98	1000	2.23	0.99	998	1.47	0.98	996	0.81	0.98	995	0.53
Q17	0.95	993	2.86	0.95	993	2.28	0.96	982	0.88	0.96	986	0.63
Q18	0.94	958	2.16	0.92	992	1.48	0.95	961	0.80	0.94	979	0.51
Q3	0.93	1000	2.23	0.93	999	1.47	0.93	998	0.82	0.93	996	0.53
Q9	0.90	1000	2.94	0.90	1000	2.28	0.90	1000	0.89	0.90	1000	0.89
Q16	0.58	1000	2.29	0.58	998	1.51	0.70	1000	0.85	0.71	1000	0.55

Table 5. A portion of the results we got with the vLLM framework using Llama 3.1 8B with two different quantizations. Mean stands for the mean run time. The rest of our results are in the Appendix.

Finally, to ensure that the vLLM framework can replace the OpenAI API in production, we modified the breaks' distribution and measured the response times. For each run, we calculated the mean and the 95th percentile of the run times for all questions, and then computed the mean of these statistics and their standard deviation. We present these results in Figure 2. We can clearly see that when in the output we require the confidence level as well, the model becomes much slower; however, even then, in the realistic break range -1-3 seconds, it might take this long or even longer for the responder to understand the question

and think their answer through the model is much faster than our current system, as it takes around $1.0 - 1.2$ seconds to process the user's response.

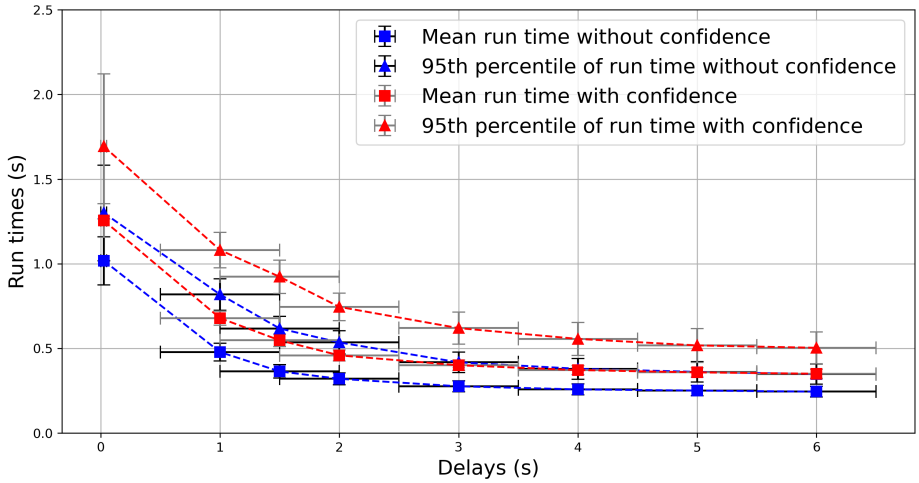


Fig. 2: The average of the mean and 95th percentile of the runs. The results with confidence required are in red, and without confidence are in blue. The x-axis error bars represent the break distribution's lower and upper limits—the breaks followed uniform distribution—and the y-axis error bars show the standard deviation of these statistics across the different questions. As expected, when we require the SLM to output confidence levels as well, they get slower. The results also demonstrate, that in the realistic break range—1-3 seconds—the model is still considerably faster than our current system.

6 Conclusions and future plans

In this study, we presented our test results for several local LLM frameworks and models using data we collected in May by our AI Voice Assistant, Minerva. We demonstrated that it is feasible to build a stable system using the available frameworks and SLMs that can partially replace OpenAI's API in latency-critical components. In the future we will integrate the best framework-model pair into the system, which turned out to be vLLM with Llama 3.1 8B with GPTQ quantization. Additionally, we intend to improve the model's performance on the open-ended questions which is currently not at an acceptable level and further enhance our system by implementing the structured output option of the prompts.

Bibliography

Chitty-Venkata, K.T., Raskar, S., Kale, B., Ferdaus, F., Tanikanti, A., Raffenetti, K., Taylor, V., Emani, M., Vishwanath, V.: Llm-inference-bench: Inference benchmarking of large language models on ai accelerators (2024), https://arxiv.org/abs/2411.00136

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C.H., Gonzalez, J.E., Zhang, H., Stoica, I.: Efficient memory management for large language model serving with pagedattention (2023), https://arxiv.org/abs/2309.06180

Meyer, A.S.: Timing in conversation. Journal of Cognition 6(1), 20 (2023), https://doi.org/10.5334/joc.268

OpenAI, :, Agarwal, S., Ahmad, L., Ai, J., Altman, S., Applebaum, A., Arbus, E., Arora, R.K., Bai, Y., Baker, B., Bao, H., Barak, B., Bennett, A., Bertao, T., Brett, N., Brevdo, E., Brockman, G., Bubeck, S., Chang, C., Chen, K., Chen, M., Cheung, E., Clark, A., Cook, D., Dukhan, M., Dvorak, C., Fives, K., Fomenko, V., Garipov, T., Georgiev, K., Glaese, M., Gogineni, T., Goucher, A., Gross, L., Guzman, K.G., Hallman, J., Hehir, J., Heidecke, J., Helyar, A., Hu, H., Huet, R., Huh, J., Jain, S., Johnson, Z., Koch, C., Kofman, I., Kundel, D., Kwon, J., Kyrylov, V., Le, E.Y., Leclerc, G., Lennon, J.P., Lessans, S., Lezcano-Casado, M., Li, Y., Li, Z., Lin, J., Liss, J., Lily, Liu, Liu, J., Lu, K., Lu, C., Martinovic, Z., McCallum, L., McGrath, J., McKinney, S., McLaughlin, A., Mei, S., Mostovoy, S., Mu, T., Myles, G., Neitz, A., Nichol, A., Pachocki, J., Paino, A., Palmie, D., Pantuliano, A., Parascandolo, G., Park, J., Pathak, L., Paz, C., Peran, L., Pimenov, D., Pokrass, M., Proehl, E., Qiu, H., Raila, G., Raso, F., Ren, H., Richardson, K., Robinson, D., Rotsted, B., Salman, H., Sanjeev, S., Schwarzer, M., Sculley, D., Sikchi, H., Simon, K., Singhal, K., Song, Y., Stuckey, D., Sun, Z., Tillet, P., Toizer, S., Tsimpourlas, F., Vyas, N., Wallace, E., Wang, X., Wang, M., Watkins, O., Weil, K., Wendling, A., Whinnery, K., Whitney, C., Wong, H., Yang, L., Yang, Y., Yasunaga, M., Ying, K., Zaremba, W., Zhan, W., Zhang, C., Zhang, B., Zhang, E., Zhao, S.: gpt-oss-120b gpt-oss-20b model card (2025), https://arxiv.org/abs/2508.10925

Stivers, T., Enfield, N.J., Brown, P., Englert, C., Hayashi, M., Heinemann, T., Hoymann, G., Rossano, F., de Ruiter, J.P., Yoon, K.E., Levinson, S.C.: Universals and cultural variation in turn-taking in conversation. Proceedings of the National Academy of Sciences 106(26), 10587–10592 (2009), https://www.pnas.org/doi/abs/10.1073/pnas.0903616106

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M.: Transformers: State-of-the-art natural language processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 38–45. Association for Computational Linguistics, Online (Oct 2020), https://www.aclweb.org/anthology/2020.emnlp-demos.6

Appendix

model type	AWQ	AWQ	GPTQ	GPTQ
confidence	True	False	False	True
number of channels	20	20	20	20
Q1 score	0.98	0.98	0.97	0.96
Q1 NoCFA	1000	1000	1000	1000
Q1 time to answer	2.23	1.48	0.54	0.82
Q1 time deviation	0.063	0.052	0.198	0.203
Q1 50th percentile	2.232	1.474	0.581	0.868
Q1 75th percentile	2.267	1.512	0.7	0.96
Q1 95th percentile	2.305	1.552	0.823	1.075
Q1 max time	3.46	1.8	0.97	1.25
Q2 score	0.98	0.99	0.98	0.98
Q2 NoCFA	1000	998	995	996
Q2 time to answer	2.23	1.47	0.53	0.81
Q2 time deviation	0.104	0.074	0.201	0.199
Q2 50th percentile	2.227	1.477	0.558	0.857
Q2 75th percentile	2.265	1.511	0.685	0.947
Q2 95th percentile	2.303	1.55	0.824	1.06
Q2 max time	4.02	1.99	1	1.26
Q3 score	0.93	0.93	0.93	0.93
Q3 NoCFA	1000	999	996	998
Q3 time to answer	2.23	1.47	0.53	0.82
Q3 time deviation	0.101	0.063	0.199	0.208
Q3 50th percentile	2.232	1.471	0.577	0.86
Q3 75th percentile	2.266	1.508	0.693	0.963
Q3 95th percentile	2.3	1.546	0.818	1.075
Q3 max time	4.11	1.99	0.99	1.36
Q4 score	0.84	0.83	0.86	0.84
Q4 NoCFA	1000	1000	1000	1000
Q4 time to answer	2.28	1.51	0.55	0.82
Q4 time deviation	0.133	0.084	0.2	0.209
Q4 50th percentile	2.259	1.503	0.589	0.865
Q4 75th percentile	2.319	1.564	0.707	0.972
Q4 95th percentile	2.428	1.674	0.836	1.08
Q4 max time	4.17	1.88	1.09	1.27
Q5 score	0.79	0.74	0.78	0.76
Q5 NoCFA	1000	1000	1000	1000
Q5 time to answer	2.05	1.36	0.53	0.78
Q5 time deviation	0.369	0.084	0.196	0.189
Q5 50th percentile	1.977	1.347	0.558	0.814
Q5 75th percentile	2.074	1.396	0.68	0.913
Q5 95th percentile	2.217	1.517	0.8	1.011

model type	AWQ	AWQ	GPTQ	GPTQ
confidence	True	False	False	True
Q5 max time	6.54	2.02	1.26	1.21
Q6 score	0.79	0.79	0.82	0.79
Q6 NoCFA	1000	1000	1000	1000
Q6 time to answer	2.33	1.54	0.56	0.83
Q6 time deviation	0.276	0.114	0.199	0.208
Q6 50th percentile	2.292	1.521	0.594	0.861
Q6 75th percentile	2.381	1.615	0.709	0.976
Q6 95th percentile	2.442	1.688	0.833	1.085
Q6 max time	6.06	3.44	1.3	1.34
Q7 score	0.93	0.94	0.9	0.9
Q7 NoCFA	1000	1000	1000	1000
Q7 time to answer	3.16	2.24	0.6	0.87
Q7 time deviation	0.914	0.393	0.22	0.239
Q7 50th percentile	3.96	2.364	0.62	0.885
Q7 75th percentile	4.052	2.41	0.764	1.018
Q7 95th percentile	4.106	2.455	0.962	1.256
Q7 max time	5.85	5.14	1.4	1.47
Q8 score	0.93	0.92	0.92	0.92
Q8 NoCFA	1000	1000	1000	1000
Q8 time to answer	2.88	2.18	0.6	0.86
Q8 time deviation	0.869	0.476	0.218	0.228
Q8 50th percentile	2.276	2.361	0.623	0.884
Q8 75th percentile	4.018	2.412	0.749	1.003
Q8 95th percentile	4.096	2.486	0.94	1.214
Q8 max time	4.17	5.12	1.39	1.47
Q9 score	0.9	0.9	0.9	0.9
Q9 NoCFA	1000	1000	1000	1000
Q9 time to answer	2.94	2.28	0.63	0.89
Q9 time deviation	0.888	0.487	0.215	0.235
Q9 50th percentile	2.285	2.368	0.649	0.908
Q9 75th percentile	4.032	2.417	0.777	1.031
Q9 95th percentile	4.103	3.227	0.954	1.279
Q9 max time	5.21	7.26	1.17	1.52
Q10 score	0.94	0.94	0.94	0.93
Q10 NoCFA	1000	1000	1000	1000
Q10 time to answer	2.91	2.28	0.62	0.86
Q10 time deviation	0.885	0.488	0.223	0.239
Q10 50th percentile	2.269	2.365	0.649	0.887
Q10 75th percentile	4.028	2.414	0.775	1.005
Q10 95th percentile	4.102	3.231	0.948	1.231
Q10 max time	5.93	6.92	1.42	1.86
Q11 score	0.88	0.89	0.87	0.87

model type	AWQ	AWQ	GPTQ	GPTQ
confidence	True	False	False	True
Q11 NoCFA	1000	1000	1000	1000
Q11 time to answer	2.92	2.23	0.62	0.88
Q11 time deviation	0.883	0.414	0.225	0.22
Q11 50th percentile	2.276	2.365	0.65	0.892
Q11 75th percentile	4.019	2.415	0.786	1.003
Q11 95th percentile	4.101	2.491	0.968	1.242
Q11 max time	5.8	5.13	1.4	1.55
Q12 score	0.86	0.86	0.86	0.83
Q12 NoCFA	1000	999	1000	1000
Q12 time to answer	3	2.29	0.63	0.89
Q12 time deviation	0.9	0.616	0.218	0.221
Q12 50th percentile	2.287	2.375	0.667	0.905
Q12 75th percentile	4.035	2.424	0.788	1.015
Q12 95th percentile	4.104	2.606	0.947	1.249
Q12 max time	5.22	15.21	1.24	1.47
Q13 score	0.69	0.7	0.8	0.78
Q13 NoCFA	1000	994	1000	1000
Q13 time to answer	2.31	1.5	0.55	0.85
Q13 time deviation	0.283	0.174	0.197	0.192
Q13 50th percentile	2.257	1.489	0.589	0.876
Q13 75th percentile	2.312	1.535	0.703	0.988
Q13 95th percentile	2.437	1.657	0.825	1.096
Q13 max time	4.1	6.22	0.98	1.39
Q14 score	0.55	0.56	0.7	0.7
Q14 NoCFA	1000	997	1000	1000
Q14 time to answer	2.3	1.5	0.54	0.84
Q14 time deviation	0.232	0.091	0.203	0.208
Q14 50th percentile	2.26	1.486	0.573	0.875
Q14 75th percentile	2.323	1.536	0.695	0.977
Q14 95th percentile	2.431	1.666	0.83	1.096
Q14 max time	4.2	2.19	1.32	1.41
Q15 score	0.59	0.59	0.72	0.71
Q15 NoCFA	1000	998	1000	1000
Q15 time to answer	2.3	1.5	0.56	0.84
Q15 time deviation	0.331	0.087	0.198	0.194
Q15 50th percentile	2.257	1.489	0.597	0.879
Q15 75th percentile	2.318	1.533	0.711	0.975
Q15 95th percentile	2.432	1.651	0.839	1.083
Q15 max time	9.43	2.45	1.31	1.29
Q16 score	0.58	0.58	0.71	0.7
Q16 NoCFA	1000	998	1000	1000
Q16 time to answer	2.29	1.51	0.55	0.85

model type	AWQ	AWQ	GPTQ	GPTQ
confidence	True	False	False	True
Q16 time deviation	0.221	0.162	0.2	0.204
Q16 50th percentile	2.255	1.492	0.582	0.884
Q16 75th percentile	2.311	1.535	0.705	0.981
Q16 95th percentile	2.434	1.652	0.84	1.114
Q16 max time	4.12	5.86	1.31	1.39
Q17 score	0.95	0.95	0.96	0.96
Q17 NoCFA	993	993	986	982
Q17 time to answer	2.86	2.28	0.63	0.88
Q17 time deviation	0.894	0.39	0.227	0.239
Q17 50th percentile	2.271	2.372	0.653	0.904
Q17 75th percentile	4.013	2.413	0.79	1.013
Q17 95th percentile	4.099	2.461	0.965	1.236
Q17 max time	5.88	5.94	1.41	1.57
Q18 score	0.94	0.92	0.94	0.95
Q18 NoCFA	958	992	979	961
Q18 time to answer	2.16	1.48	0.51	0.8
Q18 time deviation	0.376	0.19	0.203	0.227
Q18 50th percentile	2.231	1.477	0.55	0.851
Q18 75th percentile	2.271	1.514	0.67	0.949
Q18 95th percentile	2.315	1.568	0.812	1.068
Q18 max time	4.11	6.27	1.27	1.45
Q19 score	0.89	0.89	0.88	0.89
Q19 NoCFA	1000	1000	1000	1000
Q19 time to answer	2.24	1.48	0.54	0.82
Q19 time deviation	0.138	0.063	0.198	0.204
Q19 50th percentile	2.232	1.478	0.582	0.864
Q19 75th percentile	2.269	1.516	0.696	0.963
Q19 95th percentile	2.332	1.565	0.821	1.079
Q19 max time	4.1	2.03	1.32	1.35

Table 6. All of the results with the two quantizations with and without confidence using vLLM and Llama 3.1.

Mesterséges párbeszédek, valós eredmények – dialógus-szimuláció a pontosabb leiratozásért

Gedeon Máté[1,2], Mihajlik Péter[1,3]

[1] Budapesti Műszaki és Gazdaságtudományi Egyetem,
Távközlési és Mesterséges Intelligencia Tanszék
[2] Speechtex Kft.
[3] ELTE Nyelvtudományi Kutatóközpont
gedeonm@edu.bme.hu, mihajlik@tmit.bme.hu

Kivonat A spontán, több-beszélős beszélgetések pontos gépi leiratozása továbbra is komoly kihívást jelent a beszédfelismerő rendszerek számára, különösen olyan nyelveken, ahol korlátozott a párbeszédes tanítóadatok mennyisége. Munkánk célja a párbeszéd-szimuláció magyar nyelvű megvalósítása volt, amely képes a valós beszélgetésekre jellemző egyéni időzítési mintázatok reprodukálására. A megvalósítás alapját a BEA-Large korpusz egyszereplős felvételei képezték, amelyekből statisztikai modellezés (kernel-alapú sűrűségfüggvény-becslés és Markov-lánc) segítségével generáltunk kétbeszélős, valósághű időzítésű szimulált párbeszédeket. A szimulációhoz a CallHome és a BEA-Dialogue természetes beszédkorpuszokból kinyert eloszlásokat alkalmaztuk, és megvizsgáltuk a térszimuláció hatását is. Az így előállított szintetikus adatokkal kiegészített tanítóhalmazzal finomhangolt Fast Conformer CTC modellek következetesen javuló teljesítményt mutattak, mind szószintű, mind karakterszintű hibaarányok tekintetében. Eredményeink azt mutatják, hogy a beszélőfüggő párbeszéd-szimuláció hatékony módszer a magyar nyelvű, több-beszélős beszédfelismerés teljesítményének javítására.
Kulcsszavak: gépi beszédfelismerés, spontán beszéd, párbeszéd-szimuláció

1. Bevezetés

A spontán beszélgetések leiratozása a mai napig komoly kihívást jelent a gépi beszédfelismerők számára. Az olyan sajátosságok, mint az egymásra beszélés, vagy a közbeszólások megnehezítik a modellek számára a pontos felismerést (Yu és mtsai, 2016; Kanda és mtsai, 2020). Ennek következtében azok a modellek, amelyeket elsősorban tiszta monológokat tartalmazó beszédanyagon tanítottak, gyakran nehézségekbe ütköznek természetes, többszereplős beszélgetések feldolgozása során, ahol előfordul egyidejű beszéd vagy gyors beszélőváltás.

Robusztus, több-beszélős modellek tanításához nagy mennyiségű, párbeszédes stílusú hanganyagra van szükség. Magyar nyelven azonban ilyen adatból komoly hiány mutatkozik, mivel előállításuk rendkívül munkaigényes annotációt igényel. Az adathiány egyik ígéretes megoldása beszélgetések szimulálása, pontosabban új párbeszédek létrehozása különálló, egybeszélős felvételek egymás

után helyezésével. Mára bevett gyakorlat ehhez figyelembe venni, hogy általában az emberek mekkora szüneteket tartanak mondataik között, vagy mennyire hajlamosak a másik szavába vágni (Landini és mtsai, 2022). A gyakorlatban ez tipikusan olyan statisztikai modellekkel történik, amelyek a természetes beszélgetések mintázatait – például a megszólalások közti szünetek hosszát vagy az átfedések gyakoriságát – próbálják reprodukálni. A legtöbb korábbi megközelítés azonban kizárólag általános statisztikákat használ és a beszélőket egymással felcserélhetőnek tekinti. Ennek következtében a szimulált párbeszédek gyakran elveszítik a valós beszélgetések szereplőire jellemző egyéni sajátosságokat.

E probléma kezelésére született meg a beszélőfüggő párbeszéd-szimuláció (speaker-aware conversation simulation) (Gedeon és Mihajlik, 2025a) koncepciója. Ennek alapgondolata, hogy minden beszélő őrizze meg a saját, jellemző mintázatait – hasonlóan ahhoz, ahogyan egy valós beszélgetésben viselkedne. Ahelyett, hogy minden szünet- vagy átfedéshosszt egyetlen általános eloszlás jellemezne, minden beszélőre külön eloszlás van illesztve az adatok segítségével. Például ha egy adott személy jellemzően gyorsan reagál, a szimuláció ezt konzisztensen tükrözi minden alkalommal, amikor ez a beszélő kerül sorra. Az angol nyelvű dialóguskorpuszokon végzett kiértékelések azt mutatták, hogy a beszélőfüggő megközelítés sokkal jobban közelíti a valós beszélgetések mintázatait számos metrika szerint, mint a korábbiak.

Ezekből az eredményekből kiindulva munkánk célja a beszélőfüggő párbeszéd-szimuláció megvalósítása és kiértékelése magyar nyelvre. Vizsgálatunk a BEA-Large korpuszon (Gedeon és mtsai, 2025) alapul, ami egyszereplős felvételekből áll. Ez ideális alapanyagot biztosít a módszertan alkalmazására: rendelkezésre áll számos különböző magyar beszélő nagyszámú megszólalása, amelyekből realisztikus kétbeszélős párbeszédeket generálhatunk. Az így előállított adathalmazt a BEA-Dialogue korpusszal (Gedeon és mtsai, 2025) együtt használjuk beszédfelismerő modellek tanítására, majd a BEA-Dialogue kiértékelési halmazán hasonlítjuk össze a különböző konfigurációk teljesítményét.

A tanulmányt a használt adatok bemutatásával folytatjuk a 2. fejezetben, amit a módszertan részletezése követ a 3. fejezetben. Aztán a kísérletek bemutatása az eredményekkel együtt következik a 4. fejezetben, majd összegzés zárja a tanulmányt.

2. Adatok

2.1. BEA-Large

A *BEA-Large* a BEA magyar beszélt nyelvi korpusz egy részhalmaza, amely 433 beszélő többnyire spontán beszédét tartalmazza, ezzel megfelelő alapot nyújt a magyar spontán beszédre épülő beszédfelismerő rendszerek kutatásához és összehasonlító értékeléséhez (Mády és mtsai, 2024).

Tanulmányunkban a BEA-Large azon beszélőinek felvételeiből szimuláltunk párbeszédeket, akik nem szerepelnek a BEA-Dialogue validációs és kiértékelési halmazaiban. Így 240 beszélő összesen 58 ezer megszólalását tudtuk felhasználni, körülbelül 70 órás összidővel.

2.2. BEA-Dialogue

A *BEA-Dialogue* a BEA korpusz spontán párbeszédeket tartalmazó része, amely 85 órányi természetes magyar beszélgetést foglal magában. Az adatok beszélőfüggetlen (minden beszélő csak egy halmazban szerepelhet) bontásban érhetők el, és kifejezetten a párbeszéd-alapú beszédfelismerés és a beszélőszegmentálás (diarizáció) kutatásához készültek. A korpuszt eredeti formájában használtuk fel a kísérletekhez, tanítás esetén a szimulált adathalmazt a *BEA-Dialogue* tanító halmazához adtuk, kiértékeléshez pedig csak a *BEA-Dialogue* megfelelő részét használtuk.

3. Módszertan

3.1. Beszélőfüggő párbeszéd-szimuláció

A beszélőfüggő párbeszéd-szimuláció (SASC, *Speaker-Aware Conversation Simulation*) módszer (Gedeon és Mihajlik, 2025a) olyan több-beszélős (tetszőleges számú) beszélgetéseket generál, amelyek időbeli, szerkezeti és akusztikai tulajdonságai valós beszélgetések alapján vannak modellezve. A beszélőváltásoknál tartott szüneteket illetve átfedéseket egy δ változó írja le, ahol $\delta < 0$ átfedést (overlap), $\delta \geq 0$ szünetet jelez, és a negatív tartomány feletti integrál az átfedés valószínűségének, p_{overlap}-nek felel meg (2. ábra).

Hisztogram-alapú megközelítések helyett kernel-alapú sűrűségfüggvény-becslést (KDE, *Kernel Density Estimation*) alkalmazunk, hogy a szünetek eloszlásáról folytonos becsléseket kapjunk. Az időzítésbeli konzisztencia érdekében két eloszlás kerül meghatározásra: $\hat{D}_{=}$ azonos beszélő közötti átlagos szünetekre (amikor nincs beszélőváltás), és $\hat{D}_{\neq}$ különböző beszélők közötti átlagos szünetekre. Minden s beszélő esetén egy kezdeti bázisérték (μ) kerül mintavételezésre a megfelelő eloszlásból, míg az azt követő szünetek egy eltérés (v) hozzáadásával adódnak:

$$\delta_n = \begin{cases} \mu_s^{\text{same}} + v & \text{ha } X_n = X_{n-1},\ v \sim V_{=}, \\ \mu_s^{\text{diff}} + v & \text{ha } X_n \neq X_{n-1},\ v \sim V_{\neq}. \end{cases}$$

Itt $V_{=}$ és $V_{\neq}$ nulla várható értékű beszélőnkénti "eltérés-eloszlások", amelyek szintén az adatok alapján kerültek becslésre.

A beszélőváltás egy elsőrendű (általánosítható n-edik rendűre) Markov-lánccal kerül modellezésre, ahol az átmeneti mátrix P_{turn} adja meg annak valószínűségét, hogy az előző beszélő után ki következik.

Az összes beszélő ugyanabban az akusztikus környezetben helyezkedik el: egy szoba kerül kiválasztásra a rendelkezésre álló RIR-ek (Room Impulse Response, "teremben felvett impulzusválasz") közül, majd minden beszélőhöz egyedi pozíció rendelődik a szobán belül.

A folyamat tömörített változatát egy pszeudokód (Algorithm 1) mutatja.

Algorithm 1 Egyszerűsített beszélőfüggő párbeszéd-szimuláció

N_{spk} beszélő ($\mathcal{S}'$) választása; RIR-ek hozzárendelése (azonos szoba, különböző pozíciók)
Kezdő beszélő kiválasztása X_1
for $n = 1 \dots N_{\text{u}}$ **do**
    if $n > 1$ **then** $X_n \sim P_{\text{turn}}(X_{n-1})$
    $u_n \in U_{X_n}$ megszólalás mintavételezése, RIR hozzáadása $\rightarrow y_n$
    if $n = 1$ **then** $\delta = 0$
    else if $X_n = X_{n-1}$ **then**
        if X_n első szünete **then** $\mu_s^{\text{same}} \sim \hat{D}_{=}$
        $\delta = \mu_s^{\text{same}} + v,\ v \sim V_{=}$
    else
        if X_n első szünete **then** $\mu_s^{\text{diff}} \sim \hat{D}_{\neq}$
        $\delta = \mu_s^{\text{diff}} + v,\ v \sim V_{\neq}$
    y_n hozzáadása a párbeszédhez, δ szünettel

3.2. Statisztikák készítése

A beszélőfüggő párbeszéd-szimulációhoz három fő statisztikára van szükség: *azonos beszélő megszólalásai között tartott szünetek*, *beszélőváltás esetén tartott szünetek*, *beszélőváltás valószínűsége*. Ezek kinyeréséhez két adathalmazt használtunk fel: az angol telefonbeszélgetéseket tartalmazó CallHome korpuszt (Canavan és mtsai, 1997) és a BEA-Dialogue tanítóhalmazát.

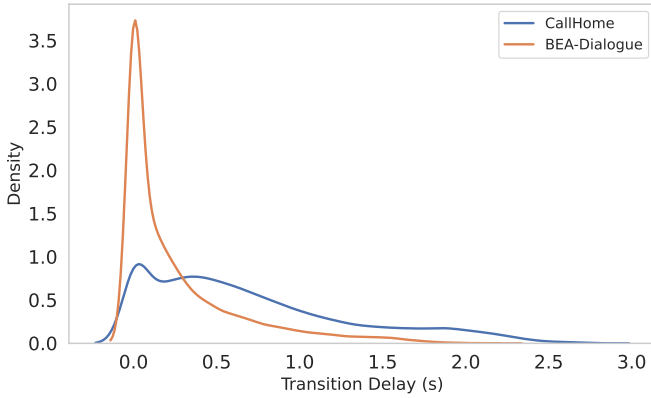
A CallHome esetén egy angol nyelvű szimulált adatbázis létrehozása közben megfigyeltük (Gedeon és Mihajlik, 2025b), hogy a publikusan elérhető annotációk pontatlanok annak tekintetében, hogy az elhangzott szövegek pontosan mikor hangoztak el, ami jelen felhasználáshoz kulcsfontosságú információ. Ezért ehelyett Voice Activity Detection (VAD) modellel újraannotáltuk ezeket és az így kapott adatokat használtuk fel a modellezéshez. Ehhez alkalmazkodva, itt is a Silero VAD (Team, 2024) modelljét használtuk fel erre a célra. Mivel a korpusz sztereóban, mindkét beszélőt külön csatornán van tárolva, így könnyen kinyerhetőek a szükséges statisztikák.

Hogy legyen olyan statisztika is, amit magyar beszédből nyerünk ki, a BEA-Dialogue tanítóhalmazát is felhasználtuk. Az annotációk esetén itt is pontatlanságokat fedeztünk fel, ennek legfőbb példájaként, hogy a szavak/egységek elhangzási időpontjai érintették egymást, tehát az annotáció szerint nem tartottak szünetet a beszélők. A hangfájlok meghallgatása viszont ennek ellenkezőjét mutatta, így itt is az elérhető annotációk felülvizsgálatára volt szükség. Ebben az esetben ezt nehezítette, hogy egy sávra lett rögzítve minden beszélő, ezért nem volt elégséges a VAD használat, többlépéses megközelítésre volt szükség.

Első lépésként, egy szinkronizáló (Montreal Forced Aligner) (McAuliffe és mtsai, 2017) modellel meghatároztuk az elhangzott szavak (leirat alapján) időpontját. Mivel szavak menti határokat kaptunk és a statisztikákhoz nagyobb (pl.

mondatszintű) egységek az ideálisak, ezért egy írásjel-visszaállító modell[1] segítségével igyekeztünk visszafejteni a mondatok határait.

Mindkét adathalmaz esetén a használt eloszlásokat kernel-alapú sűrűségfüggvény-becslés segítségével határoztuk meg, $\alpha = 0,1$ kernel-paraméterrel. Az így kapott eloszlásokat a 1. ábra mutatja az azonos beszélők közti szünetek esetén, 2. ábra pedig a beszélőváltások esetén.



1. ábra: Azonos beszélők szüneteinek modellezése.

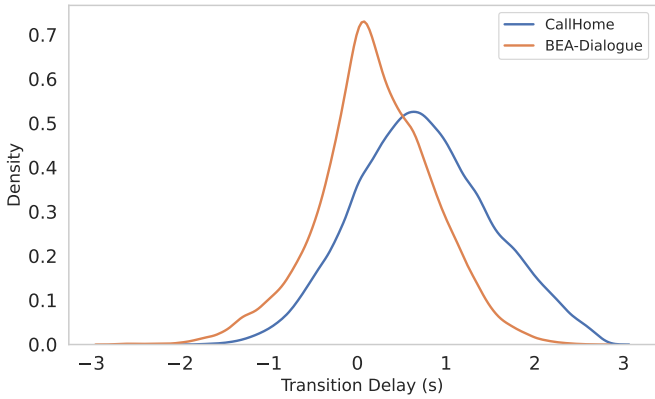
A beszélőváltásokat a CallHome adathalmaz alapján, egy elsőrendű Markov lánccal modelleztük. Az így kapott átmeneti valószínűségeket a 3. ábra illusztrálja.

3.3. Adathalmazok létrehozása

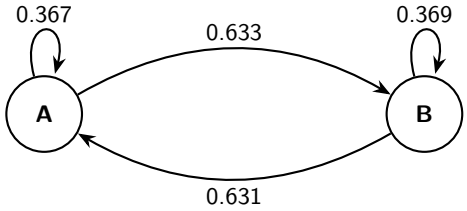
Mint a második fejezetben írtuk, a BEA-Large azon beszélőinek felvételeiből dolgoztunk, akik nincsenek benne a BEA-Dialogue validációs vagy kiértékelő halmazaiban. Ezekből párokat képeztünk, mégpedig úgy, hogy egy adott beszélő pontosan két párosnak legyen tagja, így próbálva egyensúlyozni a minél nagyobb létrehozható adathalmaz és a beszélők túlhasználtsága között.

A mondathosszúsághoz hasonló egységek használata érdekében a felvételeket időtartam alapján szűrtük, és csak a 2–10 másodperc közötti szegmenseket tartottuk meg. Ezt követően minden beszélőpárhoz a lehető leghosszabb szimulált párbeszédet állítottuk elő. A folyamat során egy Markov-lánc segítségével iteratívan határoztuk meg, hogy egy n szegmensből álló beszélgetéshez hány felvétel szükséges az A és B beszélőktől. Az n értékét addig növeltük, amíg meg nem találtuk a legnagyobb olyan számot, amelyhez mindkét beszélőtől elegendő felvétel állt rendelkezésre. Az egyes beszélők felvételeinek eredeti sorrendjét megtartottuk, hogy a szegmensek koherensek maradjanak.

[1] https://huggingface.co/oliverguhr/fullstop-punctuation-multilang-large



2. ábra: Beszélőváltásoknál tartott szünetek modellezése. A negatív értékek rábeszélést jelentenek.



3. ábra: Beszélőváltási modell.

A szimulált adatbázis statisztikái	
Párbeszédek száma	240
Beszélők száma	240
Átlagos szegmensszám (párbeszédenként)	371.33
Átlagos megszólaláshossz (párbeszédenként)	4.37 mp
Átlagos párbeszédhossz	1623.77 mp

A szimulált adathalmazból minden esetben két verziót hoztunk létre. Az egyikben a párbeszédek 40%-ára térszimulációt alkalmaztunk Gedeon és Mihajlik (2025b) módszere alapján, míg a másik egy natúr változat volt. A kísérleteket mindkét verzión elvégeztük az eredmények összehasonlíthatósága érdekében.

4. Beszédfelismerési kísérletek

4.1. Előkészítés

A tanítások és kiértékelés során azonos módszertant alkalmaztunk, mint a BEA-Large cikkjében (Gedeon és mtsai, 2025). Ez a párbeszédek 30 másodperces

szegmensekre történő bontásával kezdődött, majd ezeken belül a beszélőváltásokat egy speciális (`<sc>`) tokennel annotáltuk. Utóbbiakat a tanítások közben is használtuk, így a finomhangolt modellek esetén ezek megtanítása is cél volt.

A modellek tanításához az *NVIDIA NeMo* (Kuchaiev és mtsai, 2019) keretrendszert használtuk. Az összehasonlíthatóság érdekében itt is az eredeti cikk módszertanát követtük és egy Fast Conformer Large CTC (Rekesh és mtsai, 2023) modellt finomhangoltunk az összes konfiguráció esetén. A tanítások egy *NVIDIA RTX 5000 Ada Genereration* GPU-n futottak, 16-os batch mérettel.

4.2. Eredmények

A párbeszédek leiratozásánál előfordulhat, hogy a modellek helyesen ismerik fel az elhangzott szöveget, azonban felcserélik a beszélők szerinti hozzárendelést, például amikor az *A: "Aha."* és *B: "Szuper."* megszólalások részben átfedik egymást, a modell kimenete egyaránt lehet *"Aha. <sc> Szuper."* és *"Szuper. <sc> Aha."* is. A hagyományos WER (Word Error Rate, szószintű hibaarány) és CER (Character Error Rate, karakterszintű hibaarány) metrikák ilyen esetekben csak egy sorrendet kezelnek helyesen, jóllehet a tartalmi felismerés helyes. Ennek kiegyensúlyozására a táblázat a cpWER és cpCER értékeket is tartalmazza, amelyek minden lehetséges megszólalás-permutációra kiszámítják a hibát, és a legkisebb értéket veszik figyelembe, így pontosabb képet adva a modellek tényleges leiratozási teljesítményéről, különösen átfedő beszéd esetén. Emellett az `scAcc` mutató azt jelzi, hogy a modellek a szegmensek hány százalékában azonosították helyesen a beszélőváltások számát. A validációs (dev) halmazon elért eredményeket a 1. táblázat foglalja össze.

A szimulációs konfiguráció azt jelöli, hogy a CallHome vagy a BEA-Dialogue korpuszból kinyert statisztikák szolgáltak-e a párbeszédek szimulációjának alapjául, illetve hogy alkalmaztunk-e térszimulációt (RIR) a felvételek 40%-án, ahogy azt a 3.3 fejezetben leírtuk. A `whisper` a whisper-large-v3[2] modellt jelöli.

Az `fc_base` modell angol súlyokból[3] került inicializálásra (ahogy az összes többi Fast Conformer alapú modell is), majd a BEA-Dialogue tanítóhalmazán lett finomhangolva. A táblázat két referenciamodell eredményeit is tartalmazza, amik esetében a BEA-Dialogue tanítóhalmazát olyan szimulált adathalmazzal egészítettük ki, amihez nem használtuk a kinyert statisztikákat:

- `fc_naivesim`: statisztikák használata helyett fix, 0,25 másodperces szüneteket illesztettünk a megszólalások közé;
- `fc_nosim`: a szimulációhoz kiválasztott megszólalásokat egymástól függetlenül adtuk a modellnek (összefűzés és így `<sc>` token nélkül), így a modell jellemzően rövidebb mintákon és több iteráción keresztül tanult. A BEA-Dialogue tanítóhalmazában megtartottuk az `<sc>` tokeneket, így ez a modell is képes a beszélőváltások jelölésére.

[2] https://huggingface.co/openai/whisper-large-v3
[3] https://huggingface.co/nvidia/stt_en_fastconformer_ctc_large

A `statsim` jelzésű modellek esetében szintén ugyanabból az angol alapmodellből indultunk ki, azonban a BEA-Dialogue tanítóhalmazát (kb. 67 óra) a bemutatott szimulációs eljárással generált adatokkal (kb. 108 óra) egészítettük ki. A *CH+BEA* statisztika a két különböző statisztikával szimulált adathalmazok összességét jelölik, így ott a szimulált halmaz mérete kétszeres.

Approach	**Sim. setup**		**dev**				
	stat	**RIR**	**WER ↓**	**cpWER ↓**	**CER ↓**	**cpCER ↓**	**scAcc ↑**
`whisper`	–	–	21.19	21.04	12.74	12.56	-
`fc_base`	–	–	20.07	19.86	8.27	8.09	69.67
`fc_nosim`	–	✗	17.52	17.34	7.79	7.65	67.07
`fc_naivesim`	–	✗	17.34	17.11	7.25	7.03	68.46
`fc_statsim`	CH	✓	17.20	16.91	7.21	6.98	69.67
`fc_statsim`	CH	✗	16.75	16.54	7.09	6.90	68.63
`fc_statsim`	BEA	✓	16.78	16.57	7.07	6.90	67.42
`fc_statsim`	BEA	✗	16.66	16.48	7.04	6.86	69.50
`fc_statsim`	CH+BEA	✗	**16.41**	**16.20**	**6.99**	**6.79**	**70.19**

1. táblázat. Eredmények a validációs halmazon.

A kiértékelési (eval) halmazon elért eredményeket a 2. táblázat foglalja össze.

Approach	**Sim. setup**		**eval**				
	stat	**RIR**	**WER ↓**	**cpWER ↓**	**CER ↓**	**cpCER ↓**	**scAcc ↑**
`whisper`	–	–	22.21	22.13	12.27	12.18	-
`fc_base`	–	–	21.10	21.01	9.42	9.33	81.95
`fc_nosim`	–	✗	18.41	18.31	8.70	8.59	78.67
`fc_naivesim`	–	✗	18.34	18.24	8.37	8.27	82.58
`fc_statsim`	CH	✓	18.35	18.23	8.36	8.27	82.27
`fc_statsim`	CH	✗	17.93	17.84	8.30	8.22	82.37
`fc_statsim`	BEA	✓	18.12	18.03	8.28	8.19	**82.95**
`fc_statsim`	BEA	✗	17.85	17.76	8.25	8.17	82.90
`fc_statsim`	CH+BEA	✗	**17.46**	**17.35**	**8.11**	**8.02**	82.58

2. táblázat. Eredmények a kiértékelési halmazon.

Amint a táblázatok mutatják, mindkét halmazon jelentős javulást eredményezett a nagyobb mennyiségű tanítóadat bevonása, még akkor is, ha ehhez nem használtunk semmilyen statisztikai információt. Ezekhez képest azonban további javulást hozott a statisztikai modellezés alkalmazása. A térszimuláció hozzáadá-

sa minden esetben enyhe teljesítményromlást okozott, amit magyarázhat, hogy a felvételek stúdiókörnyezetben készültek, így az akusztikai feltételek nem voltak eléggé diverzek ahhoz, hogy az ilyen típusú augmentáció előnyt jelentsen.

A BEA statisztikái alapján szimulált adathalmazok mindkét esetben jobb eredményt mutattak, mint a CallHome-ból származó szimulációk. A legjobb teljesítményt ugyanakkor az adta, amikor a két szimulációs eljárás kombinálásával egyesítettük az adathalmazokat. Tesztjeink során azt is tapasztaltuk, hogy ha az eredeti konfigurációtól eltérve nem két, hanem öt párosban szerepeltetünk minden beszélőt, akkor a modell teljesítménye romlik a két párosos beállításhoz képest. Ez arra utal, hogy a két statisztika alapján végzett szimuláció egyesítése nem pusztán a tanítóadat mennyiségének növekedése miatt kedvező, hanem azért is, mert ilyen módon nagyobb diverzitás kerül a statisztikákba.

A *dev* halmazon az *scAcc* mutató enyhe romlást mutatott, míg az *eval* halmazon vegyes tendenciák figyelhetők meg: egyes konfigurációkban kismértékű javulás, másokban mérsékelt romlás tapasztalható. Külön érdekes, hogy a WER és CER értékek alapján legjobbnak bizonyuló modell ebben a mutatóban nem a legerősebb.

5. Összegzés

Munkánkban a beszélőfüggő párbeszéd-szimuláció magyar nyelvű megvalósítását és hatásának vizsgálatát mutattuk be beszédfelismerő rendszerek teljesítményére. Célunk az volt, hogy a korlátozott magyar párbeszédes adatforrások problémáját enyhítsük olyan, egyszereplős felvételekből generált mesterséges párbeszédek segítségével, amelyek a beszélők egyéni időzítési sajátosságait is megőrzik.

A szimulációs folyamat során a BEA-Large korpusz egyszereplős felvételeit használtuk kiindulási pontként, és mind a CallHome, mind a BEA-Dialogue adathalmazból származó statisztikákat alkalmaztuk a szünetek, beszélőváltások és átfedések modellezésére. A valósághű időzítési mintázatokat kernel-alapú sűrűségfüggvény-becslés és elsőrendű Markov-lánccal határoztuk meg. Az így előállított, 240 párbeszédet tartalmazó szimulált korpusz két változatban készült el: térszimulációval (RIR) és anélkül, hogy a módszer robusztusságát több konfigurációban is vizsgálni lehessen.

Eredményeink azt mutatták, hogy a szimulált párbeszédekkel kibővített tanítóhalmaz következetesen javította a modellek teljesítményét mind a WER/CER, mind pedig a cpWER/cpCER mutatók tekintetében, függetlenül attól, hogy a szimuláció alapjául mely statisztikák szolgáltak. A statisztika-alapú szimuláció további előnyt biztosított a pusztán adatnöveléshez képest, míg a térszimuláció hozzáadása enyhe romlást eredményezett. A BEA-Dialogue alapján generált minták mindkét adathalmazon jobbnak bizonyultak, és a két statisztikai forrás egyesítése hozta a legerősebb eredményeket, ami arra utal, hogy nem csupán az adatmennyiség növelése, hanem a statisztikai diverzitás bővítése is kritikus tényező.

Összességében a vizsgálatok igazolják, hogy a párbeszéd-szimuláció hatékony eszköz a magyar nyelvű több-beszélős beszédfelismerő rendszerek fejlesztésére.

A generált szintetikus adatok érdemben hozzájárulnak a modellek általánosítási képességének javításához, miközben csökkentik a valós párbeszédes korpuszok előállításával járó annotációs költségeket.

Ígéretes jövőbeli kutatási lehetőséget mutat a használt statisztikák mélyebb elemzése, javítása, illetve a módszer megvalósítása több mint két beszélőre.

Köszönetnyilvánítás

A munka a Kulturális és Innovációs Minisztérium Nemzeti Kutatási Fejlesztési és Innovációs alapból nyújtott támogatásával, az EKÖP_KDP-25-1-BME-21 pályázati program finanszírozásában, a 2025-2.1.2-EKÖP-KDP-2025-00005 számú projekt, valamint az NKFIH K143075 és K135038 projektek támogatásával valósult meg.

Hivatkozások

Canavan, A., Graff, D., Zipperlen, G.: CALLHOME American English Speech. Web Download (1997), https://catalog.ldc.upenn.edu/LDC97S42, lDC Catalog No.: LDC97S42, ISBN: 1-58563-111-6, ISLRN: 952-976-147-406-5

Gedeon, M., Barta, P.Z., Mihajlik, P., Gráczi, T.E., Kohári, A., Mády, K.: Toward Conversational Hungarian Speech Recognition: Introducing the BEA-Large and BEA-Dialogue Datasets (2025), https://arxiv.org/abs/2511.13529

Gedeon, M., Mihajlik, P.: From Independence to Interaction: Speaker-Aware Simulation of Multi-Speaker Conversational Timing (2025a), https://arxiv.org/abs/2509.15808

Gedeon, M., Mihajlik, P.: LibriConvo: Simulating Conversations from Read Literature for ASR and Diarization (2025b), https://arxiv.org/abs/2510.23320

Kanda, N., Gaur, Y., Wang, X., Meng, Z., Yoshioka, T.: Serialized Output Training for End-to-End Overlapped Speech Recognition. In: Interspeech (2020), https://api.semanticscholar.org/CorpusID:214714409

Kuchaiev, O., Li, J., Nguyen, H., Hrinchuk, O., Leary, R., Ginsburg, B., Kriman, S., Beliaev, S., Lavrukhin, V., Cook, J., Castonguay, P., Popova, M., Huang, J., Cohen, J.M.: NeMo: a toolkit for building AI applications using Neural Modules (2019), https://arxiv.org/abs/1909.09577

Landini, F., Lozano-Diez, A., Díez, M., Burget, L.: From Simulated Mixtures to Simulated Conversations as Training Data for End-to-End Neural Diarization. In: Interspeech (2022), https://api.semanticscholar.org/CorpusID:247939646

Mády, K., Gráczi, T.E., Kohári, A., Mihajlik, P.: Revised annotation conventions in hungarian speech corpora. Beszédtudomány / Speech Science 4(1), 185–202 (2024)

McAuliffe, M., Socolof, M., Mihuc, S., Wagner, M., Sonderegger, M.: Montreal Forced Aligner: Trainable Text-Speech Alignment Using Kaldi. In: Interspeech 2017. pp. 498–502 (2017)

Rekesh, D., Koluguri, N.R., Kriman, S., Majumdar, S., Noroozi, V., Huang, H., Hrinchuk, O., Puvvada, K., Kumar, A., Balam, J., Ginsburg, B.: Fast Conformer with Linearly Scalable Attention for Efficient Speech Recognition (2023), https://arxiv.org/abs/2305.05084

Team, S.: Silero VAD: pre-trained enterprise-grade Voice Activity Detector (VAD), Number Detector and Language Classifier. https://github.com/snakers4/silero-vad (2024)

Yu, D., Kolbæk, M., Tan, Z.H., Jensen, J.H.: Permutation invariant training of deep models for speaker-independent multi-talker speech separation. ICASSP 2017 pp. 241–245 (2016), https://api.semanticscholar.org/CorpusID:7331600

Ágens-alapú chatbot architektúra valós idejű vezetői Big Data lekérdezésekhez

Vándor Péter[1] és Csáki Csaba[1]

[1] Budapesti Corvinus Egyetem, Fővám tér 8,
H-1093 Budapest, Magyarország
peter.vandor@stud.uni-corvinus.hu
csaki.csaba@uni-corvinus.hu

Kivonat: A vállalatok, szervezetek működése során a különböző területek – értékesítés, marketing, ügyfélszolgálat, ellátási láncok, IoT-eszközök, felhasználói interakciók – hatalmas mennyiségű adatot generálnak folyamatosan. Az ennek eredményeként létrejövő, sokszor több milliárd rekordot tartalmazó Big Data adattárházak, OLAP kockák, dashboardok, elemzések nehezen áttekinthető és sokszor csak bonyolultan lekérdezhető környezetet teremtenek a vezetők számára, akiknek kulcsfontosságú az azonnali, pontos, valós idejű információ a stratégiai döntések meghozatalához. A nagy nyelvi modellek (LLM) megjelenésével lehetővé vált a természetes nyelvi lekérdezés, de a pontos adatok szolgáltatása és a hallucinációk kiküszöbölése továbbra is kihívást jelent. Erre nyújt megoldást az intelligens ágensre épülő, folyamat-vezérelt chatbot fejlesztés, melynek célja, hogy a vezetők természetes nyelven, technikai nehézségektől mentesen kérdezhessenek és megbízható információkat kapjanak. A bot saját MCP (Model Context Protocol) eszközök biztosításával és széleskörű paraméterezésével lehetővé teszi a valós idejű, részletes adat lekérdezést, grafikon rajzolást és adatelemzést. A komplex lekérdezés-paraméterezés egy kontrollált háttérfolyamat részeként valósul meg. E folyamat a végén egy könnyen értelmezhető, mondatokba foglalt válasszal szolgál. Az információszerzés során az ágens önállóan paraméterezi az adattárház lekérdezéseket, szükség esetén visszakérdez vagy több lekérdezést is futtat. Ez a módszer jelentősen csökkenti az adat lekérdezéshez és értelmezéshez szükséges speciális szakértelmet és elősegíti az adat-vezérelt működést.

Kulcsszavak: LLM, ágens, chatbot, MCP, RAG, Big Data

1 Bevezetés

A nagy nyelvi modellek (LLM) térnyerését követően a mesterséges intelligencia (MI) technológiai fejlesztések egy újabb hulláma is elindult, terjedni kezdtek az intelligens

ágensek és az ezekre épülő megoldások. Habár az OpenAI 2025-öt kikiáltotta az ágensek évének (Altman, 2025), ennek ellenére a technológia üzleti adoptálása lassan halad. Egy szervezeti környezetben kifejezetten erős szempont a pontosság az ágens rendszerek bevezetésekor, mivel a nem megfelelően szabályozott, pontatlanul kontrollált ágensek üzletkritikus hibákat is elkövethetnek.

Az MI ágensek olyan rendszerek, melyek bizonyos fokú autonómiával rendelkeznek, intelligens döntéseket hoznak a rendelkezésre álló adatok alapján egy előre meghatározott cél elérése érdekében és az adott környezettel proaktív módon interakcióba lépnek (Deng et al., 2024).

A legtöbb cég működése során rengeteg adat keletkezik folyamatosan, melyek feldolgozása, gyors és pontos lekérdezhetősége kritikus a felsővezetői stratégiai döntéshozatal szempontjából. A méret, keletkezési sebesség és eltérő formátumok miatt az esetek nagy részében speciális eszközökkel kezelhető Big Data adattárházakról beszélhetünk. Az információk lekérdezését követően elemzés, dashboard, riport vagy más típusú kimenet is készülhet az adatok átlátható összegzésére és a jelentéstartalom megvilágítására. A nyelvi modellek lehetővé teszik egy újfajta interfész kialakítását, amellyel természetes nyelven megfogalmazott kérések is kiszolgálhatóak, ezáltal ledöntve a gyakran adatelemzőkre szabott, nem intuitív kezelőfelületek által jelentett akadályokat a vezetői pozíciókat betöltő személyek számára.

Egy ilyen új típusú felület esetében azonban kritikus a szolgáltatott adatok és válaszok minősége. Az LLM által közvetlenül, az adatforrás részletes ismerete nélkül írt függvények szuboptimális lekérdezési sebességet okoznának. A vállalatok feltehetően már rendelkeznek az adatbázis lekérdezésére szolgáló függvényekkel, kódokkal. Nem ajánlott a nyelvi modellre bízni a lekérdezések megírását a beérkező kérésekre, még az egyre fejlődő kódolási képességeket figyelembe véve sem. Az LLM-alapú modern MI ágensek számára ezeket az interakciókat az elérhető eszközök teszik lehetővé, melyek API-k vagy függvények közvetlen meghívásával kibővítik a lehetőségeket: legyen szó valós idejű keresésről, adatlekérdezésről, számítási módszerekről, vagy bármilyen a feladat megoldását támogató, az ágensnek új adatokat szolgáltató kódról vagy külső alkalmazásról (Sapkota et al., 2025). Mivel az üzleti szférában a megbízhatóság egyértelmű elvárás, ezért jelen kutatás fő célját a pontos válaszok biztosítása jelenti. További cél a tisztán LLM-re alapozó és általános ágens megoldásokhoz képest a hallucinációk elkerülése a valós idejű Big Data lekérdezések területén. Javaslatunk középpontjában az integrált eszközhasználat jelenik meg, mint célra vezető út: a már meglévő szervezeti tudást egy LLM ágens által felhasználható eszközzé konvertálni, ezáltal megőrizve a korábbi determinisztikus folyamatok biztonságát. Lehetővé válik a felhalmozott tudásbázis természetes nyelven történő lekérdezése és a válaszok is így jelenhetnek meg, ezáltal téve közvetlenül is elérhetőbbé az összetett adatbázisok tartalmát a nem-technikai vezetők számára is.

2 Módszertan

Az üzleti folyamatok során felmerülhetnek olyan kérdések, amelyek megoldásához domain-specifikus tudásra van szükség. Minden eshetőséget szinte lehetetlen az

utasítások közt megfogalmazni, a munkavállalók fejében létező tudást kell átadni az ágensnek ilyen szituációkban. A felhasználói beavatkozás az ágensek működésébe az eddigi munkákban ritkán vizsgált terület (Händler, 2023). Ez egyben a valós használat során bővülő memória és tanulási mechanizmus bevezetését jelenti. A tapasztalatokból tanulásra képes ágensek kiváló eredményeket demonstrálnak a szakirodalomban az egyszerű ágensekhez képest a többlépéses döntéshozatali feladatokban (Zhao et al., 2023). Az érvelés és akció összekapcsolásával a modell a környezetből szerzett információkra reflektálva képes terveket készíteni és döntéseket hozni. Egy ezt teljesíteni képes architektúra és folyamat megvalósításához számos meglévő technikai megoldásra építettünk, ezeket tekintjük át ebben a fejezetben.

A vállalati tudásbázis természetes nyelven lekérdezéséhez egy általános ágens architektúra megtervezése és fejlesztése képezte a kutatási projekt magját. A konkrét példát a Magyar Turisztikai Ügynökség (MTÜ) Statisztikai Adattárház jelentette. Ez a több milliárd rekordos, naponta frissülő adattárház elsősorban jól strukturált adatokat tartalmaz, de ezen felül több száz elkészült riport jelenti a nem strukturált adatforrást az ágens számára – mindkét forrástípus kezelése elengedtethetetlen a teljes vállalati tudásbázis kiaknázásához. A naponta változó belső adattárházon túl a havi bontásban publikusan elérhető KSH adatokkal is szükséges dolgozni. Az eredeti lekérdező felület egy komplex megoldás volt, melynek hatékony kezelése mélyebb ismereteket várt el az adattárház felépítéséről és a szélsőséges esetekről. Hibás használata gyakran az adatelemző és fejlesztő kollégák idejébe került, mivel nekik kellett végrehajtani a megfelelő lekérdezést, majd a vezetőségnek elmagyarázni az eredményeket. E felület kiváltása érdekében merült fel egy természetes nyelven alapuló megoldás kidolgozása.

Az itt bemutatott kutatásban az ágens architektúrájának kidolgozása során inspirációt merítettünk a ReAct (Yao et al., 2022) keretrendszer struktúrájából. A kivitelezés központi elemének az Anthropic által kifejlesztett Model Context Protocol (MCP) technológiát választottuk, mely egy nyílt forráskódú standard az AI applikációk külső rendszerekhez csatolásához. Az MCP gyorsan a modell-eszköz kommunikáció standard protokolljává vált (Anthropic, 2024) az elmúlt évben, még a versenytárs LLM fejlesztő cégek is bevezették és adoptálták az új modellagnosztikus protokollt. Az MCP eszközök könnyen, újra felhasználható módon csatlakoztatják az egyedi eszközöket bármely LLM-hez, sőt, akár több nyelvi modellhez is. A külső rendszerek lehetnek adatforrások, eszközök, vagy akár munkafolyamatok. Az MCP-t gyakran hasonlítják az USB-C csatlakozóhoz, mint egyszerű kapcsolódást biztosító megoldás a nyelvi modellek számára (Introduction - Model Context Protocol, 2025). A protokoll egy kliens-szerver architektúrát követ, ahol a kliens megszerzi a kontextust a szervertől. Ebből következik, hogy MCP szervert lokálisan vagy távoli eléréssel is futtathatunk. Az adatcsere egy JSON-RPC 2.0 üzenetstruktúrán keresztül valósul meg (Introduction - Model Context Protocol, 2025).

A legtöbb ágens eszközhasználattal interaktál a környezetével. Azonban az, hogy a nyelvi modell által végrehajtott tervezési lépésben meddig terjedhet az autonóm döntéshozatal, az adott ágens architektúrától függően változhat (Händler, 2023). A hatékonyságon túl átláthatósági és biztonsági szempontokból is fontos a kódból kezelt és a nyelvi modell által kezelt feladatrészek elkülönítése.

Az LLM-ek feladatának meghatározásához fontos technikai elem a rendszer prompt, ami egy átfogó utasításhalmaz, mely meghatározza az MI ágens viselkedését

valamennyi LLM interakció során. Az ágens-alapú rendszerek feladat- és hatáskörét a rendszer prompt szabja meg, ezen promptok ismerete jelentős információkat árul el az adott ágens működéséről (GitHub, 2025). A rendszerutasítások általában egyszer kerülnek beállításra, és változatlanok maradnak, konzisztenciát biztosítva az MI ágens általános szerepében, feladatkörében. Az alapvető működés szempontjából a rendszer promptban megfogalmazott utasítások hatása erősebb a felhasználói promptolásnál.

Az LLM-et irányító instrukciók pontos és részletekbe nyúló megfogalmazása kritikus pontja az ágens létrehozásának. A kontextus a legfontosabb az LLM-ek válaszminőségét tekintve, ezt támasztja alá a context engineering megjelenése az egyszerű promptoláson túl (Mei et al., 2025). További lehetőség az ún. docstring-ek használata, ami a forráskód adott szegmensének működését dokumentáló szöveg. Az ezekben megfogalmazott leírások a függvény funkcióját, a paraméterek lehetséges értékeit, vagy túl nagy értékkészlet esetén a keresési módszert, továbbá a default értékeket, a formátumokat és ha-akkor felépítésű szabályokat is tartalmaznak. A függvényekből a FastMCP könyvtár eszköz definíciókat készít, melyek legfontosabb része a függvényleírás tartalma.

Az ágenshez beérkező kérések komplexitásának elemzésének egy praktikus módja a feladatok lépésszámának meghatározása, mivel ez határozza meg a válaszidőt és növeli a hibázási lehetőséget is.

A tesztelés alatt zárt és lokálisan futtatható nyílt modelleket is alkalmaztunk: GPT-5 és GPT-5 Mini (OpenAI, 2025), Gemini-2.5-flash és Gemini-2.5-flash-lite (Comanici et al., 2025), Llama 4 Maverick (Meta, 2025), Qwen3 80B és Qwen3 235B (Yang et al., 2025), valamint gpt-oss-20b és gpt-oss-120b (Agarwal et al., 2025).

3 Az ágens-alapú chatbot architektúra és működése

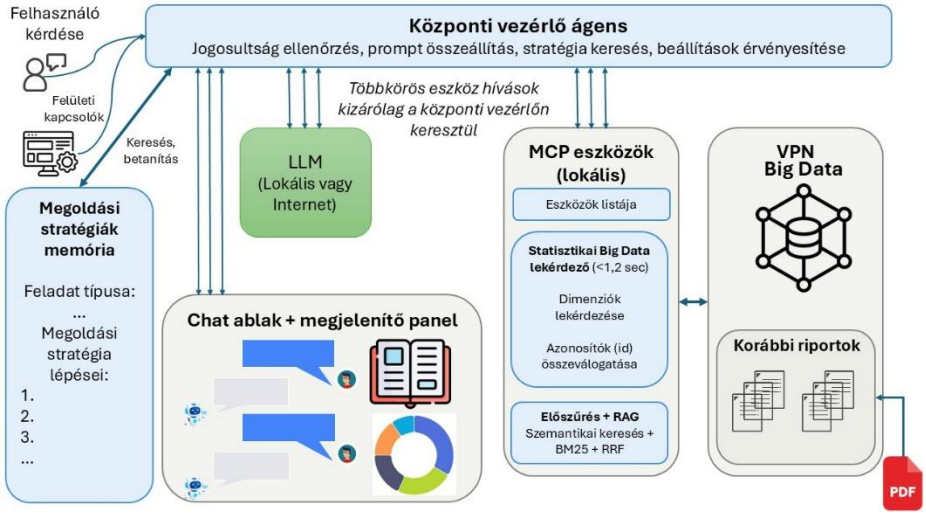
A vezetői kommunikációs kihívás megoldására a javasolt megoldás egy intuitív chatbot felület és egy dedikált ágens architektúra. A saját ágens-alapú chatbot architektúra tervezésekor két fő irányelvet vezettünk be és alkalmaztunk a hallucinációk csökkentése céljából: a *hibrid vezérlés* és a *folytatólagos prompttanulás* elvét. Ezeket az elveket a módszertani részben bemutatott technikai megoldásokra épített sajátságos architektúrában valósítottuk meg. A következőkben a tervezési fázisban megfogalmazott, fejlesztést irányító elveket, majd az ennek hatására kidolgozott rendszer architektúrát mutatjuk be, külön kiemelve a működés folyamatát. A fejlesztés első fázisában optimalizációs teszteket futtattunk a tényleges alkalmazás előtt.

A hibrid vezérlés a determinisztikusan, algoritmussal megoldható feladatrészeket továbbra is explicit kódból kezeli, míg a nyelvi modell hatáskörét leszűkíti annak általános nyelvi képességeit kihasználó feladatkörökre. Utóbbihoz tartozik a felhasználói kérdés jelentéstartalmának dekódolása és a megfelelő lekérdezési eszköz (függvény) kiválasztása és felparaméterezése, valamint a válasz megfogalmazása és összefoglaló készítése. Már a folyamat elején a felhasználó kezébe adjuk a vezérlést az elérési felületen állítható kapcsolókkal, amelyek determinisztikusan, algoritmus segítségével megváltoztatják a lekérdezések paraméterezését (pl. számítási mód kiválasztása) vagy egy más lefutási ágba terelik az ágenst (pl. riportokban keresés). Ez

a hibrid megközelítés teret ad az algoritmikus működésnek az LLM előnyeinek kihasználása mellett. A hibrid vezérlés fő kihívása az egyensúly megtalálása a jól definiált algoritmusok és a széleskörű feladatvégzésre alkalmas nyelvi modellek között, ahol az optimális működés elérése az adott feladat igényeitől függ.

A folytatólagos prompttanulás elve azt jelenti, hogy az egyes kérdéstípusoknál előforduló nem determinisztikus működést a felhasználó képes korrigálni általános megoldási stratégiák leírásával. Ezek beépülnek az ágens belső tudásbázisába, és minden új beérkező kérdésnél egy RAG (Retrieval Augmented Generation) keresést hajt végre az ágens a tudásbázis szövegében, hogy csak az adott feladattípushoz tartozó releváns kontextust töltse be a modell rendelkezésére álló információk közé.

A két elvet leképező saját architektúránk többszöri eszközhívásokra, akciókra, majd ezek kiértékelésére, reflektálásra épül. A kiértékelés során az ágens hozza meg a döntést a rendszer prompt, a docstring tartalma, a felhasználói prompt, az eddigi csevegést eltároló memória és az akciók segítségével begyűjtött adatok segítségével, hogy készen áll-e a végleges kimeneti válasz megfogalmazására.



1. ábra: A Big Data ágens architektúrája (saját munka)

Az ágens-alapú chatbot architektúra hat fő komponensből tevődik össze (lásd 1. ábra): (1) a központi vezérlő ágens, (2) az LLM (lokális vagy API használaton keresztül), (3) az MCP eszközök, (4) a Big Data adatbázis, (5) a megoldási stratégiákat tartalmazó tudásbázis és (6) a felhasználói felület, ami áll egy chat ablakból és egy megjelenítő panelből. A központi vezérlő ágens irányítja a megoldási munkafolyamat menetét, mely függ a beérkező kérdéstől és a felületen aktív beállításoktól. Először ellenőrzi a felhasználói jogosultságokat, majd, ha ezek megfelelőnek bizonyultak, érvényesíti a felhasználó által alkalmazott beállításokat és a beérkezett kérdést feldolgozva lekérdezi a tudásbázisban tárolt stratégiákat. Ezt követően a kiválasztott LLM számára listázza az eszközöket és megjeleníti a stratégiákat. Az LLM ekkor vagy

rögtön válaszol (ritka), vagy egy MCP eszközt választva felparaméterez egy függvényt, és visszakéri annak az eredményét. Az eredmény alapján döntést hoz, majd további lekérdezéseket futtat, ha nem elegendő az információ, vagy visszatér a végső válasszal a felhasználó számára, ha már elegendő információt gyűjtött össze. Érdemes érvelő modellt alkalmazni, mely képes levezetni, hogy a kapott eredmény megfelelő-e a kérdés megválaszolására.

Fontos megjegyezni, hogy nem az LLM hívja direktben az MCP eszközöket, hanem miután megkapta az eszközlistát, és kiválasztotta, felparaméterezte a megfelelő függvényt, akkor visszatér a függvényhívással, amit a vezérlő ágens hív meg lokálisan, és ezen a ponton algoritmikusan közbe tud avatkozni új paraméterek átadásával, meglévők megváltoztatásával vagy a teljes vezérlés eltérítésével.

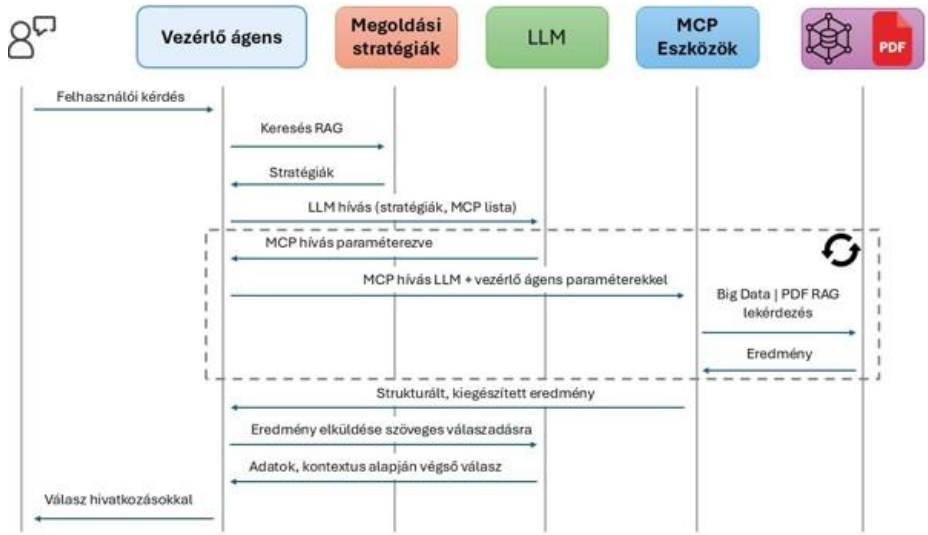
A Big Data lekérdező ágensben a rendszer prompt tartalma a legerősebb vezérlő tényező az LLM számára, hiszen ez minden interakció során érvényes. Az utána következő felhasználói kérésekben, utasításokban hiába próbáljuk felülírni a rendszer promptban megfogalmazott iránymutatásokat, az LLM tesztjeink alapján ragaszkodik azok betartásához. Hangsúlyos figyelmeztetésként fogalmaztuk meg az LLM számára, hogy (1) nem térhet el a tárgytól a felhasználó kifejezett kérésére sem; (2) mindenképpen valamilyen eszközt kell használnia a válaszadáshoz, amihez megadható az eszközök listája; (3) ha hiányos az információ a lekérdezés paraméterezéséhez, akkor mindenképpen kérdezzen vissza a konkrét hiányzó paraméterek megnevezésével; és (4) amennyiben nincs olyan eszköze vagy nem tudja felparaméterezni a felhasználó kérdésének megfelelően, akkor ezt jelezze a válaszában és ne találjon ki számokat. Már az első teszek bizonyították, hogy a jól megfogalmazott rendszerutasítás jelentősen csökkentette a pontatlan válaszok arányát és a visszakérdezés sok esetben több lépéses, komplex feladatmegoldást eredményezett. A csevegési előzmények elküldése tovább erősíti a természetes nyelvi kommunikációt és ebből következően az eredmények pontosságát, mivel általában az egyik kérdésből következik a másik (pl. az előző eredmény további szempontok szerinti bontása) és az ember adottnak tekinti a korábbi kérdés-válaszokban definiált paramétereket.

A Big Data lekérdezés (a központi, legtöbbet használt függvény) esetén általában 20+ dimenzió és akár 100+ értékmező is megadható, melyek példánkban tipikusan idő és területi behatárolások, és csoportosítások, melyekre az aggregálás jellemzően több milliárd rekordon fut le. Az LLM számára részletesen és minél pontosabban el kell magyarázni az egyes dimenziók lehetséges értékeit és használati módjukat. A kisebb számosságú attribútumok esetében javasolt, hogy az MCP eszköz leírása tartalmazza az összes választható értéket és ezt külön jelezzük is az LLM felé, hogy csak ezen értékek közül választhat. A nagyobb (pl. 1000+) elemszám esetén a token-alapú árazás és a túl hosszú kontextus miatt érdemes csak röviden leírni az értékkészletet, néhány példát adni a sokszor előforduló esetekre, valamint egy külön MCP segédfüggvényt biztosítani a dimenzió értékeinek keresésére vonatkozóan. Ha az LLM magabiztosan rosszul paraméterez, akkor az MCP lekérdező függvény mindjárt a legelején kiszűri a nem megfelelő értékeket, a válaszban jelzi a hibát, és az LLM figyelmébe ajánlja a dimenzió kereshetőségét biztosító segédfüggvényt. Tesztjeink szerint több esetben is ez vezetett el egyes feladatok megoldásához.

Előfordulnak olyan dimenziók, ahol egy összetett attribútum struktúrával rendelkező entitást kell beazonosítani és erre általában az entitás neve önmagában nem elegendő

(pl. személynevek azonossága). A pontos eredmény elérése érdekében ekkor egy olyan eljárást kell az LLM számára definiálni, amely az entitás jellemzőivel paraméterezhető és a visszatérési értékében rekordonként egy egyértelmű egyedi (adatbázis) azonosítót tartalmaz. Ha az egyedi id-k összegyűjtésre kerültek, akkor már a fő Big Data lekérdező precízen felparaméterezhető és az eredmény is a felhasználó kérésének megfelelően pontos lesz.

Amennyiben két fogalom között egyértelmű megfeleltetés végezhető és csak az egyikkel paraméterezhető az MCP, akkor az LLM számára néhány példával ez az összefüggés megvilágítható és mellette a transzformációt elvégző program is megadható (pl. nemzetiség – országnév).



2. ábra: A Big Data ágens kommunikációs diagramja

Két fő lefutási ága van a kérdéseknek (lásd 2. ábra, elágazás): ezek az adatbázisra és a PPTX formátumú riportokra vonatkozó kérdések mentén különülnek el. Míg az előbbi esetben csak egy-két beállítás befolyásolhatja a folyamat lefutását, addig az utóbbi esetben mindig előszűrést alkalmaz az ágens a megfelelő riport kiválasztására a riportra vonatkozó fő dimenziók (metaadatok) és összefoglaló szerint. Az előszűrés után zajlik le az adott szöveges dokumentumban a részletes keresés: a RAG rendszer szemantikai és a BM25 keresést RRF újra rangsorolással egyesítő komplex keresési folyamata.

4 A belső tudásbázis építése: folytatólagos tanulás

A döntéstámogató rendszerekben alkalmazott LLM-alapú megoldások esetében gyakran előfordulnak ismétlődő hibák, mivel nincsen elég kontextusa a modellnek az adott üzleti helyzetről (Cheung, 2024). A folytatólagosan tanítható belső tudásbázis,

amit egyfajta tartós memóriának is tekinthetünk, lehetővé teszi, hogy a rendszer az elkövetett hibákból általános következtetéseket vonjon le a feladattípusok megoldásával kapcsolatban és a jövőben ezeket hasznosítsa a hasonló kérdések megoldása során. A kontextuson belüli tanulás könnyen áttekinthetővé és ellenőrizhetővé teszi az ágens memóriáját, ezáltal egy transzparens és követhető tanulási folyamatot biztosítva.

E megközelítés elválasztja egymástól az LLM általános nyelvi tudását és az adott szervezetnél előforduló, domain-specifikus műveleti szabályokat. Annak eldöntése, hogy mely feladattípusokhoz szükséges megoldási stratégiát hozzárendelni, nem triviális feladat, amelyet elsősorban a rendszer felhasználói tudnak helyesen mérlegelni. A nyelvi modellre bízni ezt a döntést csak további bizonytalanságot vezetne be, valamint, ha a felhasználó tisztában van a helyes megoldással, rengeteg felesleges próbálkozástól szabadítjuk meg az ágenst. A felhasználó által biztosított helyes megoldás úgy kerül eltárolásra, hogy nem maga a válasz szövege, hanem a feladat típusa és a hozzá tartozó megoldási stratégia legyen visszakereshető és alkalmazható.

Az egyszerű memóriához képest itt általános megoldási helyzeteket definiálunk, melyek a feladatok széles körét lefedik és a szükséges megoldási stratégia is ennek megfelelően kerül megfogalmazásra. Tehát egyfajta típusfeladatok megoldási útmutatóját nyújtjuk át a rendszernek, hogy determinisztikussá tegyük az eredményeket. Tesztjeink alapján ugyanis volt több olyan feladat, amikor a paraméterezést elrontotta a rendszer, majd rájött a hibára és onnan kezdve véletlenszerű eloszlásban három stratégiát alkalmazott: (1) meghívta újra, immár helyes paraméterezéssel a Big Data lekérdezőt; (2) belenyugodott az eredménybe és közölte a felhasználóval, hogy ez így sikerült; (3) teljesen rossz paraméterezéssel hívta újra a lekérdezést.

A többlépésben lekérdezhető komplex problémák megoldása kihívást jelent a mai legfejlettebb LLM számára is. Ilyen esetekben a lefutás nem determinisztikus, több úton is elindulhat a nyelvi modell. Ennek a bizonytalanságnak a kezelésére alkalmas a belső tudásbázis, szerepét a következőkben egy konkrét példával illusztráljuk.

A turisztikai adathalmazban az első ’n’ külföldi küldőországok lekérdezésekor a kezdeti eredmény első helyen mindig Magyarországot tartalmazza. Ez egy döntési pontot jelent, mivel így csak n-1 külföldi ország szerepel az eredményben. Az LLM az esetek egy részében n-1 országot sorol fel a felhasználónak, másik részében új lekérdezést indít n+m elemre, ahol m jellemzően 5 és így már vissza tudja adni a kért n elemet. Az első megoldás hibás, míg a második token- és időpazarlással jár. A feladattípushoz megoldási stratégia definiálásával mindig egy lépésben n+1 elemet kér le a modell, ezáltal 100%-ban pontos lekérdezések valósulnak meg ennél a kérdésnél.

Az elemzők rendszeresen készítenek bizonyos időszakokra és területekre vonatkozó, grafikonokat is tartalmazó jelentéseket, melyek azonos (esetünkben turisztikai) mutatókkal dolgoznak. Ezek a riportok szemantikailag nagyon hasonlók, mert ugyanazon kulcsfontosságú mutatókkal készülnek, ezért a keresési eredményeket nehéz rangsorolni. Mivel több száz nagy hasonlóságú riport áll rendelkezésre, ezért kellett egy szűrési lépést (szabályt) bevezetni a metaadatok és a generált összefoglaló alapján, és az összes dokumentumban keresés helyett riportonként keresni a megfelelő adatokat.

A vállalati elemző riportok esetében a korábban már kifejlesztett saját RAG (Vándor és Csáki, 2025) algoritmusunkat kívántuk használni, de felmerült az a probléma, hogy

az azonos jellegű riportok szövegezésben alig különböznek, csak a számok változnak, mert például egy másik időszakra vonatkoznak. Ebben az esetben a releváns szövegrészeket megtaláló szemantikai kereső nagyon sok, értelmezésüket tekintve a felhasználói kérdésre tökéletesen válaszoló szövegrészt talál eltérő értékekkel. A felhasználó sok esetben nem definiálja az időszakot, hanem egyszerűen a legfrissebb adatokra kíváncsi. Ennek következtében nem alkalmazható egy lépésben a RAG algoritmus, hanem szükségessé vált bevezetni egy riport előválasztó lépést. A dokumentumok feldolgozása során LLM-el általános, a riport típusára vonatkozó összefoglalót generáltatunk az időszakok megemlítése nélkül, valamint az egyes riportokra vonatkozó metaadatok (terület, időszak, típus) is kitöltésre kerülnek. A felhasználó kérdése alapján az LLM listázza a releváns riport típusokat és az azon belül rendelkezésre álló riportok legfontosabb adatait, majd kiválasztja a megfelelő dokumentum azonosítókat és ezekkel hívja meg a RAG alapú rendszert. A RAG MCP eszköz egy saját LLM hívás segítségével megfogalmazza a választ, amihez algoritmikusan hozzákapcsoljuk a három legrelevánsabb szövegrészlet (chunk) hivatkozását, melyeket a központi vezérlő ágens azonnal továbbít a felhasználó felé és nem futtat keresztül a többi feladathoz használt LLM-en az utolsó lépésben. A felhasználó pedig megtekintheti a riport hivatkozott oldalát.

5 Grafikonrajzolás és adatelemzés

Adatbázis lekérdezések eredményeinek döntési felhasználása esetén igen fontos funkció az adatok szemléletes megjelenítése és elemzése. Ezért az ágens rendszer felhasználói lekérdezésekre adott válaszokat nem csupán szöveges formában, hanem interaktív grafikonokon keresztül is megjelenítheti, ezzel elősegítve a döntéshozók gyors és intuitív információ-feldolgozását. A grafikus megjelenítés célja, hogy a nagyméretű, komplex adathalmazokból kiemelje a releváns trendeket, eltéréseket és összefüggéseket, ezáltal támogatva a valós idejű vezetői döntéshozatalt.

A feladatmegoldáshoz hasonlóan a grafikonrajzolás során is rögzítésre kerülnek a legfontosabb paraméterek: csak Echarts oszlop- és kördiagramok készítése engedélyezett az előre definiált hivatalos színpalettával. Az eszköz azonban rugalmas, mivel lehetővé teszi az egyszerű diagramokon túl a többdimenziós összehasonlításokat is (pl. időszakok vagy kategóriák direkt összehasonlítása).

A kötelező paraméterek biztosítják az alapvető funkcionalitást, míg az opcionális beállítások a testreszabhatóságot. A diagramok címének, feliratainak generálása, formátumának beállítása, elhelyezése már az ágens feladata. Az értékek megjeleníthetők az oszlopokon belül vagy kívül, a kördiagramoknál pedig formázási lehetőségek állnak rendelkezésre, amelyek megjeleníthetik a neveket, értékeket és százalékos arányokat egyaránt. A tooltip és jelmagyarázat elhelyezését is ágens kezeli.

Az eszköz a kimenetét speciális formátumban adja vissza: egy JSON alapú konfigurációs objektum tartalmazza a teljes diagram specifikációt, mely közvetlenül értelmezhető a frontend komponensek által és a diagram azonnal megjelenik a felhasználói felületen.

6 Lekérdezési eredmények

A feladatok komplexitását vizsgálva négy szintet állapítunk meg. A feladatok között egyszerűnek tekintjük az egy lépésben lekérdezhető és megválaszolható kérdéseket, amelyeknél az első eszközhívás után képes válaszolni az ágens, az eszközválasztást nem számítva plusz lépésnek. A bonyolultság következő foka, amikor valamelyik MCP segédfüggvényt kell igénybe venni először a helyes paraméterek beállításához és a végső lekérdezéshez. Majd a három vagy több, de nem nagy számosságú lépésben megoldható feladatok következnek, amelyek esetén a végső eredményhez nem elég egy jól paraméterezett Big Data lekérdezés, hanem több aggregációs lekérdezés eredményét kell kombinálni megfelelően. Egy alternatív példa a riport generálása mintaszövegbe beillesztéssel, ahol a megfelelő számok kinyerése és beillesztése többkörös lefutást eredményez. Komplex feladatok esetén (mint például az aggregált célérték keresése egy nagy számosságú tartományban) pedig előfordult log2(N) lépést igénylő folyamat is. Megjegyezzük, hogy ez utóbbi esetben a paraméter értéktartomány többszöri felezésével érhető el a megoldás. Az első és az utolsó szint ritkán fordul elő, a legtöbb feladat két-három lépést igényel.

Mind a strukturált adatbázis lekérdezések, mind a nem strukturált riportokban keresés tesztelésre került több nyelvi modellel. A 28 egyre nehezedő lekérdező feladatban az egyszerű, egyetlen szűrést igénylő kérésektől (pl. „Hány turista járt Baján 2025-ben?”) az időszakösszehasonlításon és százalékszámításon át egészen a többlépéses grafikonrajzolásig (pl. „Készíts kördiagramot 2025-ben a legtöbb turistát fogadó településekről. Az első 7 település mellett jeleníts meg egy egyéb kategóriát.”) terjedt. A visszautaló kérdésekkel egy csevegésen belül zajlott a tesztelés, valós kérdés-válasz folyamot szimulálva.

A kiválasztott modellek között egyaránt szerepel lokális, nyílt (gpt-oss, Qwen, Llama) modell és API-n elérhető, zárt (GPT, Gemini) modell is. A távolról elérhető modellek leginkább a teljesítmény mérése és az összehasonlítás szempontjából kerültek be, a gyakorlati megoldás esetében csak lokális szerveren (L40 kártyákkal) futtatott modellek jöhetnek szóba. Az adatbázisban csak anonimizált hash található, személyes adatok nem kerültek ki a tesztelés során.

1. táblázat: Modellek teljesítménye 14 adatbázis lekérdezés feladaton

Modell	Eredmény	Minőség
GPT-5	100,0%	visszakérdez (+), grafikon szerkezete furcsa (-)
GPT-5 mini	96,4%	1 paraméter kihagyása
GPT-4o	82,1%	összehasonlítás (-), 1 paraméter kihagyása, komplex megoldási stratégia (-)
gpt-oss-120b	32,1%	paraméterek pontosan definiált neveit sem adja meg jól
gpt-oss-20b	32,1%	paraméterek pontosan definiált neveit sem adja meg jól
Qwen3-235B	71,4%	összehasonlítás (-), grafikonrajzolás (-), megoldási stratégia (-)
Llama-4-Maverick-17Bx128E	42,9%	grafikon készítés kérés nélkül, rossz kimeneti mező választás, végtelen ciklus

A gyakorlati tapasztalatok azt mutatták, hogy a GPT-5 és mini modelleket alkalmazva magas pontosság érhető el. Azonban nem várt probléma, hogy a modellek egyes paraméterek behelyettesítését ritkán végezték el helyesen, pl. a szálloda típus kiválasztása is ezek közé tartozik a „Sorold fel az 5*-os szállodákat…” típusú kérdéseknél.

A modellek teljesítményét vizsgálva kizárólag a GPT-5 tudta megválaszolni az összes kérdést, mindezt úgy, hogy visszakérdezett a bizonytalan megfogalmazásoknál (pl. szoba vagy férőhely kapacitás). A kisebb GPT-5 mini esetében már megfigyelhető paramétertévesztés. A 4o nem érvelő modell, ezért az összehasonlítási feladatban hibázott és az utolsó, „egyéb” kategória létrehozására vonatkozó megoldási startégiát sem tudta sikeresen alkalmazni. A lokális modellek közül a gpt-oss modellek mérettől függetlenül súlyos hibákat vétettek, legtöbbször a paraméterek pontosan definiált neveit sem tudták helyesen behelyettesíteni a függvénybe. Az OpenAI nyílt modelljei nem rendelkeznek a GPT-5-höz hasonlítható minőségű eszközhívás betanítással, ez látszik az eredményeken. A Qwen3 összességében a legjobb nyílt modell lett, de számos problémával rendelkezik, mint a hibás grafikonrajzolás és a megoldási stratégiák nem megfelelő kihasználása. A Llama 4 modell váratlan hibákat vétett, két alkalommal grafikont készített kérés nélkül, egy kérdésnél változtatás nélkül, végtelen ciklusban újra lekérte az adatokat, és hallucinációt produkált az utolsó kérdésnél. A teljes kérdéssorozatban ez az egy tipikus hallucinációs hiba, az összes többi hibatípus az utasítások, stratégiák nem megfelelő értelmezéséből ered.

2. táblázat: Modellek teljesítménye 6 riportokban keresés feladaton

Modell	Eredmény	Minőségi hibák	Idő (sec)
GPT-5	100%	felesleges információk	52
GPT-5 mini	100%	felesleges információk	66
GPT-4o	67%	2 hiányzó adat; táblázat értelmezése	19
gpt-oss-120b	100%	legjobb megfogalmazás és minőség	44
gpt-oss-20b	67%	2 hiányzó adat; táblázat értelmezése	17
Qwen3-80B	50%	3 hiányzó adat; 1 db fájl kizárása; hivatkozás blokk rossz	28
Qwen3-235B	67%	2 hiányzó adat; táblázat értelmezése	42
Llama-4-Maverick-17Bx128E	0%	összes PDF kizárása	17
Gemini-2.5-flash	83%	1 hiányzó adat	28
Gemini-2.5-flash-lite	67%	2 hiányzó adat; válasz nem koherens	14

A riportokban keresés a RAG módszerrel kapott eredményeket mutatja meg, 6 adat lekérdezésével tesztelve. A GPT-5 és GPT-5 mini modellek bár hibátlanul vizsgáztak, a vártnál több tokent használtak el és felesleges információkat is közöltek a válaszokban. A 4o két esetben nem találta meg az adatokat, beleértve egy táblázat közepén található adatpontot. A gpt-oss-120b 100%-os eredményt produkált, ráadásul

a legjobb megfogalmazással és minőséggel, felesleges információk nélkül. A 20 milliárdos, kisebb oss modell a 4o-val egyenértékű válaszokat nyújtott, a táblázat értelmezése ennek a modellnek is kihívást jelent. A nagyobb Qwen3 is 2 hibával, míg a kisebb már 3 hibával szerepelt – utóbbi az egyik adatpontot tartalmazó fájlt kizárta és másik riportban keresett. A Llama 4 Maverick nem megfelelő dokumentumokban végezte az információkeresést.

Az 1 perc körüli feldolgozási idők a GPT-5 modelleknél a válasz előtti érvelési folyamatnak tudhatók be, éppen ezért szignifikánsan gyorsabb a 4o modell 20 másodperc alatti ideje. A nyílt, lokálisan futtatható modelleknél jelentősen függ a sebesség a paraméterszámtól: 100 milliárd felett több, mint 40 másodperc, míg 20 milliárd aktív paraméter alatt kevesebb, mint 20 másodperc a mért válaszidő. A kifejezetten sebességre optimalizált Gemini-2.5-flash modellek a válaszadás felgyorsítása céljából kerültek tesztelésre. A 2.5-flash-lite modell a legjobb feldolgozási időt érte el (14 sec), de a válasz minősége nem elegendő a gyakorlati alkalmazáshoz: a 2 hiányzó adaton túl a válasz koherenciája sem érte el a kívánt szintet.

7 Tanulságok összefoglalása

A valós üzleti körülmények között pontos és hallucinációmentes válaszokat szolgáltató, természetes nyelven lekérdezhető ágens-alapú chatbot rendszer a szervezeti tudásbázis újszerű, de megbízható kiaknázását teszi lehetővé. A válaszok nemcsak reprodukálták a manuálisan lekérdezhető eredményeket, de validálhatóak is a függvénybe beillesztett paraméterek segítségével. Emellett a legtöbb rossz lekérdezés felismerhető a hibás felparaméterezésből, így erre a későbbiekben akár egy korrekciós módszer is építhető.

Az adatbázis lekérdezés feladatokban csak a GPT-5 szerzett 100%-os eredményt. A nyílt modelleknek kihívást jelentett a feladatsor, a gpt-oss modellek esetén a rendszer prompt rövidítése vagy angol utasítások megadása orvosolhatja a gyengébb eredményeket. A nagyobb paraméterszámú modellek általában jobb teljesítményt nyújtottak. A riportokban történő keresési feladatoknál a GPT-5 modellek hibátlan eredményeket értek el, míg a gpt-oss-120b modell a legjobb válaszminőséget nyújtotta felesleges információk nélkül. A feldolgozási idők elemzése kimutatta, hogy a modellméret és a válaszadási sebesség között szoros, negatív irányú összefüggés található: a nagyobb modellektől szignifikánsan lassabb válasz várható.

A feladattípusokhoz megfogalmazott megoldási stratégiák beépítésével az ágens a nem egyértelműen elvégezhető kérések kezelésére is megtanítható. A visszakérdezés fontos pontja a bizonytalanság redukálásának. Az LLM-ek rugalmassága és a determinisztikus folyamatok biztonsága teszi lehetővé a nagyfokú autonómiát.

Az itt bemutatott megoldás új eszközt kínál a Big Data közvetlen vezetői hozzáférhetőségére a szervezeten belül és elősegíti a pontos, adatvezérelt döntéshozatalt. Bár egy adott, konkrét esethez készült ágens architektúra került bemutatásra, a megoldás más adatbázishoz is hozzákapcsolható, és annak sajátságaira átszabható a lekérdező eszközök kiválasztásával és illesztésével.

Bibliográfia

Agarwal, S., Ahmad, L., Ai, J., Altman, S., Applebaum, A., Arbus, E., ... & Zhao, S. (2025). gpt-oss-120b & gpt-oss-20b model card. arXiv preprint arXiv:2508.10925.

OpenAI. (2025, August 13). GPT-5 system card. OpenAI. https://cdn.openai.com/gpt-5-system-card.pdf

Altman, S. (2025, January 6). *Reflections*. Sam Altman. https://blog.samaltman.com/reflections

Anthropic. (2024). *Introducing the Model Context Protocol*. Anthropic.com. https://www.anthropic.com/news/model-context-protocol

Cheung, M. (2024). A Reality check of the benefits of LLM in business. *arXiv preprint arXiv:2406.10249*.

Comanici, G., Bieber, E., Schaekermann, M., Pasupat, I., Sachdeva, N., Dhillon, I., ... & Mehta, S. V. (2025). Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. arXiv preprint arXiv:2507.06261.

Dong, Z., Guo, Y., Han, C., Ma, W., Xiong, J., Wen, S., & Xiang, Y. (2024). AI Agents Under Threat: A Survey of Key Security Challenges and Future Pathways. *ACM Computing Surveys, 57*, 1 - 36.

GitHub. System prompts and models of AI tools. (2025). GitHub - x1xhlol/system-prompts-and-models-of-ai-tools: FULL Augment Code, Claude Code, Cluely, CodeBuddy, Comet, Cursor, Devin AI, Junie, Kiro, Leap.new, Lovable, Manus Agent Tools, NotionAI, Orchids.app, Perplexity, Poke, Qoder, Replit, Same.dev, Trae, Traycer AI, VSCode Agent, Warp.dev, Windsurf, Xcode, Z.ai Code, dia & v0. (And other Open Sourced) System Prompts, Internal Tools & AI Models. https://github.com/x1xhlol/system-prompts-and-models-of-ai-tools

Händler, T. (2023). A Taxonomy for Autonomous LLM-Powered Multi-Agent Architectures. *International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management.*

Introducing ChatGPT agent: bridging research and action. (2025). Openai.com. https://openai.com/index/introducing-chatgpt-agent/

Introduction - Model Context Protocol. (2025). Modelcontextprotocol.io; Model Context Protocol. https://modelcontextprotocol.io/docs/getting-started/intro

Mei, L., Yao, J., Ge, Y., Wang, Y., Bi, B., Cai, Y., Liu, J., Li, M., Li, Z.-Z., Zhang, D., Zhou, C., Mao, J., Xia, T., Guo, J., & Liu, S. (2025). *A Survey of Context Engineering for Large Language Models*. ArXiv.org. https://arxiv.org/abs/2507.13334

Meta. (2025). The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation. Meta.com. https://ai.meta.com/blog/llama-4-multimodal-intelligence/

Sapkota, R., Roumeliotis, K. I., & Karkee, M. (2025). AI agents vs. agentic AI: A conceptual taxonomy, applications and challenges. *arXiv preprint arXiv:2505.10468*.

Vándor, P. & Csáki, Cs. (2025). RAG módszerek implementálásának kihívásai egy célorientált chatbot rendszer kontextusában. In: Berend, Gábor; Gosztolya, Gábor; Vincze, Veronika (Eds.) Magyar Számítógépes Nyelvészeti Konferencia online edition, Hungary: Szegedi Tudományegyetem TTIK, Informatikai Intézet (2025) 278 p., pp. 97-110., 14 p.

Yang, A., Li, A., Yang, B., Zhang, B., Hui, B., Zheng, B., ... & Qiu, Z. (2025). Qwen3 technical report. arXiv preprint arXiv:2505.09388.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K. R., & Cao, Y. (2022, October). React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

Zhao, A., Huang, D., Xu, Q., Lin, M., Liu, Y., & Huang, G. (2023). ExpeL: LLM Agents Are Experiential Learners. *AAAI Conference on Artificial Intelligence*.

Alkalmazások

Mondatszintű határozatrész-címkézés magyar bírósági határozatokon

Csányi Gergely Márk[1], Üveges István[1,4], Lakatos Dorina[1], Dóra Ripszám[2], Kornélia Kozák[3], Nagy Dániel[1], Vadász János Pál[2]

[1]GriffSoft Zrt., 1041 Budapest
[2]Ludovika Nemzeti Közszolgálati Egyetem, Információs Társadalom Kutatóintézet, UNESCO Tanszék, 1083 Budapest
[3]Ludovika Nemzeti Közszolgálati Egyetem, Államtudományi és Nemzetközi Tanulmányok Kar, Európai Köz- és Magánjogi Tanszék, 1083 Budapest
[4]ELTE Társadalomtudományi Kutatóközpont, 1097 Budapest
{gergely.csanyi,istvan.uveges,dorina.lakatos,daniel.nagy}@griffsoft.hu
{ripszam.dora,kozak.kornelia,vadasz.pal}@uni-nke.hu

Kivonat A cikk egy magyar bírósági határozatokon működő, mondatszintű határozatrész-címkéző (Rhetorical Role Labeling-RRL) rendszert mutat be. Az RRL feladata, hogy a bírósági határozatok minden mondatát a határozatban betöltött szerepe szerinti címkével lássa el (pl. tényállás, bírói érvelés, döntés, felek érvelése stb.), ezáltal támogatva többek között a szemantikus keresést, per kimenetelének predikcióját vagy az automatikus összefoglalást. A munkában szakértők által kézileg annotált korpuszon hasonlítottunk össze több architektúrát (BiLSTM, Attention és BiLSTM+Attention) és egy lineáris SVM referenciamodellt, különböző beágyazási stratégiákkal (huBERT CLS vs. Jina v3, late chunkinggal és anélkül). A címkekészlet nyolc osztályból állt. Eredményeink szerint a legjobb teljesítményt a huBERT CLS beágyazásokkal táplált BiLSTM érte el, mind dokumentumszintű pontosságban, mind (súlyozott és makró) F1-ben, érdemben felülmúlva az SVM-et. Meglepő módon a late chunking nem javította, hanem rontotta a mondatszintű RRL pontosságát, ami arra utal, hogy a túl tág dokumentumkontextus zajt vihet a mondatvektorokba. A rendszer már éles környezetben is működik: RAG-alapú jogi információkeresést támogat a magyar bírósági határozatokon.
Kulcsszavak: rhetorical role labeling, retorikaiszerep-címkézés, határozatrész-címkézés, late chunking, mondatszintű osztályozás

1. Bevezetés

A retorikaiszerep-címkézés (Rhetorical Role Labeling-RRL) olyan NLP feladat, amelyben a dokumentum egyes részeit (jelen esetben minden mondatát) a szövegben betöltött szerepe szerint osztályozzuk. A jogi területen azért érdekes probléma, mert a szerepek szerint felbontott határozatok jól hasznosíthatók több egyéb feladat esetében is. Ha képesek vagyunk felismerni egy szövegből, hogy mi az a szövegrész ami a tényállásról, vagy a bírói érvelésről szól, akkor ezeket a szövegrészeket felhasználhatjuk pl. jogeset predikcióhoz (feltételezve persze, hogy

hasonló bírósági ügyek hasonló ítélettel végződnek), de szemantikus kereséshez is, pl. a tényállásokra szűrve gyorsabban találhatók hasonló ügyek, vagy kivonatok minőségét lehet javítani azzal, hogy csak a releváns szerkezeti egységeket használjuk fel a kivonatoláshoz. Tudomásunk szerint magyar nyelvre még nem készült hasonló megoldás, nemzetközi szinten pedig a legtöbb hasonló munka angol nyelvre készült (Bhattacharya és mtsai, 2019; Malik és mtsai, 2021; Bambroo és mtsai, 2025), pár nem angol mellett (Marino és mtsai, 2023; Aragy és mtsai, 2021). Ebben a munkában egy magyar bírósági döntéseken alkalmazható, élesben is működő RRL-osztályozót mutatunk be. Több architektúrát hasonlítunk össze (BiLSTM és Attention-alapú modellek, lineáris SVM), és megvizsgáljuk a vektorizálás „late chunking" technikájának hatását is, amely a széles kontextusablakú beágyazási modellek segítségével képes a szövegrészeket átívelően kontextust biztosítani. A modelleket jogi szakértők által kézzel annotált, mondatszintű korpuszon tanítottuk és teszteltük.

2. Kapcsolódó irodalom

Bhattacharya és mtsai (2019) az indiai Legfelsőbb Bíróság öt jogterületről származó 50 ítéletén (összesen 9 380 mondat), mondatszintű RRL-feladaton dolgoztak 7 címkével. BiLSTM és BiLSTM-CRF modelleket vizsgáltak, a mondatok sorrendiségét a hierarchia és a CRF-tranzíciók révén hasznosítva. A mondatvektorokat sent2vec-ből származó, nagy jogi korpuszon előtanított beágyazásokkal állították elő.

Aragy és mtsai (2021) 70 darab portugál nyelvű polgári jogi beadványt, tehát nem bírósági határozatot címkézett fel mondatszinten. A mondatokat a BERT CLS tokenjével klasszifikálták, finomhangolva egy klasszifikációs réteget illetve a BERT egyes rétegeit is, tehát nem vették figyelembe a mondatok sorrendjéből származó plusz információt.

Malik és mtsai (2021) 100 angol nyelvű indiai ítéletből (50 versenyjogi, 50 adóügyi) álló, szakértőkkel annotált kb. 21 ezer mondatos korpuszt hoztak létre 7 retorikai címkével, mondatszinten, kifejezetten RRL-feladatra. A szerzők többfeladatos (Multi-Task Learning-MTL) modellt javasolnak, ahol a címkézéshez egy címkeváltozás predikció (label shift prediction) segédfeladat is társult a CRF mellett. Ez a megközelítés érdemben felülmúlta a mondat sorrendjétől független megoldásokat, bizonyítva a sorrendből származó extra információ fontosságát.

Marino és mtsai (2023) két korpuszon vizsgálódtak: egy kb. 1 500 olasz ítéletből álló ITA-RhetRoles korpuszon (5 címke) és a 275 dokumentumból álló BUILD angol korpuszon (13 címke), 96 ezer illetve kb 32 ezer mondattal. A szerzők hierarchikus modellt vizsgáltak (LEGAL-ToBERT: LEGAL-BERT fölé épített transzformer), amely a mondatok sorrendiségét és kölcsönös viszonyait is kiaknázza (pozicionális kódolással és mondatszintű encoderrel), és mindkét nyelven felülmúlta a sima LEGAL-BERT referenciamodellt.

Bambroo és mtsai (2025) mondatszintű RRL-t végeztek 7 címkével az indiai DIN (150 ítélet, 31 ezer mondat) és az angliai DUK (50 ítélet, 18 ezer mon-

dat) korpuszon. A javasolt MARRO modell BiLSTM-CRF-re épül multi-headed self-attentionnel, sent2vec vagy legal-bert-small beágyazásokkal és multi-task tanulással és label-shift segédfeladattal a modell a mondatok sorrendiségéből és a dokumentumon belüli távolabbi összefüggésekből is profitál.

3. Adathalmaz

3.1. Általános jellemzők

A célunk az volt, hogy az összes jelenleg nyilvánosan elérhető anonimizált bírósági határozatra elvégezzük az RRL címkézést, amely körülbelül 235 ezer dokumentumot jelent. A határozatok felépítése általában meglehetősen hasonló: a két fél, a bíróság, az előző bíróságok és az ügyek száma, valamint a tárgy információival kezdődnek, majd egy rövid bekezdés jön a döntésről. Ezt követi általában a tényállás és az előző bíróságok ítéleteinek ismertetése, majd a felek érvelései, újra a döntés, végül pedig a bíróság részletes érvelése a döntéssel kapcsolatban. A magyar bíróságok 2016 előtt másképp strukturálták ezeket a dokumentumokat, nem adtak egyértelmű fejezetcímeket a dokumentumokban, 2016 után pedig csak a Kúria volt köteles ilyen leíró fejezetcímeket adni, amit lassan az alacsonyabb szintű bíróságok is követtek. Mára már a fejezetcímeket tartalmazó megoldás vált az általánosabb szerkesztési móddá, amely egy neurális modell számára könnyen megtanulható címkézést tesz lehetővé. A továbbiakban a fejezetcímeket nem tartalmazó dokumentumokat a **régi típusú**nak, az ezeket tartalmazókat pedig **új típusú** dokumentumnak nevezzük.

A tanító, validációs és teszthalmazok jogterület szerinti eloszlását az 1. táblázat mutatja be. Eltérő volt az eloszlása a tanító és a teszteléshez használt halmazoknak. A tanítóhalmazban a jogterületek szinte egyenlően oszlottak el, a katonai büntetőügyeket büntetőügyként számolva, míg a teszthalmaz az egész, kb. 235 ezres korpusz eloszlását követte, hogy a tényleges pontosságot lehessen becsülni. A teljes korpuszban az új-régi típusú dokumentumok közti arány kb. 1:9-hez, mely fokozatosan egyenlítődik ki, mert az újabb dokumentumok már jellemzően új típusúak. A tanítóhalmaz minden jogterületen közel egyenlő számú régi és új típusú dokumentumot tartalmazott, kivéve a közigazgatási és katonai büntetőjogi eseteket. A régi típusú dokumentumok különösen fontosak voltak az értékelés szempontjából, mert az új típusú dokumentumokkal ellentétben nem tartalmaztak könnyen azonosítható fejezetcímeket, amelyek leegyszerűsíthették volna a címkézési feladatot, illetve a jelenlegi korpusz nagy többségben ilyen dokumentumokból áll. A dokumentumokat összesen hat jogi szakértő címkézte. Az annotátorok iránymutatásként megkapták a rendelkezésre álló címkék listáját (lásd 3.3. szakasz), amelyek egy jogász számára érthetők, és azt az utasítást, hogy mondatonként csak egy címkét adjanak, több lehetséges címke esetén a legmegfelelőbbet kiválasztva. Végül egy 299 dokumentumból álló tanító+validálási és egy 120 dokumentumból álló teszthalmazt hoztunk létre.

Annotátorok közti egyetértést 10 dokumentumon (összesen 2734 mondat) számítottunk két annotátor bevonásával. Metrikaként az átlagos egyezést illetve

a Krippendorff alfa (Krippendorff, 2011) értéket választottuk. A részletes eredményeket az A. Függelék 5. és 6. táblázatai mutatják be. A nyolc címkéből hat jó egyezést mutatott ($\alpha > 0,67$) a Tényállás (0,3899) és a Bíróság döntése (0,4243) kategóriák szerepeltek rosszabbul.

1. táblázat. A tanító, validációs és teszthalmaz jogterületenkénti eloszlása

Jogterület	Tanító+validációs halmaz			Teszthalmaz			Arány [%]	
	Régi típus	Új típus	Össz.	Régi típus	Új típus	Össz.	Teszt	Teljes korpusz
Büntető	27	26	53	13	3	16	13,33	15,83
Gazdasági	28	28	56	12	3	15	12,50	12,41
Katonai büntető	5	0	5	3	0	3	2,50	2,01
Közigazgatási	49	24	73	21	3	24	20,00	20,19
Munkaügyi	26	26	52	10	4	14	11,67	8,34
Polgári	30	30	60	42	6	48	40,00	41,22
Összes	165	134	299	101	19	120	100	100

3.2. Mondatra bontás

A jogi dokumentumokban a szöveg mondatokra bontása kihívást jelent, mivel a jogi dokumentumok sokkal több pont karaktert tartalmaznak, mint más területek korpuszai. Ezek közé tartoznak a különböző ítélkezési gyakorlatra, ügyszámra vagy jogszabályra való hivatkozások (pl. II. Pfv.35.125/2010/4, 32/2008. (VII. 19.) IM rendelet 8. § stb.), valamint az anonimizálás eredményeként monogramok és három pont is gyakran előfordul. A mondatokat a Csányi és mtsai (2024) című cikkben leírt, kifejezetten bírósági dokumentumokhoz módosított heurisztikus szegmentálóval bontottuk szét. Ez a mondatszegmentáló nagyon jó eredményeket ért el jogi dokumentumok esetében, jobb teljesítményt nyújtott a HuSpaCy magyar változatában (Orosz és mtsai, 2023) található transzformer-alapú megoldáshoz képest, miközben lényegesen gyorsabb is.

3.3. Címkekészlet

A mondatokat az alábbi címkék egyikével címkéztük:

- **Tényállás**: minden mondat, amely leírja, hogy miről szól a jogvita.
- **Perelőzmény**: mivel a bírósági határozatok minden szintjét vizsgáltuk (Kúria, Törvényszékek, Ítélőtáblák, Közigazgatási Bíróságok stb.), a dokumentumok tartalmazhatnak információkat korábbi döntésekről is.
- **Felek érvelése**: a felek érveiről és kérdéseiről szóló mondatok.
- **Bíróság döntése**: Az a jellemzően egy-két mondat, ami a döntés maga, pl. A bíróság a felperes keresetét elutasítja.
- **Bírói érvelés**: a bírói érvelés a döntés jogi alapját illetően.
- **Perköltség**: mondatok arról, hogy az ítélet alapján melyik félnek mennyit kell fizetnie perköltségként.

– **Rendelkező rész**: az ítélet gyakorlati következményeit leíró mondatok, például hogy az alperesnek X összeget kell fizetnie kártérítésként a felperes részére.
– **Egyéb**: egyik fenti kategóriába sem tartozó részek, például aláírások, a dokumentum fejléce, keltezés, fejezetcímek stb.

A címkék eloszlását, a mondatszámot, arányukat az adatbázisban, valamint a mondatonkénti tokenek számát a 2. táblázat mutatja be. A tokenek számát a `jinaai/jina-embeddings-v3` huggingface modell segítségével számítottuk ki.

2. táblázat. Címkék eloszlása, tokenátlagok mondatonként

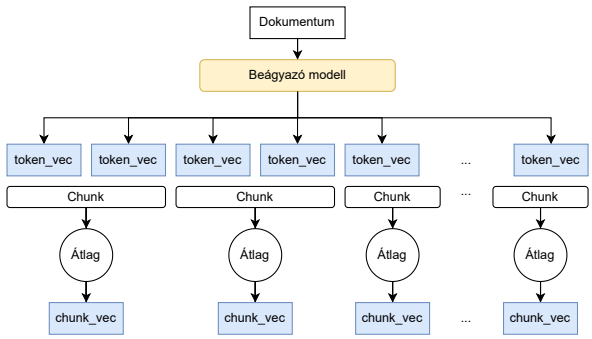
Label	Tanító és validációs adathalmaz			Teszthalmaz		
	Mondatszám	Arány	Token/mondat	Mondatszám	Arány	Token/mondat
Bíróság döntése	1657	0,0492	40,15	710	0,0402	47,70
Bírói érvelés	9864	0,2931	57,16	5899	0,3340	54,11
Felek érvelése	6639	0,1973	52,56	3198	0,1811	49,71
Egyéb	5518	0,1640	11,38	1380	0,0781	17,14
Perelőzmény	3735	0,1110	57,44	1653	0,0936	56,13
Perköltség	967	0,0287	57,08	450	0,0255	59,14
Rendelkező rész	441	0,0131	62,93	352	0,0199	55,45
Tényállás	4832	0,1436	49,32	4020	0,2276	49,52

4. Beágyazások

4.1. Jina beágyazások late chunkinggal és anélkül

A beágyazási modellek egy adott szöveget egy vektortérbe képeznek le. A korszerű, transzformer-alapú beágyazási megoldások azonban egy fix kontextusablakkal rendelkeznek. Hosszabb szövegek esetén gyakran szükséges a bemenetet kisebb darabokra felosztani, hogy mindegyik beférjen a modell kontextusablakába. Ezeknek a szövegrészeknek az egymástól függetlenül történő beágyazása azonban csökkenti a rendelkezésre álló dokumentumszintű kontextust.

Ennek kezelésére Günther és mtsai (2024) a JinaAI-tól egy kiváló, mégis egyszerű ötlettel állt elő, melyet late chunkingnak (utólagos felbontásnak) neveztek el. A koncepciót az 1. ábra szemlélteti. A late chunking során a teljes szöveget átadjuk a beágyazó modellnek, és kiszámításra kerülnek a token szintű beágyazások. Ezt követően történik csak a szöveg felbontása, vagyis az egyes chunkok kialakítása, mely során chunkvektorokat a chunkok tokenvektorainak átlagával képzünk. Így biztosítható, hogy az egyes szövegrészek közötti kontextus a beágyazásokban is megjelenjen. Ennek kiszámításához a beágyazó modellnek tokenszintű beágyazásokat is vissza kell adnia, és viszonylag nagyobb tokenablakkal kell rendelkeznie.



1. ábra. Late chunking

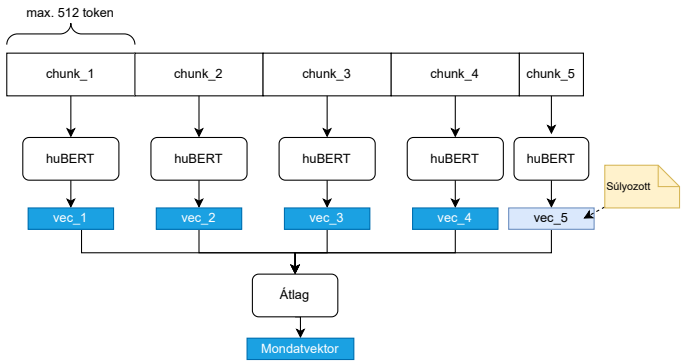
Több olyan beágyazási modell is van, amely akár 8192 tokent is képes lefedni, ám jellemzően ezek nem adnak vissza tokenszintű beágyazásokat, csak egy beágyazást a beadott szövegre, tehát ezekkel nem lehetséges a late chunking. Ilyen például az OpenAI `text-embedding-3-large` modellje, a Gemini `gemini-embedding-001` modellje. Tokenszintű beágyazásokat is visszaad például a Pekingi Mesterséges Intelligencia Akadémia BGE-M3 (Chen és mtsai, 2024) modellje, valamint a JinaAI `jina-embeddings-v3` (Sturua és mtsai, 2024) és `jina-embeddings-v4` modelljei (Günther és mtsai, 2025), valamint a Stella V5 modellje (Merola és Singh, 2025). Ebben a cikkben a Jina V3-as modelljét használtuk, a natív API segítségével, amelyben a late chunking beállítás egy egyszerű paraméterként elérhető.

4.2. BERT CLS

Második beágyazási modellként a huBERT (`SZTAKI-HLT/hubert-base-cc`) modell (Nemeskey, 2021) CLS token reprezentációját használtuk. Az előtanítás kizárólag magyar adatokon, többek között jogi dokumentumokon történt, ez némi előnyt jelentett feladatunkhoz, azonban a modellt nem finomhangoltuk. A modell maximális kontextusablaka 512 token, és 768 dimenziós CLS beágyazásokat hoz létre. A kontextusablaknál hosszabb mondatok kezelését a 2. ábra mutatja be. Ezeket a mondatokat maximum 512 token széles darabokra osztottuk, a szöveget csak a szavak határain bontva, a darabolt részek között átfedés nélkül. Minden egyes szövegrészhez kiszámítottuk a BERT CLS vektort. A kontextusablaknál hosszabb mondat beágyazása a felosztott adatok beágyazásainak átlaga volt, kivéve az utolsó darabot, ahol a vektort a tokenek számának és a kontextusablak arányával szoroztuk be, hasonlóan a Csányi és mtsai (2025) cikkhez.

4.3. Pozíciós jellemző

Ha a mondatokat úgy vektorizáljuk, hogy nem alkalmazzuk a late chunking módszerét, akkor a mondat helyzete a dokumentumban a kontextussal együtt teljesen



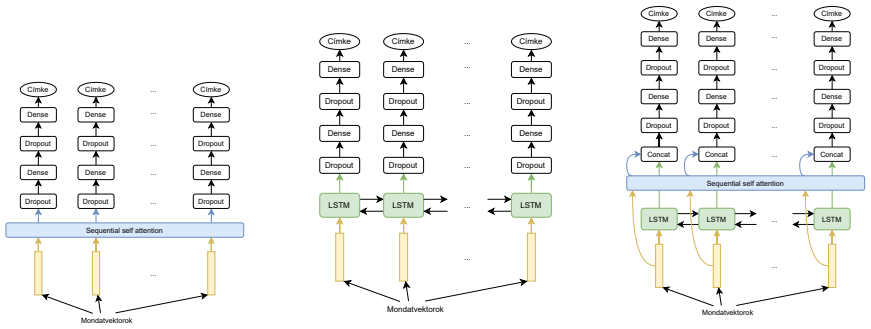
2. ábra. Kontextusablaknál hosszabb mondatok vektorizálása BERT CLS módszerrel

elveszik. Ezért kiszámítottuk a pozíciós jellemzőt, amely az adott mondat relatív helyzete a dokumentumban. Ez egy 0 és 1 közötti szám, amelyet plusz egy helyiértékként hozzáadtuk a már rendelkezésre álló beágyazásokhoz.

5. Osztályozási modellek

Az RRL egy szekvenciális osztályozási feladat, ahol a mondatok jelentése nem független egymástól, tehát a címkék sem függetlenek, így a mondat szövegben elfoglalt helye fontos információ az osztályozás során. Ennek igazolására kipróbáltuk referenciamodellként a lineáris SVM-et, amely nem képes kihasználni ezt az információt, valamint a szekvenciális címkézésre alkalmas neurális modelleket: BiLSTM (Hochreiter és Schmidhuber, 1997), Attention (Vaswani és mtsai, 2017) és BiLSTM+Attention hálókat. Az egyes architektúrák felépítése a 3. ábrán látható.

Minden architektúrában minden mondatot először beágyaztunk, majd ezek bekerültek az adott szekvenciális neurális modellbe. Minden szekvenciális kimeneti vektor egymásra helyezett Dense és Dropout rétegekbe, valamint egy mondatonkénti softmaxba tápláltunk be, nyolc neuront használva osztályozási rétegként. Az Attention architektúra (3a. ábra) self-attention-nel a mondatsor felett hoz létre kontextusérzékeny reprezentációt. A BiLSTM (3b. ábra) a kétirányú mondatkörnyezettel ad kontextust, azonban hosszabb szekvenciákra nem működik nagyon jól. A BiLSTM+Attention (3c. ábra) pedig mindkét módszer erősségeit egyesíti. Az implementáció során a `keras` framework-öt használtuk (2.15.0 verzió), a szekvenciális self-attention-höz a `keras-self-attention` (0.51.0 verzió) könyvtárat, a keresztvalidáláshoz és a lineáris SVM-hez pedig a `scikit-learn` (1.6.1 verzió) könyvtárat használtuk (Pedregosa és mtsai, 2011).



(a) Attention modell (b) BiLSTM modell (c) BiLSTM+Attention modell

3. ábra. Neurális modellek

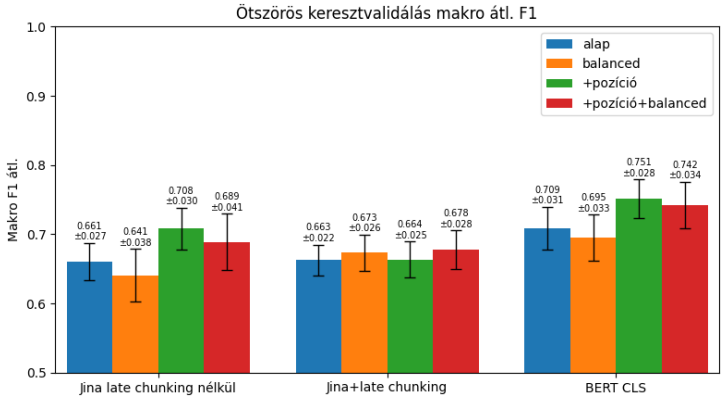
6. Eredmények

6.1. Lineáris kernelű SVM

Referenciaként a mondatvektorokat lineáris kernelű SVM-mel osztályoztuk. Finomhangoltuk a C paramétert, amely C=1 beállításnál volt a legjobb, és kipróbáltuk a `class_weight=balanced` beállítást is, mivel az adatunk nem homogén eloszlású volt. Az SVM nem képes kihasználni a mondatszekvenciából származó plusz információkat, kivéve ha late chunkingot alkalmazunk. Ötszörös keresztvalidálást végeztünk a 299 dokumentumból álló tanítási és validálási halmazok felhasználásával, ügyelve a jogterület szerinti arányos mintavételezésre. Összehasonlítottuk a Jina és a BERT CLS beágyazásokat pozíciós jellemzőkkel és anélkül, valamint a `class_weight=balanced` beállítással. Mérőszámként a makro átlag F1 metrikát választottuk. Az eredményeket a 4. ábra mutatja.

A late chunking nélküli beágyazások (Jina late chunking nélkül és BERT CLS) hasonló mintát követtek: a `class_weight=balanced` beállítás használata rontotta a teljesítményt, azonban a pozíciós jellemző hozzáadása előnyös volt, a makro F1 átlagot 4,76%-kal (66,07%-ról 70,83%-ra) és 4,25%-kal (70,85%-ról 75,10%-ra) emelte. Ezzel szemben late chunking esetén a `class_weight=balanced` beállítás bizonyos mértékben javította az eredményeket, és a pozíciós jellemző hozzáadása is pozitív, de marginális hatással volt, míg a legjobb late chunkinggal elért eredmény jelentősen a late chunking nélküli legjobb eredmény alatt maradt (70,83% vs. 67,77%).

A Jina vektorok alulteljesítése meglepő a BERT CLS vektorokkal szemben, egyfelől mert ez egy finomhangolt beágyazás, másfelől pedig a late chunkinggal végzett vektorizálás számos adathalmazon nagyobb javulást hozott a rövidebb, mint a hosszabb chunkoknál (Günther és mtsai, 2024). Korpuszunk mondatai



4. ábra. Különböző vektorformák és tanítási paraméterek összehasonlítása: `pozíció`: mondat relatív pozíciója, `balanced`: `class_weight=balanced` beállítás.

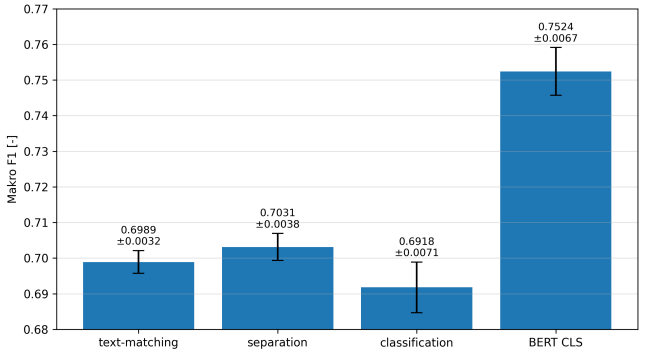
viszonylag rövidek voltak (átlagosan 50 token/mondat), ezért érdemi teljesítménynövekedést vártunk. Ezzel szemben egy friss tanulmányban Merola és Singh (2025) azt kapták, hogy Q&A feladatban a late chunking a visszakeresés teljesítményét rontotta, vagy legfeljebb csekély mértékben javította. A rossz teljesítmény miatt felmerült a Jina vektorizáló nem megfelelő beállítása. A Jina az alábbi vektorizálási feladatokat kínálja: *classification*, *text-matching*, *separation*, *retrieval.passage*, valamint *retrieval.query*, melyek közül a *classification* beállítást választottuk. Összehasonlítást végeztünk 5-szörös keresztvalidációval a tanító+validációs adatokon, már nem egy, hanem két ismétléssel, rögzített `random_seed` paraméterrel. A halmazok képzéséhez jogterület szerinti arányosított felosztást alkalmaztunk, mivel a mondatszintű osztályozás független a mondatok sorrendjétől. Összehasonlítottuk a late chunking nélküli Jina vektorokat a `separation`, `text-matching`, `classification` vektortípusokkal, illetve a BERT CLS beágyazásokkal. Az eredményeket az 5. ábra mutatja.

Érdekes módon a legjobb feladatnak nem a `classification` bizonyult, hanem a `separation` beállítás, de semelyik Jina beállítással sem sikerült megközelíteni a BERT CLS beágyazást.

Az eredmények tehát azt mutatták, hogy a mondatok jelentős része további szekvenciális információ kihasználása nélkül is helyesen osztályozható, illetve a late chunkinggal kapott kontextus rontott a klasszifikáció során. Ugyanakkor bebizonyosodott, hogy a szekvenciális információ fontos, mert a pozíciós jellemző hozzáadásával az eredmények számottevően javultak.

6.2. Neurális modellek

Mivel a lineáris SVM-mel kapott eredmények arra utaltak, hogy a Jina beágyazások nem teljesítenek jól az RRL feladatunkban, a neurális architektúrákat



5. ábra. Jina `separation`, `text-matching` és `classification` beágyazási feladatok összehasonlítása a BERT CLS-sel lineáris SVM osztályozóval

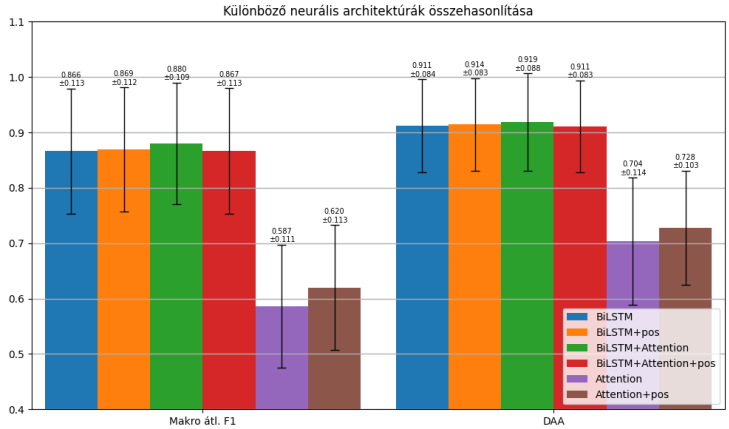
kizárólag a legjobban szereplő BERT CLS beágyazásokkal hasonlítottuk össze, pozíciós jellemző hozzáadásával, illetve anélkül. A tanítás során az adathalmazt 85% tanító és 15% validációs részre osztottuk, arányosított felosztással a jogterület és a régi ill. új dokumentumtípus szerint is. Minden epoch elején a dokumentumokat megkevertük (de a bennük lévő mondatokat nem), és kategorikus keresztentrópia veszteségfüggvényt alkalmaztunk.

A tanítás során használt paramétereket a B. Függelék 7. táblázata tartalmazza, melyekhez rövidebb próbálgatásokat követve jutottunk el. Minden tanítást háromszor végeztünk el, három különböző véletlenszerű állapotot (random state) állítva be a tanító és validálási halmazok bontásához.

Fő összehasonlítási metrikaként a makro F1 átlagot és a dokumentumszintű átlagos pontosság (Document Average Accuracy-DAA) metrikát választottuk. Ezeket minden dokumentumra a validációs halmazon külön számoltuk ki, és ezek átlagát és szórását a 6. ábra mutatja be. Az eredmények alapján az Attention háló képes megtanulni a jogi ügyek szabályszerűségeit, de csak korlátozott mértékben. A BiLSTM és a BiLSTM+Attention hálók sokkal jobban megragadták a jogi dokumentumok szabályszerűségeit, mintegy 20-25 F1 pontos különbséggel megelőzve az Attention modellt. A pozíciós jellemző hozzáadásának hatása az Attention modellnél marginálisan, de javított, míg a másik két architektúrában elhanyagolható volt. Ezért csak a legjobban szereplő BiLSTM és BiLSTM+Attention hálókat hasonlítottuk össze a teszthalmazon.

6.3. Eredmények a teszthalmazon

A BiLSTM és a BiLSTM+Attention hálókat a korábbi szakaszban bemutatott beállításokkal újratanítottuk az egyesített tanító és validációs adatokon. Az így kapott modelleket ezután a teszthalmazon értékeltük ki, melyet a 3. táblázat mutat be. Az eredmények azt mutatták, hogy a BERT CLS+BiLSTM beállítás működött a legjobban, bár az összes beállítás viszonylag hasonlóan teljesített, és mindegyik jó teljesítményt nyújtott az RRL feladatban.



6. ábra. Neurális architektúrák összehasonlítása BERT CLS vektorokkal a validációs halmazon; +**pos**: pozíciós jellemzővel

3. táblázat. Eredmények a teszthalmazon

Beágyazás	Neurális modell	DAA	Accuracy	Makro F1	Súlyozott F1
BERT CLS	BiLSTM	0,9226	0,9247	0,8849	0,9252
BERT CLS+pos	BiLSTM	0,8926	0,8828	0,8356	0,8853
BERT CLS	BiLSTM+Attention	0,8806	0,8668	0,8209	0,8690
BERT CLS+pos	BiLSTM+Attention	0,8964	0,8731	0,8317	0,8751

6.4. Címkeszintű eredmények a teszthalmazon

A legjobb modelllel számolt címkeszintű eredményeket a 4. táblázat tartalmazza.

4. táblázat. A legjobb modell címkeszintű eredményei

Címke	Pontosság	Fedés	F1	F1 Régi	F1 Új
Bírói érvelés	0,9747	0,9425	0,9583	0,9507	0,9915
Bíróság döntése	0,8341	0,8066	0,8201	0,8177	0,8421
Felek érvelése	0,9073	0,9407	0,9237	0,9078	0,9732
Egyéb	0,9939	0,8768	0,9317	0,9167	0,9625
Perelőzmény	0,9458	0,9178	0,9316	0,9131	0,9827
Perköltség	0,9389	0,9862	0,9620	0,9577	0,9783
Rendelkező rész	0,6144	0,6676	0,6399	0,6646	0,4267
Tényállás	0,8823	0,9432	0,9117	0,9114	0,9151

A címkék többsége esetében 0,9 feletti F1 értéket mértünk, két címke kivételével: a Bíróság döntése (0,8201) és a Rendelkező rész (0,6399) esetében. Jól

látható volt az is, hogy az új típusú dokumentumok esetében a modell jobban teljesített a régi típusú dokumentumoknál a Rendelkező rész kategória kivételével, alátámasztva, hogy az újabb típusú dokumentumok könnyebbek a kategorizálás szempontjából. Megállapítható volt még, hogy a régi dokumentumok esetében is nagyon jó eredményeket ért el a modell, a Bíróság döntése (0,8177) és a Rendelkező rész (0,6646) címkék kivételével mindenhol 0,9 feletti F1 értéket mértünk.

A Rendelkező rész címke a legkevesebbszer előforduló címke volt mind a tanító, mind a teszthalmazban. Ennek oka, hogy ez amolyan egyéb címke: csak azok a mondatok kapták ezt a címkét a határozat rendelkező részéből, amely egy bekezdésnyi szöveg a határozat elején, amelyek nem voltak besorolhatóak vagy a Perköltség vagy a Bíróság döntése kategóriába. Ezért ennek a címkének a fontossága sem nagy gyakorlati szempontból.

7. Összegzés

Bemutattuk tudomásunk szerint az első, magyar bírósági határozatokon működő, mondatszintű határozatrész-címkéző (Rhetorical Role Labeling-RRL) megoldást, amelyet egy újonnan összeállított korpuszon értékeltünk, és klasszikus, illetve neurális architektúrákkal vetettünk össze. Összhangban más nemzetközi kutatások eredményével azt tapasztaltuk, hogy a mondatok sorrendjéből származó információ jelentősen segíti a klasszifikáció pontosságát. A magyar BERT (huBERT) CLS-beágyazásokkal táplált BiLSTM adta a legerősebb összteljesítményt a teszthalmazon, egyértelműen felülmúlva a lineáris SVM alapmodellt, ami alátámasztja a szekvenciális információ jelentőségét. A visszakereséssel kapcsolatos irodalom friss eredményeivel ellentétben a „late chunking" rontotta a teljesítményt a mondatszintű RRL-ben, és a többnyelvű Jina v3 beágyazások sem bizonyultak jobbnak a magyar BERT CLS-nél. Ez arra utal, hogy a dokumentumkontextus „befecskendezése" fix mondatvektorokba zajt vihet a beágyazásokba, ezzel nehezítve a kategorizálást. A legjobb modellel a dokumentumok mondatai 92,2%-os átlagos pontossággal voltak osztályozhatók. A munka gyakorlati hatással is bír: a legjobb modell az Országos Bírósági Hivatalnál egy RAG-pipeline-ban teszi lehetővé a határozatrészek alapján történő szűrést, javítva a különböző jogi problémák kereshetőségét és magyarázhatóságát.

Hivatkozások

Aragy, R., Fernandes, E.R., Caceres, E.N.: Rhetorical role identification for portuguese legal documents. In: Brazilian Conference on Intelligent Systems. pp. 557–571. Springer (2021)

Bambroo, P., Adhikary, S., Bhattacharya, P., Chakraborty, A., Ghosh, S., Ghosh, K.: Marro: multi-headed attention for rhetorical role labeling in legal documents. Artificial Intelligence and Law pp. 1–30 (2025)

Bhattacharya, P., Paul, S., Ghosh, K., Ghosh, S., Wyner, A.: Identification of rhetorical roles of sentences in indian legal judgments. In: Legal knowledge and information systems, pp. 3–12. IOS Press (2019)

Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., Liu, Z.: M3-embedding: Multilinguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In: Findings of the Association for Computational Linguistics ACL 2024. pp. 2318–2335 (2024)

Csányi, G.M., Lakatos, D., Üveges, I., Vági, R., Megyeri, A., Fülöp, A., Nagy, D., Vadász, J.P.: Bírósági határozatok automatikus mondatszegmentálásának hatékonyságmérése. In: XX. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2024). Szegedi Tudományegyetem, Informatikai Intézet (2024)

Csányi, G.M., Lakatos, D.P., Vadász, J.P., Nagy, D., Üveges, I.: A kontextusablakon kihajolni nem veszélyes: jogi szövegek hatékony szemantikus keresése. In: XXI. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2025). Szegedi Tudományegyetem TTIK, Informatikai Intézet (2025)

Günther, M., Mohr, I., Williams, D.J., Wang, B., Xiao, H.: Late chunking: contextual chunk embeddings using long-context embedding models. arXiv preprint arXiv:2409.04701 (2024)

Günther, M., Sturua, S., Akram, M.K., Mohr, I., Ungureanu, A., Wang, B., Eslami, S., Martens, S., Werk, M., Wang, N., és mtsai: jina-embeddings-v4: Universal embeddings for multimodal multilingual retrieval. arXiv preprint arXiv:2506.18902 (2025)

Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation 9(8), 1735–1780 (1997)

Krippendorff, K.: Computing krippendorff's alpha-reliability (2011), https://repository.upenn.edu/entities/publication/034a6030-c584-4d14-9d3d-7b7e8d16df20

Malik, V., Sanjay, R., Guha, S.K., Hazarika, A., Nigam, S., Bhattacharya, A., Modi, A.: Semantic segmentation of legal documents via rhetorical roles. arXiv preprint arXiv:2112.01836 (2021)

Marino, G., Licari, D., Bushipaka, P., Comandé, G., Cucinotta, T., és mtsai: Automatic rhetorical roles classification for legal documents using legal-transformer over bert. In: CEUR WORKSHOP PROCEEDINGS. vol. 3441, pp. 28–36. CEUR-WS (2023)

Merola, C., Singh, J.: Reconstructing context: Evaluating advanced chunking strategies for retrieval-augmented generation. In: International Workshop on Knowledge-Enhanced Information Retrieval. pp. 3–18. Springer (2025)

Nemeskey, D.M.: Introducing `huBERT`. In: XVII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2021). p. TBA. Szeged (2021)

Orosz, G., Szabó, G., Berkecz, P., Szántó, Z., Farkas, R.: Advancing Hungarian Text Processing with HuSpaCy: Efficient and Accurate NLP Pipelines. In: Text, Speech, and Dialogue. pp. 58–69. Springer Nature Switzerland (2023)

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., és mtsai: Scikit-learn: Machine learning in python. the Journal of Machine Learning Research 12, 2825–2830 (2011)

Sturua, S., Mohr, I., Akram, M.K., Günther, M., Wang, B., Krimmel, M., Wang, F., Mastrapas, G., Koukounas, A., Wang, N., és mtsai: jina-embeddings-v3:

Multilingual embeddings with task lora. arXiv preprint arXiv:2409.10173 (2024)

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems 30 (2017)

Függelék

A. Annotátorok közötti egyetértés

5. táblázat. Annotátorok közötti egyetértés

Metrika	Érték
Átl. egyetértés	0,8706
Krippendorff Alfa	0,7777

6. táblázat. Címkeszintű annotátorok közötti egyetértés

Címke	Krippendorff Alfa	Átl. egyetértés
Bíróság döntése	0,4243	0,9693
Bírói érvelés	0,7568	0,8815
Felek érvelése	0,9479	0,9832
Egyéb	0,6998	0,9682
Perelőzmény	0,6768	0,9203
Perköltség	0,9283	0,9971
Rendelkező rész	0,8066	0,9942
Tényállás	0,3899	0,8175

B. Tanítási paraméterek

7. táblázat. Tanítás során használt paraméterek

Paraméter	Érték
Epochok	max 200
Tanulási ráta	0,001
Dropout	0,4
Rekurrens dropout	0,4
Batch méret	32
LSTM cellák	128
Elosztott dense neuronok	32
Early stopping	validációs veszteség
Early stopping patience	10 epoch, legjobb modell marad
Optimalizáló	AdamW
Attention window	10

MangaliCa: A Bilingual Vision-Language Model for Hungarian-English Image Captioning and Retrieval

Armand Szabó[1], Attila Borbély[1], Kevin Ye[1], Natabara Gyöngyössy[1],

[1]ELTE Eötvös Loránd University, Faculty of Informatics, Department of Artificial Intelligence
{txh7sp, nlz44o, d4s67v, natabara}@inf.elte.hu

Abstract. Recent advances in vision-language modeling are been driven by large-scale datasets and architectures that combine contrastive and generative objectives. While these developments have led to strong performance in high-resource languages such as English, comparable multimodal data for medium-resource languages remain restrained. As a step toward addressing this limitation, we introduce MangaliCa, the first Hungarian-English bilingual vision-language model designed for both image captioning and image-text retrieval, built upon the CoCa framework with a CLIP ViT-L/14 image encoder and a TinyLlama 1.1B language model. This model is supported by a newly constructed synthetic silver-standard 70-million-sample Hungarian-English image-text dataset that provides the multimodal foundation for bilingual training at scale.

A central contribution of this work is the construction of the bilingual image-caption dataset, the largest multimodal training set to date involving Hungarian. This dataset was created by aggregating four diverse English-language image-caption sources and extending them with Hungarian machine translations produced through a distributed, large-scale automated translation pipeline. The dataset pipeline incorporates parallelized image fetching, caption cleaning, translation quality control, and strict post-processing filters to ensure caption fidelity and cross-lingual alignment at scale. The resulting set provides rich linguistic variety, high visual diversity, and paired annotations for contrastive bilingual training.

Leveraging this dataset, MangaliCa is trained with a dual-objective setup that jointly optimizes contrastive alignment and autoregressive caption generation across both languages. The trained 1.8B parameter model supports both standalone caption generation and efficient image-text retrieval through its contrastive backbone. Experiments demonstrate effective retrieval and captioning performance, particularly on long-form data. With MangaliCa we intend to take a first step towards extending multimodal AI capabilities to Hungarian and establishing scalable data curation methodologies for multilingual dataset creation.

Keywords: Vision-Language modeling, Contrastive Captioner, Bilingual Image-Text Dataset, Synthetic Dataset Creation, Hungarian Image-Text Retrieval

1 Introduction

Recent progress in multimodal AI has been driven by large-scale vision–language models such as CLIP (Radford et al., 2021), BLIP (Li et al., 2022), PaLI (Chen et al., 2023), and CoCa (Yu et al., 2022). These systems achieve strong performance in captioning, retrieval, and cross-modal representation learning, but they rely almost exclusively on high-resource languages (primarily English). As a result, medium-resource languages such as Hungarian remain severely underrepresented in multimodal research.

This lack of multimodal resources limits both applied and scientific progress. No publicly available models provide high-quality Hungarian captioning or retrieval, and no large-scale Hungarian image-text datasets currently exist to support such development. Hungarian's complex morphology and flexible syntax further complicate adaptation from English-centric systems, making direct transfer suboptimal.

To address this gap, we introduce MangaliCa, the first Hungarian–English bilingual vision–language model for image captioning and image–text retrieval. Built on the CoCa framework, MangaliCa integrates a CLIP ViT-L/14 (openai, 2021) image encoder with a TinyLlama 1.1B language model (TinyLlama, 2025) and learns both contrastive alignment and generative captioning across two languages.

A central contribution of this work is the creation of a 70-million-sample bilingual image-caption dataset, constructed by translating large-scale English datasets into Hungarian through a distributed automated pipeline. This enables consistent cross-lingual supervision at a scale not previously available for Hungarian.

MangaliCa incorporates language-conditioning tokens and cross-attention-based multimodal decoding, enabling explicit language control and improved grounding. Despite being trained under constrained computational settings, the model achieves competitive bilingual retrieval performance and generates visually grounded captions in both Hungarian and English.

2 Related Work

2.1 Vision Representation Learning

The foundation of modern multimodal systems lies in the progress of visual representation learning. Early convolutional architectures were gradually replaced by the introduction of the Vision Transformer (Dosovitskiy et al., 2021), which demonstrated that transformer-based architectures can outperform CNNs when trained on sufficiently large datasets.

2.2 Contrastive Vision–Language Models

Contrastive learning has emerged as one of the most influential paradigms in aligning image and text modalities. Models such as CLIP (Radford et al., 2021),

ALIGN (Jia et al., 2021), and SigLIP (Zhai et al., 2023) use dual-encoder architectures where images and texts are independently encoded and aligned in a shared embedding space through a symmetric contrastive objective. This approach enables highly efficient image–text retrieval and zero-shot transfer to classification tasks.

2.3 Generative and Cross-Attention–Based Multimodal Models

Beyond purely contrastive dual-encoder models, several architectures such as BLIP, BLIP-2 (Li et al., 2023), PaLI, and Flamingo (Alayrac et al., 2022) introduce generative capabilities through cross-attention, enabling textual representations to attend to visual embeddings for richer multimodal reasoning and fluent captioning. CoCa unifies contrastive and generative objectives, using a frozen vision encoder and a transformer-based multimodal decoder with cross-attention to generate captions and support contrastive alignment. MangaliCa adapts CoCa to a bilingual setting, extending cross-attention–based multimodal generation to Hungarian and English.

2.4 Multilingual and Low/Medium-Resource Vision–Language Modeling

Multilingual multimodal learning is still in its early stages compared to monolingual English models. Large-scale multilingual datasets, such as those used in PaLI or multilingual CLIP variants, predominantly contain captions in globally spoken languages, with Hungarian appearing only in negligible quantities.

Previous work addressing medium-resource languages relies on translation-based augmentation or aligning multilingual text encoders to visual embeddings. However, such approaches are limited by the quality of translated captions, differences in linguistic structure, and the lack of consistent bilingual image-text supervision.

This work contributes to constructing the largest Hungarian–English bilingual image–caption dataset to date and by training a model specifically optimized for bilingual multimodal alignment.

3 Dataset Curation

3.1 Overview and Motivation

Training a bilingual Hungarian–English vision-language model requires a large-scale multimodal dataset with sufficient visual diversity and parallel linguistic supervision. While numerous English image–text datasets exist, no comparable Hungarian resource has been available, preventing models from learning cross-lingual alignment or producing Hungarian captions at scale.

To address this gap, we constructed the first large-scale Hungarian–English multimodal corpus by transforming several extensive English image–caption datasets into a bilingual resource through automated translation and large-scale

image retrieval. The resulting dataset contains approximately seventy million aligned Hungarian–English image-caption pairs and forms the foundation of the MangaliCa model, representing the largest multimodal Hungarian dataset to date.

3.2 Source Datasets and Integration Strategy

Our bilingual corpus integrates several large-scale English image–caption datasets selected for their size, caption quality, and domain coverage. Despite differences in structure and style, all sources provide image URLs paired with a single English caption, enabling a unified processing pipeline.

The primary source is **DataComp-1B** (Li et al., 2024), offering hundreds of millions of web-crawled image–text pairs filtered via CLIP scores. **Conceptual Captions 12M** (Changpinyo et al., 2021) contributes higher-quality alt-text descriptions, while **GBC10M** (Hsieh et al., 2025) introduces captions with scene-graph structure, enriching relational information. Finally, **PixelProse** (Singla et al., 2024), generated using Gemini Pro Vision (Gemini Team, 2025), provides long-form and stylistically diverse captions that complement the other datasets.

3.3 Automated Bilingual Dataset Construction

To construct the bilingual corpus at scale, we implemented a fully automated dataset creation pipeline that jointly handles caption translation, image retrieval, and data cleaning. Due to computational constraints, the pipeline operates on fixed-size shards processed in parallel, allowing tens of millions of samples to be handled efficiently.

For each shard, English captions are automatically translated into Hungarian using a machine translation backend. Translations are validated for completeness, and malformed or empty outputs are discarded. In parallel, images are retrieved from the original URLs using a high-throughput asynchronous downloader. Retrieved images are decoded, checked for corruption or extremely low resolution, and standardized to a uniform RGB JPEG format.

Following translation and image processing, each image-caption pair undergoes a final cleaning stage. Samples with missing captions, invalid images, or failed preprocessing steps are removed. Cleaned shards are stored in Parquet format and merged into consolidated one-million–sample batches. Repeating this process across all partitions yields the final bilingual corpus of approximately seventy million aligned Hungarian–English image-caption pairs used for training MangaliCa.

The pipeline execution is fully automated and distributed across multiple parallel workers; implementation details are provided in the appendix.

Table 1. Final remaining samples after cleaning for each dataset (approximate million).

Dataset	**DataComp-1B**	**Conceptual Captions 12M**	**GBC10M**	**PixelProse**
Remaining (M)	40	8	8	14

4 Model Architecture

4.1 Overview of the MangaliCa Architecture

MangaliCa follows a modular two-tower design inspired by the CoCa framework (Yu et al., 2022), integrating a CLIP ViT-L/14 vision encoder (openai, 2021) with TinyLlama 1.1B language model (TinyLlama, 2025). This architecture (Figure 4) allows the model to operate simultaneously as a retrieval system and a caption generator, an ability made possible by combining contrastive alignment with autoregressive generation.

The overarching design objective was to adapt the CoCa paradigm to a bilingual context, while remaining computationally lightweight enough to train on free-tier or constrained hardware environments. To achieve this, MangaliCa incorporates language-conditioning tokens, cross-attention extensions in the decoder, and LoRA-based fine-tuning modules within both the vision and language components.

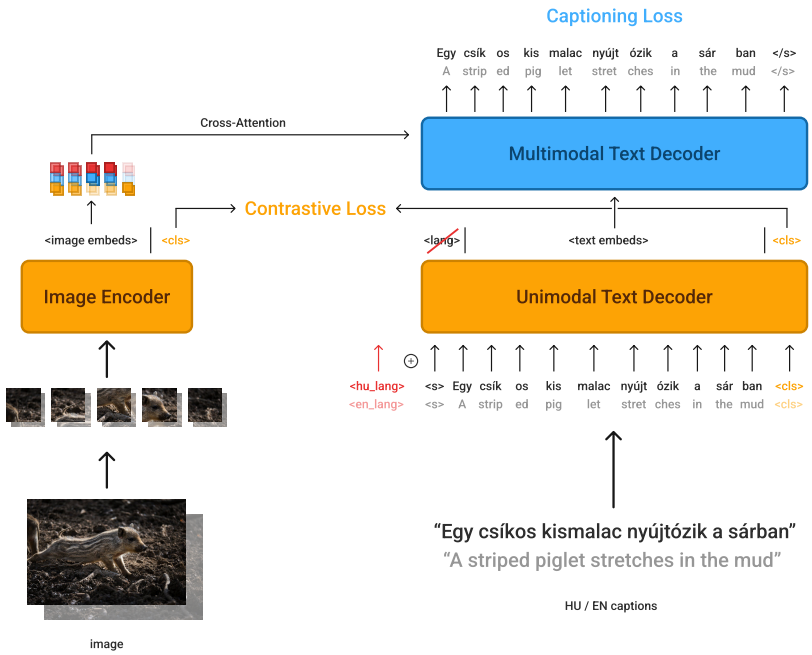


Fig. 1. High-level overview of the MangaliCa architecture, showing the frozen CLIP vision encoder, the unimodal TinyLlama decoder used for contrastive text embeddings, and the multimodal decoder that integrates image features via cross-attention for caption generation in both Hungarian and English.

These modifications allow the model to maintain expressive multimodal fusion while supporting languages with widely differing morphological and syntactic characteristics, such as Hungarian.

4.2 Vision Encoder: CLIP ViT-L/14

The visual backbone of MangaliCa is the CLIP ViT-L/14 encoder, a large Vision Transformer pretrained on hundreds of millions of image–text pairs through contrastive learning. This encoder maps an input image to a sequence of patch embeddings and a global CLS representation, it serves as the image embedding for both contrastive alignment and caption generation. Since CLIP ViT-L/14 provides strong and stable multimodal priors, we freeze the weights of it, ensuring computational efficiency and preventing catastrophic forgetting.

To allow limited adaptation without fully fine-tuning the vision tower, LoRA adapters are inserted into attention and MLP blocks. These adapters increase representational flexibility while maintaining CLIP stability and memory efficiency. This results in visual embeddings that remain CLIP-aligned but gain slight adaptation to the Hungarian linguistic distribution encountered during training.

4.3 Text Encoder–Decoder: TinyLlama 1.1B

The text-processing module of MangaliCa uses TinyLlama 1.1B, a lightweight Transformer based on the LLaMA architecture, chosen for its efficiency and suitability for low-cost fine-tuning.

The model is split into two equal components. The unimodal decoder processes text-only inputs with language conditioning tokens and produces a CLS representation for the contrastive objective, aligning caption embeddings with CLIP's image space.

The multimodal decoder augments its blocks with cross-attention to visual tokens from the vision encoder, enabling fine-grained image grounding. Each multimodal block further includes an additional SwiGLU layer after cross-attention.

For improved bilingual generation, the model is initialized from an early, instruction-free checkpoint to avoid bias and better support Hungarian adaptation.

4.4 Language Conditioning Mechanism

To support bilingual understanding, MangaliCa has two learned language tokens, one representing English and the other Hungarian. These tokens are prepended to the caption before processing in the unimodal decoder. Their purpose is to signal the target language to the model, guiding both contrastive alignment and caption generation.

During the forward pass, the language token is concatenated to the textual embedding sequence. This ensures that generative loss computation remains

identical to the original CoCa formulation while still allowing the model to distinguish between languages. Through repeated training, the model learns to associate each language token with characteristic of the respective language, enabling bilingual captioning without any architectural branching.

4.5 Multimodal Fusion via Cross-Attention

Multimodal fusion occurs within the multimodal decoder, where cross-attention mechanism fuse the image and text modalities. Each multimodal block receives a pooled sequence of visual patch embeddings produced by vision encoder, enabling the decoder to condition caption generation on the visual queries.

This design provides token-level grounding, where each generated token can attend to pooled image patches, producing captions that reflect visual structure. The cross-attention layers are followed by additional SwiGLU modules, improving both stability and representational capacity.

4.6 Dual Training Objective: Contrastive–Generative Learning

MangaliCa uses a hybrid training objective consisting of a symmetric InfoNCE contrastive loss and an autoregressive captioning loss, following the CoCa paradigm. This combination allows the model to function simultaneously as a retrieval encoder and a generative captioning model.

Contrastive alignment. For each batch of image–caption pairs, the model computes normalized image embeddings $\hat{z}_i^{\text{img}}$ and text embeddings $\hat{z}_j^{\text{text}}$. The similarity between the normalized embeddings is calculated via their dot product and scaled by a learned temperature parameter τ:

$$s_{ij} = \frac{\hat{z}_i^{\text{img}} \cdot \hat{z}_j^{\text{text}}}{\tau} \tag{1}$$

This contrastive objective is commonly instantiated through InfoNCE loss Eq. (2), which encourages matched image–text pairs to obtain higher similarity scores while uniformly pushing apart all mismatched combinations. It is applied symmetrically where the first term represents image-to-text, while the second ensures text-to-image consistency.

$$L_{Con} = -\frac{1}{2N} \sum_{N}^{i=1} \left[\log \frac{e^{s_{ii}}}{\sum_N^{j=1} e^{s_{ij}}} + \log \frac{e^{s_{ii}}}{\sum_N^{j=1} e^{s_{ji}}} \right] \tag{2}$$

The loss encourages the pairs to have high cosine similarity while pushing apart mismatched pairs. This alignment enables zero-shot retrieval in both languages, since the Hungarian and English captions are mapped to the same multilingual space.

Autoregressive generation. The captioning objective trains the multimodal decoder to predict the next token probability $P(x_t|x_{<t}, v^{\text{img}})$ given prior tokens $x = (x_1, x_2, ..., x_T)$ and visual features v^{img}:

$$L_{Cap} = -\sum_{T}^{t=1} \log P(x_t | x_{<t}, v^{\text{img}}) \tag{3}$$

This produces consistent English and Hungarian captions grounded in image content. The formulation ensures autoregressive consistency the model learns to produce fluent and contextually grounded captions token by token.

The combined loss of $L_{MangaliCa}$ is given:

$$L_{MangaliCa} = \lambda_{Con} L_{Con} + \lambda_{Cap} L_{Cap}, \tag{4}$$

where L_{Con} denotes the contrastive loss and L_{Cap} refers to the autoregressive loss. The weight coefficients of the equation (λ_{Con}, λ_{Cap}) are balancing factors regulating the contributions of each objective. This dual mechanism enables MangaliCa to serve as a bilingual CLIP-style encoder and a vision-conditioned language generator within a single unified architecture.

5 Training Setup and Process

5.1 Training Environment and Hardware Constraints

MangaliCa was trained entirely on free-tier Google Colaboratory environments equipped with NVIDIA T4 GPUs. These runtimes impose strict VRAM limits and short execution windows, which influenced the design of the training workflow. To make large-scale multimodal training feasible, we relied on memory-efficient techniques including NF4 weight quantization, LoRA adapters in both towers, gradient checkpointing, and FlashAttention in the decoder. Together, these methods reduced memory usage sufficiently to accommodate a 1.8B parameter model on a single T4 GPU.

Due to Colab's limited RAM, the full 70M-sample dataset could not be loaded at once. Instead, training proceeded in cycles of approximately 300k bilingual samples per session. Each shard included images normalized using CLIP preprocessing and captions tokenized with the TinyLlama tokenizer. Batches were constructed to jointly support contrastive and generative objectives, each batch provided image–text pairs for contrastive alignment, as well as token sequences for autoregressive captioning.

Table 2. Parameter count of each component in MangaliCa
**The added Cross-Attentions and the corresponding SwiGLU modules are not counted into the Multimodal Decoder. Altough they total count yields approximately 400M parameters.*

Components	Params	LoRA Params	Rank	Trainable (%)
Embeddings	65M	0.5M	16	0.07%
Vision Encoder	300M	7M	16	2%
Uni- Decoder	550M	6M	16	1%
*Multi- Decoder	480M	1.5M	4	0.07%
MangaliCa	1.4B	15M	-	0.01%

Training proceeded across many Colab sessions, with checkpoints loaded and resumed until runtime limits were reached. The updated checkpoint was saved before termination. Despite the fragmented execution environment, MangaliCa converged over roughly 22 days of cumulative training.

5.2 Initialization and Training Parameters

The CLIP ViT-L/14 encoder was loaded from its pretrained checkpoint and kept frozen, with LoRA adapters inserted for lightweight adaptation to the bilingual data. TinyLlama 1.1B was initialized from an early pretraining checkpoint to avoid instruction-tuning biases, and its blocks were split into unimodal and multimodal segments following the CoCa design. Newly initialized cross-attention layers were added to the multimodal part to enable visual grounding. The tokenizer and embedding layer remained unmodified to ensure consistency.

The model was trained using the AdaFactor (Shazeer and Stern, 2018) optimizer, which offers adaptive learning behavior with sublinear memory overhead. Each training step involved a batch forward and backpropagation of both the contrastive and autoregressive captioning loss with a weight of 2:1 ratio. The learning rate followed a warmup phase and cosine decay schedule, with weight decay applied for regularization. In total, the model was exposed to $11,206,656$ samples utilizing only the first random 16% of our synthetically translated data.

Table 3. Hyper-parameters used in the training of MangaliCa.

Hyper-parameter	**MangaliCa**
Optimizer	Adafactor with Weight Decay
Max Gradient Norm	1.0
Scheduler	Cosine
Mixed precision	FP16 / BF16
Weight decay	0.001
Warmup ratio	0.03
LR rate	1e-5
Per-device batch size	16
Gradient accumulation	2
Sequence per multilingual batch (per-device · grad-acc · 2)	64
Train Epochs	1400

6 Evaluation and Results

Across all retrieval benchmarks, MangaliCa shows strong bilingual alignment and competitive zero-shot performance. We evaluate on three datasets, **GBC10M** (Hsieh et al., 2025), **MS-COCO** (Lin et al., 2014), and **text-to-image-2M** (jackyhate, 2025) (reported in the appendix), each with 1,000 images and English captions translated to Hungarian using DeepL (DeepL GmbH, 2025). We additionally evaluate on 1,000 samples from **XM3600** (Thapliyal et al., 2022), which provides gold-standard Hungarian captions (reported in the appendix). Retrieval is performed in a shared contrastive space, and we primarily report Hungarian results using Recall@K, MRR, and NDCG@K.

We compare against several widely used vision–language models. CLIP-L is a large-scale contrastive model aligning image and text embeddings via paired data. SigLIP employs a sigmoid-based loss for improved scalability and training stability, with SigLIP2 (Tschannen et al., 2025) introducing further architectural and optimization refinements. M-CLIP (Multilingual CLIP) (Carlsson et al., 2022) trains a multilingual text encoder via cross-lingual teacher learning to mimic frozen English CLIP embeddings using parallel text data. The approach requires no image supervision and uses a pretrained language model with a trainable linear projection to align with the CLIP embedding space, relying on visually grounded caption datasets, including short-form captions such as MS-COCO, translated into multiple languages.

Table 4. Zero-shot image-text retrieval (Hungarian) results on our translated GBC10M (Hsieh et al., 2025) evaluation subset.

Model	R@1	R@3	R@5	R@25	R@100	NDCG@1	NDCG@10	NDCG@100	MRR
MangaliCa (ours)	35.6%	60.0%	70.0%	**91.0%**	**98.6%**	35.6%	57.5%	61.4%	0.51
CLIP-L	9.0%	13.4%	15.7%	24.0%	34.6%	9.0%	13.4%	16.6%	0.13
SigLIP	22.2%	33.3%	38.2%	56.3%	75.2%	22.2%	33.1%	39.1%	0.30
SigLIP2	16.3%	28.4%	33.9%	55.3%	75.1%	16.3%	28.6%	35.0%	0.25
M-CLIP	**43.6%**	**65.4**	**74.0%**	**91.0%**	97.5%	**43.6%**	**64.0%**	**66.1%**	**0.57**

Table 5. Zero-shot image-text retrieval (Hungarian) results on the MS-COCO (Lin et al., 2014) subset.

Model	R@1	R@3	R@5	R@25	R@100	NDCG@1	NDCG@10	NDCG@100	MRR
MangaliCa (ours)	6.05%	12.2%	17.3%	43.5%	69.3%	6.05%	14.4%	23.3%	0.13
CLIP-L	5.1%	8.6%	12.1%	29.0%	40.7%	5.1%	10.4%	15.2%	0.09
SigLIP	10.5%	18.6%	23.4%	44.1%	59.4%	10.5%	20.2%	25.7%	0.18
SigLIP2	3.6%	5.9%	7.9%	14.0%	25.7%	3.6%	6.6%	9.5%	0.06
M-CLIP	**50.0%**	**71.3%**	**81.5%**	**95.7%**	**100.0%**	**50.0%**	**69.0%**	**71.6%**	**0.63**

In general, MangaliCa delivers stable and competitive retrieval performance across all benchmarks, with its strongest results on datasets containing longer and more structured captions that resemble its training data. This verbosity aligns well with query formulation in multimodal retrieval-augmented generation systems, making MangaliCa embeddings well suited for image–text cross-retrieval tasks. Performance decreases on less structured, web-style captions, yet the model remains competitive with multilingual baselines, outperforming CLIP-L and SigLIP2.

M-CLIP achieves consistently strong performance, particularly on MS-COCO and benchmarks dominated by short-form captions, reflecting its training on concise translated image descriptions. This makes M-CLIP well suited for retrieval scenarios involving brief, visually grounded captions. In contrast, MangaliCa relies on explicit Hungarian–English parallel supervision derived from synthetic

data, demonstrating that targeted bilingual alignment can yield competitive retrieval performance without large-scale multilingual distillation pipelines.

6.1 Image Captioning

Captioning quality is evaluated by generating Hungarian and English descriptions for unseen images using a 1,000-sample subset of our evaluation data (GBC10M (Hsieh et al., 2025)). On the bilingual captioning task, MangaliCa achieves a combined BLEU score of 41, with separate scores of 32.4 for Hungarian and 45.3 for English, indicating reasonable alignment with the references. For comparison, we evaluate the Gemma-3-4B-IT (Gemma Team et al., 2025) baseline, a general-purpose instruction-tuned vision–language model, on the same Hungarian evaluation set, where it attains a BLEU score of 1.02. Manual inspection suggests that this low score primarily reflects limitations of the BLEU metric, as variability in reference length and lexical choice leads to strong brevity penalties and limited n-gram overlap despite semantically appropriate captions.

To account for these limitations, we additionally assess caption quality using a multilingual CLIP-based image–text alignment metric. Specifically, we compute cosine similarity between frozen CLIP image embeddings and multilingual CLIP text embeddings of the generated Hungarian captions. Table 6 reports the resulting alignment scores for MangaliCa and Gemma-3-4B-IT. While Gemma achieves a slightly higher mean cosine similarity, MangaliCa exhibits comparable performance, supporting the qualitative correctness and visual grounding of its generated captions.

Table 6. CLIP-based image–caption alignment on the Hungarian GBC10M subset (1,000 images). Cosine similarity is computed between frozen CLIP image embeddings and multilingual CLIP text embeddings of generated captions.

Model	Mean Cosine Similarity	Std. Dev.
Gemma-3-4B-IT	0.292	0.030
MangaliCa (ours)	0.258	0.038

6.2 Cross-Lingual Text–Text Retrieval

To evaluate the quality of the shared bilingual embedding space, we perform English-to-Hungarian text-to-text retrieval using a 1,000-sample subset drawn from our evaluation splits (GBC10M (Hsieh et al., 2025)). The task measures whether the model can correctly retrieve the Hungarian caption corresponding to an English caption using textual embeddings alone. Retrieval performance is reported using Recall@K and MRR.

Table 7. Comparison on English-to-Hungarian cross lingual retrieval task.

Model	R@1	R@3	R@5	MRR
MangaliCa (ours)	**97.0%**	**99.0%**	**99.0%**	**0.98**
CLIP-L	1.0%	14.0%	15.5%	0.13
SigLIP	23.3%	36.8%	43.2%	0.32
SigLIP2	18.5%	29.1%	34.1%	0.27
M-CLIP	2.0%	4.0%	5.4%	0.04

MangaliCa achieves exceptionally high cross-lingual alignment, reaching 97% Recall@1 and 0.98 MRR, indicating that paired English–Hungarian captions are consistently retrieved as top-ranked matches. This behavior reflects the model's training setup: each image is associated with parallel English and Hungarian captions, and the contrastive objective encourages their embeddings to converge in the shared latent space. Compared to multilingual baselines, MangaliCa preserves semantic equivalence across languages much more effectively, demonstrating a coherent and well-structured bilingual embedding space.

7 Conclusion

In this paper we presented MangaliCa, the first Hungarian–English bilingual vision–language model for captioning and image–text retrieval, trained on 11.2 million samples from our novel 70-million–sample synthetically translated multimodal dataset. Built with a fully automated pipeline and a CoCa-based architecture combining CLIP ViT-L/14 and TinyLlama, the model learns effective bilingual contrastive alignment and visually grounded generation.

MangaliCa achieves comparable Hungarian image-text retrieval to state-of-the-art SigLIP and CLIP models outperforming them on structured 1-3 sentences long text. Our model delivers solid performance in bilingual captioning, and excellent cross-lingual alignment, showing that high-quality multimodal modeling is feasible even for medium-resource languages. Benchmarks also indicate that M-CLIP is a strong model that outperforms all other models (including MangaliCa) on most tasks. While the exact translated datasets are not published, we believe that the target language text encoder selection of M-CLIP could be crucial for future multilingual model building. We hope this work provides a foundation for future research on Hungarian multimodality, native multimodal LLM training and multilingual dataset construction. Our model and dataset is available on Huggingface[1,2].

Acknowledgements

We thank Béla Szekeres and András Simonyi for their valuable feedback as internal reviewers.

[1]https://huggingface.co/Obscure-Entropy/MangaliCa

[2]https://huggingface.co/datasets/Obscure-Entropy/MangaliCa_EN-HU

Bibliography

Alayrac, J.B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millicah, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J., Borgeaud, S., Brock, A., Nematzadeh, A., Sharifzadeh, S., Binkowski, M., Barreira, R., Vinyals, O., Zisserman, A., Simonyan, K.: Flamingo: a visual language model for few-shot learning. In: Proceedings of the 36th International Conference on Neural Information Processing Systems. NIPS '22, Curran Associates Inc., Red Hook, NY, USA (2022)

Carlsson, F., Eisen, P., Rekathati, F., Sahlgren, M.: Cross-lingual and multilingual CLIP. In: Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Odijk, J., Piperidis, S. (eds.) Proceedings of the Thirteenth Language Resources and Evaluation Conference. pp. 6848–6854. European Language Resources Association, Marseille, France (Jun 2022), https://aclanthology.org/2022.lrec-1.739/

Changpinyo, S., Sharma, P., Ding, N., Soricut, R.: Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts (2021), https://arxiv.org/abs/2102.08981

Chen, X., Wang, X., Changpinyo, S., Piergiovanni, A., Padlewski, P., Salz, D., Goodman, S., Grycner, A., Mustafa, B., Beyer, L., Kolesnikov, A., Puigcerver, J., Ding, N., Rong, K., Akbari, H., Mishra, G., Xue, L., Thapliyal, A.V., Bradbury, J., Kuo, W., Seyedhosseini, M., Jia, C., Ayan, B.K., Ruiz, C.R., Steiner, A.P., Angelova, A., Zhai, X., Houlsby, N., Soricut, R.: PaLI: A jointly-scaled multilingual language-image model. In: The Eleventh International Conference on Learning Representations (2023), https://openreview.net/forum?id=mWVoBz4W0u

DeepL GmbH: DeepL Translation API. https://www.deepl.com/docs-api (2025), accessed: 2025-11-28

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=YicbFdNTTy

Gemini Team: Gemini: A family of highly capable multimodal models (2025), https://arxiv.org/abs/2312.11805

Gemma Team, Kamath, A., Ferret, J., Pathak, S., Vieillard, N., Merhej, R., Perrin, S., Matejovicova, T., Ramé, A., Rivière, M., Rouillard, L., Mesnard, T., Cideron, G., bastien Grill, J., Ramos, S., Yvinec, E., Casbon, M., Pot, E., Penchev, I., Liu, G., Visin, F., Kenealy, K., Beyer, L., Zhai, X., Tsitsulin, A., Busa-Fekete, R., Feng, A., Sachdeva, N., Coleman, B., Gao, Y., Mustafa, B., Barr, I., Parisotto, E., Tian, D., Eyal, M., Cherry, C., Peter, J.T., Sinopalnikov, D., Bhupatiraju, S., Agarwal, R., Kazemi, M., Malkin, D., Kumar, R., Vilar, D., Brusilovsky, I., Luo, J., Steiner, A., Friesen, A., Sharma, A., Sharma, A., Gilady, A.M., Goedeckemeyer, A., Saade, A., Feng, A., Kolesnikov, A., Bendebury, A., Abdagic, A., Vadi, A., György, A., Pinto,

A.S., Das, A., Bapna, A., Miech, A., Yang, A., Paterson, A., Shenoy, A., Chakrabarti, A., Piot, B., Wu, B., Shahriari, B., Petrini, B., Chen, C., Lan, C.L., Choquette-Choo, C.A., Carey, C., Brick, C., Deutsch, D., Eisenbud, D., Cattle, D., Cheng, D., Paparas, D., Sreepathihalli, D.S., Reid, D., Tran, D., Zelle, D., Noland, E., Huizenga, E., Kharitonov, E., Liu, F., Amirkhanyan, G., Cameron, G., Hashemi, H., Klimczak-Plucińska, H., Singh, H., Mehta, H., Lehri, H.T., Hazimeh, H., Ballantyne, I., Szpektor, I., Nardini, I., Pouget-Abadie, J., Chan, J., Stanton, J., Wieting, J., Lai, J., Orbay, J., Fernandez, J., Newlan, J., yeong Ji, J., Singh, J., Black, K., Yu, K., Hui, K., Vodrahalli, K., Greff, K., Qiu, L., Valentine, M., Coelho, M., Ritter, M., Hoffman, M., Watson, M., Chaturvedi, M., Moynihan, M., Ma, M., Babar, N., Noy, N., Byrd, N., Roy, N., Momchev, N., Chauhan, N., Sachdeva, N., Bunyan, O., Botarda, P., Caron, P., Rubenstein, P.K., Culliton, P., Schmid, P., Sessa, P.G., Xu, P., Stanczyk, P., Tafti, P., Shivanna, R., Wu, R., Pan, R., Rokni, R., Willoughby, R., Vallu, R., Mullins, R., Jerome, S., Smoot, S., Girgin, S., Iqbal, S., Reddy, S., Sheth, S., Põder, S., Bhatnagar, S., Panyam, S.R., Eiger, S., Zhang, S., Liu, T., Yacovone, T., Liechty, T., Kalra, U., Evci, U., Misra, V., Roseberry, V., Feinberg, V., Kolesnikov, V., Han, W., Kwon, W., Chen, X., Chow, Y., Zhu, Y., Wei, Z., Egyed, Z., Cotruta, V., Giang, M., Kirk, P., Rao, A., Black, K., Babar, N., Lo, J., Moreira, E., Martins, L.G., Sanseviero, O., Gonzalez, L., Gleicher, Z., Warkentin, T., Mirrokni, V., Senter, E., Collins, E., Barral, J., Ghahramani, Z., Hadsell, R., Matias, Y., Sculley, D., Petrov, S., Fiedel, N., Shazeer, N., Vinyals, O., Dean, J., Hassabis, D., Kavukcuoglu, K., Farabet, C., Buchatskaya, E., Alayrac, J.B., Anil, R., Dmitry, Lepikhin, Borgeaud, S., Bachem, O., Joulin, A., Andreev, A., Hardin, C., Dadashi, R., Hussenot, L.: Gemma 3 technical report (2025), https://arxiv.org/abs/2503.19786

Hsieh, Y.G., Hsieh, C.Y., Yeh, S.Y., Béthune, L., Ansari, H.P., Vasu, P.K.A., Li, C.L., Krishna, R., Tuzel, O., Cuturi, M.: Graph-based captioning: Enhancing visual descriptions by interconnecting region captions (2025), https://arxiv.org/abs/2407.06723

jackyhate: text-to-image-2m: A dataset on hugging face. https://huggingface.co/datasets/jackyhate/text-to-image-2M (2025), accessed: 2025-05-21

Jia, C., Yang, Y., Xia, Y., Chen, Y.T., Parekh, Z., Pham, H., Le, Q., Sung, Y.H., Li, Z., Duerig, T.: Scaling up visual and vision-language representation learning with noisy text supervision. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 139, pp. 4904–4916. PMLR (18–24 Jul 2021), https://proceedings.mlr.press/v139/jia21b.html

Li, J., Li, D., Savarese, S., Hoi, S.: BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In: Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., Scarlett, J. (eds.) Proceedings of the 40th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 202, pp. 19730–19742. PMLR (23–29 Jul 2023), https://proceedings.mlr.press/v202/li23q.html

Li, J., Li, D., Xiong, C., Hoi, S.: BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., Sabato, S. (eds.) Proceedings of the 39th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 162, pp. 12888–12900. PMLR (17–23 Jul 2022), https://proceedings.mlr.press/v162/li22n.html

Li, X., Tu, H., Hui, M., Wang, Z., Zhao, B., Xiao, J., Ren, S., Mei, J., Liu, Q., Zheng, H., Zhou, Y., Xie, C.: What if we recaption billions of web images with llama-3? (2024), https://arxiv.org/abs/2406.08478

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) Computer Vision – ECCV 2014. pp. 740–755. Springer International Publishing, Cham (2014)

openai: clip-vit-large-patch14: A pretrained model by openai on hugging face. https://huggingface.co/openai/clip-vit-large-patch14 (2021), accessed: 2025-05-21

Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: Meila, M., Zhang, T. (eds.) Proceedings of the 38th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 139, pp. 8748–8763. PMLR (18–24 Jul 2021), https://proceedings.mlr.press/v139/radford21a.html

Shazeer, N., Stern, M.: Adafactor: Adaptive learning rates with sublinear memory cost. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80, pp. 4596–4604. PMLR (10–15 Jul 2018), https://proceedings.mlr.press/v80/shazeer18a.html

Singla, V., Yue, K., Paul, S., Shirkavand, R., Jayawardhana, M., Ganjdanesh, A., Huang, H., Bhatele, A., Somepalli, G., Goldstein, T.: From pixels to prose: A large dataset of dense image captions (2024), https://arxiv.org/abs/2406.10328

Thapliyal, A.V., Pont-Tuset, J., Chen, X., Soricut, R.: Crossmodal-3600: A massively multilingual multimodal evaluation dataset (2022), https://arxiv.org/abs/2205.12522

TinyLlama: Tinyllama-1.1b-chat-v1.0: A pretrained model by tinyllama on hugging face. https://huggingface.co/TinyLlama/TinyLlama-1.1B-Chat-v1.0 (2025), accessed: 2025-05-21

Tschannen, M., Gritsenko, A., Wang, X., Naeem, M.F., Alabdulmohsin, I., Parthasarathy, N., Evans, T., Beyer, L., Xia, Y., Mustafa, B., Hénaff, O., Harmsen, J., Steiner, A., Zhai, X.: Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features (2025), https://arxiv.org/abs/2502.14786

Yu, J., Wang, Z., Vasudevan, V., Yeung, L., Seyedhosseini, M., Wu, Y.: Coca: Contrastive captioners are image-text foundation models. Transactions on Machine Learning Research (2022), https://openreview.net/forum?id=Ee277P3AYC

Zhai, X., Mustafa, B., Kolesnikov, A., Beyer, L.: Sigmoid loss for language image pre-training. In: 2023 IEEE/CVF International Conference on Computer Vision (ICCV). pp. 11941–11952 (2023)

Appendix

A Selenium orchestration

Algorithm 1 Distributed Colab Orchestration via Selenium Automation

Input: Global start index $start\_idx$; environment credentials $\{EMAIL_i, PASSWORD_i\}_{i=1}^{10}$; notebook IDs $\{NOTEBOOK\_ID_i\}_{i=1}^{10}$.

Output: All shards processed and merged into a single 1M-row Parquet dataset uploaded to Hugging Face.

for $i \leftarrow 1$ **to** 10 **do**
  Launch Chrome driver instance D_i with user profile i. Navigate to Colab. Log in using credentials $(EMAIL_i, PASSWORD_i)$. Open notebook with ID $NOTEBOOK\_ID_i$.
end

foreach $i \in \{1, \ldots, 10\}$ **do**
  Locate the notebook cell marked `[config-cell]`. Insert shard parameters for chunk $start\_idx + 100{,}000 \cdot (i-1)$. Execute all cells sequentially up to the `[upload-chunk]` marker. Wait until the output contains `[chunk-complete]`.
end

Main session (session 1): Download all 10 uploaded Parquet files. Merge them into a single 1M-row dataset. Upload the merged dataset to Hugging Face.

for $i \leftarrow 1$ **to** 10 **do**
  Execute the `[delete-runtime]` cell in session i. Close Chrome driver D_i.
end

B Dataset creation pipeline

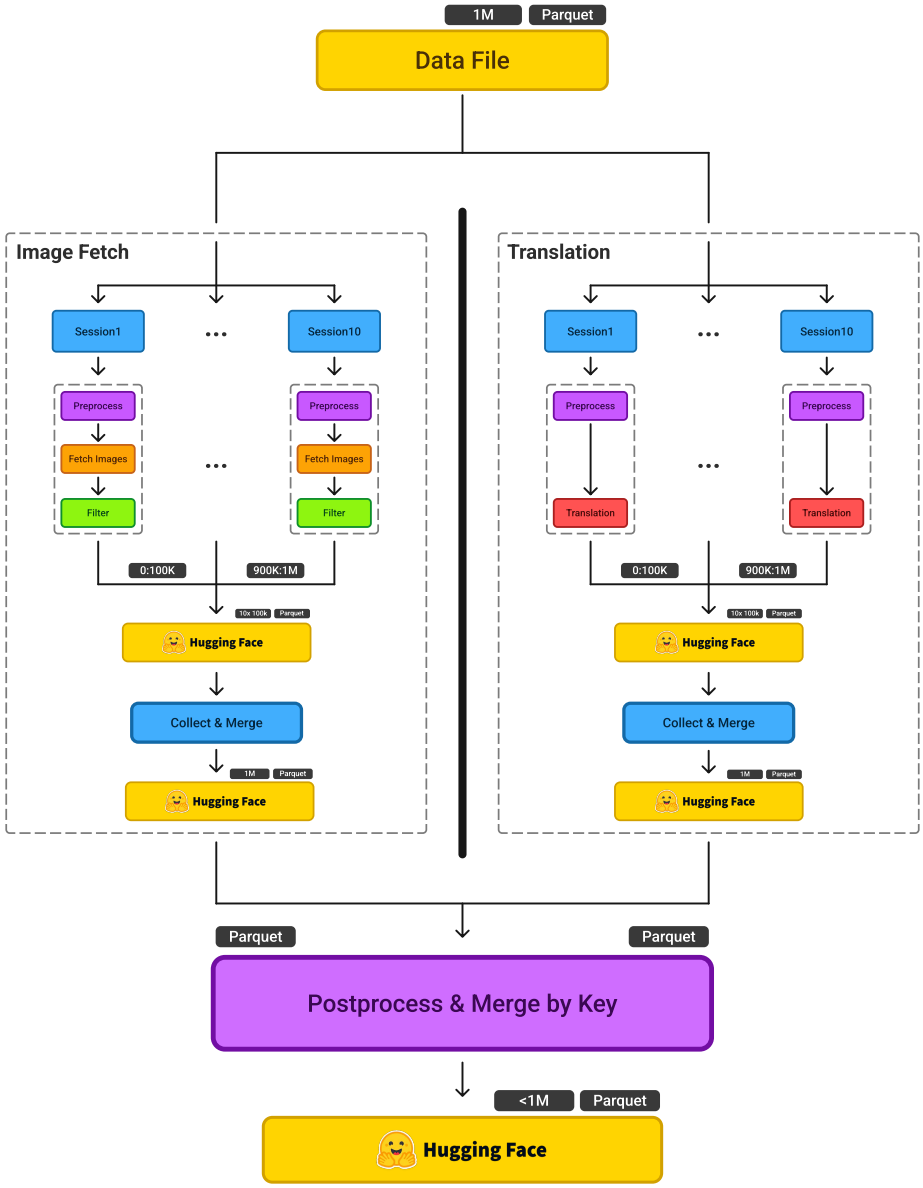


Fig. 2. High-level overview of the dataset pipeline.

C Silver standard image-text retrieval on text-to-image-2M

Table 8. Zero-shot image-text retrieval (Hungarian) results on the text-to-image-2M (jackyhate, 2025) subset.

Model	R@1	R@3	R@5	R@25	R@100	NDCG@1	NDCG@10	NDCG@100	MRR
MangaliCa (ours)	41.5%	62.7%	72.6%	91.7%	98.7%	41.5%	61.0%	64.6%	0.55
CLIP-L	19.5%	24.7%	26.9%	35.5%	48.3%	19.5%	24.4%	28.0%	0.24
SigLIP	47.3%	61.3%	66.7%	82.6%	94.1%	47.3%	60.1%	64.2%	0.57
SigLIP2	39.2%	52.3%	58.1%	77.6%	89.8%	39.2%	52.4%	57.0%	0.49
M-CLIP	**56.6%**	**76.5%**	**84.7%**	**95.7%**	**99.1%**	**56.6%**	**73.9%**	**75.7%**	**0.68**

D Gold standard image-text retrieval

Table 9. Zero-shot image-text retrieval (Hungarian) results on the short-form gold standard XM3600 dataset.

Model	R@1	R@3	R@5	R@25	R@100	NDCG@1	NDCG@10	NDCG@100	MRR
MangaliCa (ours)	11.3%	22.5%	28.9%	53.8%	76.9%	11.3%	23.4%	31.4%	0.20
SigLIP	26.3%	39.1%	45.4%	63.4%	79.6%	26.3%	38.7%	44.2%	0.35
SigLIP2	12.2%	19.0%	21.8%	35.0%	52.5%	12.2%	19.2%	24.2%	0.17
CLIP	7.0%	9.0%	11.0%	18.0%	29.6%	7%	10.0%	13.0%	0.09
M-CLIP	**40.0%**	**59.2%**	**67.2%**	**88.5%**	**97.5%**	**40.0%**	**58.1%**	**62.5%**	**0.53**

E TinyLlama 1.1B architecture split

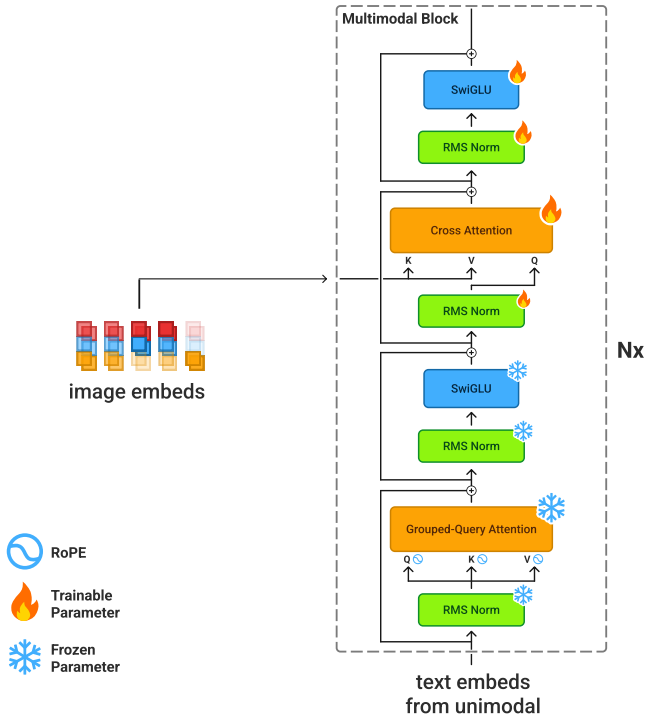


Fig. 3. Architecture overview of the Multimodal part.

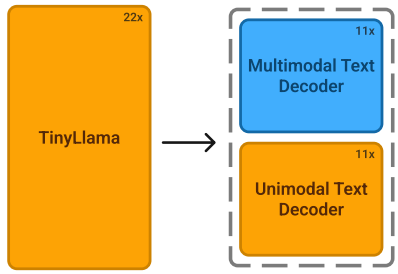


Fig. 4. TinyLlama 1.1B architecture split into two equal halves: the unimodal decoder for text-only processing and the multimodal decoder with a randomly initialized cross-attention for image grounding.

F Simulation results

HU: Festői kilátás egy hegységre, tiszta kék égbolttal és bolyhos fehér felhőkkel.

EN: A scenic view of a mountain range with a clear blue sky and fluffy white clouds.

HU: A kép egy szoba hangulatos sarkát ábrázolja, ahol egy fából készült könyvespolc tele van különféle könyvekkel és díszítőelemekkel.

EN: A image captures a cozy corner of a room featuring a wooden bookshelf filled with various books and decorative items.

HU: Egy kis vitorlás lebeg a nyugodt óceánon, felette tiszta kék ég.

EN: A small sailboat with floating on a calm sea with a clear blue sky above.

HU: Modern fürdőszobai mosdó fehér porcelán mosdóval és fa szekrénytalppal.

EN: A modern bathroom sink with a white porcelain basin and a wooden cabinet base.

HU: Nyugodt tengerparti jelenet naplementekor, tiszta égbolttal, amely kékről narancssárgára változik. A nap alacsonyan van a horizonton

EN: A serene beach scene at sunset with a clear sky transitioning from blue to orange hues. The sun is low on the horizon

Fig. 5. Selected Qualitative Results, where MangaliCa performed well during image captioning.

HU: A képen a férfi látható, aki Amerika Kapitányynak öltöz a Marvel Comics \"Amerika Kapitány: A első bosszúálló" című filmjéb. Aiem öltönyt visel, piörös ékezetek

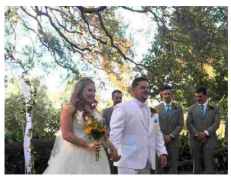
EN: A image features a man with as Bat America, the Marvel Cin movie \"Captain America: The Winter Avenger\". He is wearing a blue suit with red accents, a white star on his chest. The suit has short hair hair and is looking directly to the side with a serious expression on his

HU: Fe képen egy fehér kerámia bögre látható, amelysején élénk illusztrációval. Az illusztráció egy szűs jájjelenetet ábrázol, szatal lúval, aki egyokos tsvény

EN: A image showcases a white ceramic mug with a vibrant illustration on its side. The illustration depicts a youngene scene scene with a woman girl sitting along a pathy path towards a maj tree.er. The boy isries a basketpack and his back, holds a

HU: Egy bar barna klyökkutya hajlékony fülekkel és aranyos arckifejezéssel álll egy zja zöld pázsiton. A kölyökkutya fkér mellkasaú, és köz,na sz

EN: A brown brown puppy with floppy ears and a cute expression is standing on a lush green lawn. The puppy has a sh chest and p looking directly at the camera with a big, eyes. The grass features a clear blue sky, a few clouds, and there grass is v v



HU: A kép egy esküvői szertartásnak tűnő örömteli pillanatot örökít meg. Az előtérben két menyasszony és a vőlegény csymás kezét fogva, a vyass

EN: A image captures a joyfarming scene at two individuals who what appears to be a wedding day. The b on the left, dressed in a white bridal gown with la long vequet of flowers, is next their back turned towards the camera. their attention seemingly drawn towards something outside of

Fig. 6. Randomly Sampled Qualitative Results.

Beszédtechnológia

Időskori enyhe demencia és Alzheimer-kór felismerése x-vektor segítségével

Halász Dávid Péter[1], Kálmán János[2], Hoffmann Ildikó[2,3], Kiss Gábor[1]

[1] Távközlési és Mesterséges Intelligencia Tanszék, Villamosmérnöki és Informatikai Kar,
Budapesti Műszaki és Gazdaságtudományi Egyetem,
Műegyetem rkp. 3., H-1111 Budapest, Hungary

[2] Szegedi Tudományegyetem, Pszichiátriai Klinika

[3] ELTE Nyelvtudományi Kutatóközpont

kiss.gabor@vik.bme.hu

Kivonat: A demencia, és különösen az Alzheimer-kór, egyre jelentősebb társadalmi probléma, amely jelentős kihívásokat jelent a diagnosztika és az ellátás területén. Kutatásunkban a demencia diagnosztikájának beszédalapú támogatásának lehetőségét vizsgáltuk. Háromféle gépi tanuló modellt – Support Vector Machine (SVM), K-Nearest Neighbors (KNN) és Random Forest (RF) – alkalmaztunk, a beszédfelvételekből kinyert x-vektor jellemzőkön, hogy elkülönítsük az egészséges (HC), az enyhe kognitív zavarral diagnosztizált (MCI) és az enyhe Alzheimer-kórral diagnosztizált (MAD) személyeket. Az általunk vizsgált adatbázisban minden személyhez háromféle beszédminta állt rendelkezésre (azonnali felidézés - AF, előző napi narráció - ENN, késleltetett felidézés - KF). Kutatásunkban arra a kérdésre kerestük a választ, hogy az x-vektor technológia segítségével mennyiben különíthetők el a HC, MCI és MDA személyek és az egyes beszédtípusok (AF, ENN, KF) közül melyik a legalkalmasabb a beszédalapú diagnosztika automatikus támogatására. Legjobb eredményt, 58,7%-os pontosságot, SVM gépi tanuló eljárással a késleltetett felidézés (KF) típusú beszédmintákon kaptuk.

1 Bevezetés

A demencia egy krónikus, progresszív állapot, amelyet kognitív hanyatlás jellemez, jelentősen befolyásolva a mindennapi életet, a viselkedést és a gondolkodást. A demenciát legalább hat hónapig fennálló memóriavesztésként, kommunikációs nehézségekként, dezorientációként vagy a döntéshozatal károsodásaként definiálják. A demencia világszerte közel 55 millió embert érint, és ez a szám várhatóan 2050-re háromszorosára emelkedik (World Health Organization, 2021). A genetikai, neurológiai, környezeti és életmódbeli tényezők szerepet játszanak a demencia kialakulásában, így kezelése összetett kihívást jelent. A demencia diagnózisához több, tartós kognitív tünet jelenléte szükséges, bár a korai kezelés az enyhébb tünetekkel rendelkezők számára is előnyös lehet. A tünetek súlyossága és progressziója a betegség típusától, az

egyéntől és a betegség stádiumától függően változik (Better, 2023). Bár a demencia nem gyógyítható, a korai diagnózis jelentősen javíthatja az életminőséget és lassíthatja a tünetek progresszióját. A kezelési lehetőségek közé tartoznak a gyógyszeres terápiák, a kognitív stimulációs terápiák és a pszichoszociális támogatás. Egyes betegek olyan tüneteket mutatnak, amelyek rezisztensek a gyógyszeres kezelésre, ami összetettebb kezelési megközelítéseket igényelhet, például multidiszciplináris kezelési terveket (Better, 2023).

Az Alzheimer-kór a demencia leggyakoribb formája, amely a memória, a gondolkodási képességek, valamint a mindennapi életviteli funkciók progresszív romlásával járó neurodegeneratív betegség. Az Alzheimer-kór főként az idősebb korosztályt érinti, de ritkán fiatalabb életkorban is megjelenhet. A betegség során az agyban jellemzően abnormális fehérje-lerakódások, úgynevezett béta-amiloid plakkok és tau-fehérje által alkotott neurofibrilláris kötegek alakulnak ki, amelyek az idegsejtek pusztulását eredményezik (Better, 2023).

Az Alzheimer-kór kezdeti tünetei között szerepel a rövid távú memóriazavar, például nehézség az új információk megjegyzésében. A betegség előrehaladtával további kognitív problémák, kommunikációs nehézségek, a tájékozódási képesség elvesztése és hangulatváltozások jelentkeznek. A betegség végstádiumában a betegek gyakran már nem képesek önálló életvitelre és folyamatos gondozást igényelnek. Az Alzheimer-kór pontos oka még nem ismert teljes mértékben, azonban genetikai tényezők, életmódbeli hatások, valamint az öregedési folyamat együttesen hozzájárulnak kialakulásához. A betegség kezelésében jelenleg alkalmazott gyógyszerek a tünetek enyhítésére, a betegség progressziójának lassítására és a betegek életminőségének javítására irányulnak. A gyógyszeres kezelés mellett jelentős szerepe van a kognitív tréningeknek, pszichológiai támogatásnak és a megfelelő szociális környezet biztosításának. A korai diagnózis kulcsfontosságú, mivel ez lehetővé teszi a tünetek hatékonyabb kezelését és a beteg életminőségének hosszabb ideig tartó megőrzését (Better, 2023), (World Health Organization, 2021).

A demencia jelentős hatással van a beszédre, befolyásolva a nyelvi kifejezést, a beszéd folyamatosságát és a kommunikáció hatékonyságát. Demenciában szenvedő egyéneknél gyakran előfordulnak szótalálási nehézségek, megfigyelhető leegyszerűsödött szókincs, és a beszéd lassabbá vagy zavarosabbá válása. Emellett a beszed minőségének romlása is megfigyelhető. Ezek a tünetek feltehetőleg a memória és a kognitív funkciók általános romlását tükrözik. A betegek beszédét általában az ismétlések, a bizonytalanság és a beszédben megjelenő szünetek jellemzik, melyek mind a kognitív folyamatok hanyatlásának következményei (Vigo és mtsai, 2022).

Így a beszéd egy lehetséges biomarkere lehet az enyhe kognitív zavarnak vagy az enyhe Alzheimer kórnak. Azonban a beszédnek nagy a természetes varianciája (Olaszy, 2010), így a beszédproduktumot nem csak az esetleges enyhe demencia jelenléte befolyásolja, hanem a beszélő érzelmi állapota, fáradtsága, neme, kora, fiziológiai állapota. Így a feladat, hogy megtaláljuk a demencia által okozott változásokat igen nehéz. Klasszikusan a feladat megoldásához akusztikai-fonetikai jellemzőket kellene kinyerni a beszéd mintákból, amelyeket befolyásol a demencia, majd ezek segítségével lehetne gépi tanuló eljárásokkal modelleket létrehozni a demencia felismeréséhez. Azonban napjainkban elterjedtek a különböző mély neurális hálózatokon alapuló jellemző kinyerő eljárások (Jenei és mtsai., 2022), (Egas-López és mtsai., 2022), (Vetráb és Gosztolya, 2023) így jelen kutatásban mi is ilyet használtunk, ennek

az előnye, hogy a jellemzők kinyerése egyszerű és általában az adott betegség beszédre gyakorolt hatásának típusától függetlenül, a kinyert jellemzők alapján hasonló vagy jobb minőségű modellek hozhatók létre a klasszikus jellemző kinyerési eljárásokkal szemben. A hátránya, hogy viszont keveset vagy szinte semmit sem tudunk meg a vizsgált betegség beszédre gyakorolt hatásáról.

Vizsgálataink során arra a kérdésre kerestük a választ, hogy a beszédminták x-vektor-ral (Snyder és mtsai., 2017 és 2018) történő reprezentálása esetében, különböző gépi tanuló eljárásokkal megalkotott modellekkel, mennyiben lehetséges az egészséges személyek, az enyhe kognitív zavarral diagnosztizált személyek és az enyhe Alzheimer kórral diagnosztizált személyek elkülönítése. Továbbá, hogy melyik beszédtípus (azonnali felidézés, előző napi narráció, késleltetett felidézés) felhasználásával hozható létre nagyobb pontosságú modell.

2 Módszerek

A vizsgálatainkat a Szegedi Tudományegyetem Pszichiátriai Klinikája által létrehozott beszédadatbázison végeztük. A beszédmintákból előtanított modell alkalmazásával x-vektor-okat nyertünk ki. Egymásba ágyazott keresztvalidációval végeztük el a modellek tanítását, optimalizálását és tesztelését. A modelleket az egészséges személyek, az enyhe kognitív zavarral diagnosztizált személyek és az enyhe Alzheimer kórral diagnosztizált személyek elkülönítésére tanítottuk. A kiértékelés során elsődleges metrikának a pontosságot használtuk fel, továbbá vizsgáltuk az átlagos precizitást, fedést és f1 értékeket. Az adatbázisban minden személytől három beszédminta állt rendelkezésre (azonnali felidézés, előző napi narráció, késleltetett felidézés), így három vizsgálatot végeztünk a beszédminta típusától függően.

2.1 Adatbázis

A kutatás során a Szegedi Tudományegyetem Pszichiátriai Klinikáján rögzítettet beszédfelvételekkel dolgoztunk. A rögzítés digitális diktafonnal történt, külső mikrofon használatával. Az alanyok spontán beszédét rögzítették, minden személytől háromféle beszédmintát gyűjtöttek, amelyek különböző beszédfeladatokat tartalmaznak. Ezenkívül további teszteket is elvégeztettek az alanyokkal, hogy objektív mérőszámokkal jellemezzék kognitív állapotukat, Mini-Mental State Examination (Folstein és mtsai., 1975), Clock Drawing Test (Freedman és mtsai., 1994) és Alzheimer's Disease Assessment Scale (Rosen és mtsai., 1984). Az alanyok először egy némafilmet tekintettek meg, amelyet azonnal el kellett mesélniük (azonnali felidézés - AF). Ezt követően a tegnapi napjuk eseményeit kellett röviden elmesélniük (előző napi narráció - ENN), végül egy második némafilmet néztek meg, majd egy perc szünet után kellett elmesélniük a történteket (késleltetett felidézés - KF). A beszélőket orvosi diagnózis alapján három különböző kategóriába sorolták: kontroll csoport (HC): egészséges, kognitív károsodást nem mutató alanyok, enyhe kognitív zavar (MCI): enyhe kognitív zavarban szenvedő alanyok, enyhe Alzheimer (MAD): enyhe stádiumú Alzheimer kórban szenvedő alanyok.

Az adatbázisban összesen 75 beszélőtől vannak beszédfelvételek, mindegyik csoportban pontosan 25 alany található és minden alanytól mindhárom típusú beszédfelvétel rendelkezésre áll, így összesen az adatbázis 225 beszédfelvételből áll. Sajnos a felvételek technikai minősége változó, ami egy jelentős limitációja lehet jelen kutatásnak, mivel a gépi tanuló modellek érzékenyek az eltérő felvételi minőségekre, az adatbázis részletesebb leírása megtalálható Vincze és mtsai. 2021-es cikkében (Vincze és mtsai., 2021).

2.2 Korábbi eredmények

Az adatbázist felhasználva több kutatási eredmény is található. Egas-López és mtsai. hasonlóan hozzánk azt vizsgálták, hogy mennyiben lehetséges a HC, MCI, és MAD elkülönítése, illetve, hogy mely felvétel típussal képesek jobb eredményt elérni, azonban i-vector jellemző kinyerő eljárást alkalmaztak. A vizsgálat során legjobb eredményeket a késleltetett felidézés típusú felvételeken kapták, bár az eltérés nem volt jelentős a különböző típusú felvételeken: AF-en 42,7%-os pontosságot, ENN-en 41,3%-os pontosságot, míg a KF-en 46,7%-os pontosság értek el (Egas-López és mtsai., 2019).

Egy másik kutatásban Egas-López és mtsai. x-vektor jellemző kinyerő eljárással is vizsgálták az adatbázist, ám ekkor a HC és MCI elkülönítését vizsgálták az ENN (előző napi narráció) típusú felvételeken. Legjobb elkülönítést 70%-os pontossággal tudták elvégezni (Egas-López és mtsai., 2021). Kiss-Vetráb és mtsai. is elvégezték ugyanezt a vizsgálatot szekvenciális autoenkóder alkalmazásával és nagyon hasonló 72%-os pontosságot tudtak elérni (ez jelen adatbázisban azt jelenti, hogy eggyel több helyes elkülönítést értek el) (Kiss-Vetráb és mtsai, 2022).

Külföldi adatbázison elért eredmények is találhatóak x-vektor jellemző kinyerő eljárás alkalmazásával. Haulcy és Glass SVM osztályozót alkalmazva 64%-os pontosságot értek el az ADReSS adatbázison (Haulcy és Glass, 2021), Campbell és mtsai. illetve Pappagari és mtsai. ugyanezen adatbázison 73%-os pontosságot értek el (Campbell és mtsai., 2021), (Pappagari és mtsai., 2021), míg Wang és mtsai. 75%-ot tudtak elérni az adatbázison (Wang és mtsai., 2021). Subramanian és mtsai. általuk gyűjtött adatbázisoson 72%-os pontosságot értek el (Subramanian és mtsai., 2024). További összefoglaló eredmények találhatók Tripathi és Kumar 2024-es cikkében többféle jellemző kinyerő eljárás és osztályozó eljárás alkalmazása esetében (Tripathi és Kumar, 2024).

2.3 Leíró jellemzők

A beszédfelvételekből a jellemzők kinyerése x-vektor jellemző kinyerő módszer segítségével valósítottuk meg, amely mély neurális hálózatokon alapuló módszer (Snyder és mtsai., 2017 és 2018). A módszer segítségével minden felvételt 512 dimenziójú vektorral jellemeztünk. A módszert eredetileg beszélő azonosításra fejlesztették ki, de azóta több kutatás is bizonyította, hogy jól alkalmazható jellemző kinyerő eljárásként betegségek beszéd alapú diagnosztikájának támogatásához (Jenei és mtsai., 2022) (Egaz Lopez és mtsai., 2021 és 2022).

Az x-vektor jellemző kinyerő eljárás előnye, hogy a kimeneti x-vektor, főleg olyan információkra összpontosít, amelyek az emberi hallás számára relevánsak lehetnek, elhagyva a hangmintákba bekerülő zajokat vagy redundáns tulajdonságokat (Snyder és mtsai., 2018). Az x-vektor jellemző kinyerő eljárás további előnye, hogy képes eltérő hosszúságú hangfelvételekből fix dimenziójú jellemzővektort előállítani.

Az x-vektor jellemzővektoroknak az előállításához egy mély neurális hálózatot használnak, amelyben az adatáramlás egyirányú, így az információk visszacsatolás nélkül, kizárólag a bemenettől a kimenet felé haladnak. A hálózat struktúrája „end-to-end" rendszeren alapszik, ami miatt „in-domain" adatokra van szükség a tanítás során, tehát az adatoknak összhangban kell lenniük az alkalmazási területtel. Ennek kezelése érdekében többosztályos kereszt-entrópia célfüggvényt használnak, továbbá a beágyazások összehasonlítására egy külön tanított PLDA (Probabilistic Linear Discriminant Analysis) modellt alkalmaznak (Snyder és mtsai., 2018).

A neurális hálózat első-három rétege keretszinten a beszéd kisebb időbeli szeleteit dolgozza fel, fokozatosan növelve az időbeli kontextust. Ezután a negyedik réteg összesíti a keretszintű információkat, átlagot és szórást számol így már beszédminta szinten jellemzi a felvételt, ezt követi további három, teljesen összekötött réteg, és ezeket a rétegeteket nevezzük x-vektor-nak. A végeredmény egy kompakt, univerzális reprezentáció, amely alkalmas lehet a bemenet különböző beszéd alapú osztályozási vagy regressziós feladatokra, például beszélő-azonosításra vagy betegségek diagnosztizálásának támogatására.

A vizsgálatinkhoz használt x-vektor modellnek egy nyílt forráskódú, előre betanított neurális hálózatot használtunk, amelyet a Hugging Face platformon keresztül értünk el és a Snyder féle architektúrát implementálja (https://huggingface.co/speechbrain/spkrec-xvect-voxceleb). A Hugging Face több a közösség által karbantartott és kutatási célra optimalizált modellt kínál, így megbízható és forrást jelentett a jellemzők kinyerésére. Ez lehetővé tette, hogy a beszédfeldolgozáshoz szükséges jellemzővektorokat gyorsan és hatékonyan állítsuk elő, anélkül, hogy a teljes neurális hálózatot újra kellett volna tanítani, bár az x-vektor neurális hálózat magyar nyelvre történő adaptációja javíthatja az eredményeket pár százalékkal, de nem szükségszerűen (Egaz Lopez és mtsai., 2021).

A kinyert jellemzőket Z-transzformáció segítségével normalizáltuk. A jellemzők normalizációja fontos lépés a gépi tanulást alkalmazó modellekben, mivel biztosítja, hogy a különböző magnitúdójú skálán lévő jellemzők ne zavarják össze a gépi tanulást. A Z-transzformáció alkalmazása során a jellemzők értékeit az átlagtól való eltérésük alapján, az adatok standard szórásával történő osztásával skálázzák át. Ennek következtében a jellemzők átlagosan nulla értékűek és egységnyi standard szórással rendelkeznek.

2.4 Modell tanítása, optimalizálása és tesztelése

A vizsgálatok elvégzéséhez Rapidminer Studio-t használtunk, a program használata egyszerű, mivel vizuális felületen keresztül lehet különböző adatelemzési és gépi tanuló modellek létrehozását elvégezni (Dwivedi és mtsai., 2016). A keretrendszeren belül háromféle gépi tanuló eljárást alkalmaztunk: Support Vector Machines (SVM) (Cortes és Vapnik, 1995) LibSVM (Chang és Lin, 2011) implementációban, Random

Forest (RF) (Breiman, 2001) és K-Nearest Neighbours (KNN) (Cover és Hart, 1967) gépi tanuló eljárásokat. A betegségek beszédalapú orvosi diagnosztizálásnak támogatásban az egyik legelterjedtebb gépi tanuló módszer az SVM, mivel elméletileg lehetséges ezzel a módszerrel tetszőleges probléma statisztikai értelemben vett legjobb modelljét megalkotni, ugyanakkor a gyakorlatban kis mennyiségű adatokon is elég jól tanítható, azonban a használata nem szükségszerű, általában hasonlóan jó eredményeket lehet elérni RF-el vagy egyéb neurális hálózatokkal is (Low és mtsai., 2020). Pont emiatt alkalmaztunk más gépi tanuló eljárásokat is, hogy megvizsgáljuk, hogy a kapott eredmények mennyiben függenek az alkalmazott gépi tanuló eljárástól. Előzetesen arra számítottunk, hogy az SVM-el és a RF-el hasonló de jobb eredményeket kapunk, mint a KNN-nel.

Az adatbázis kis mérete miatt (75 felvétel típusonként) egymásba ágyazott keresztvalidációt (nested cross-validation) alkalmaztunk (Varma és Simon, 2016). Ennek a lényege, hogy egymásba ágyazva kettős keresztvalidációt hajtunk végre. A külső keresztvalidáció során különítjük el a teszt halmazt és a modell létrehozásához alkalmazott halmazt, majd a belső keresztvalidációban végezzük el a modell optimalizálását és tanítását és a kapott modellt teszteljük a külső keresztvalidációban. Legvégül a kapott teszt eredményeket aggregáljuk. A módszer előnye, hogy a vizsgált eljárás tesztelését a teljes adatbázison képesek vagyunk elvégezni, így statisztikai értelemben pontosabb becslést kapunk a vizsgált módszer teljesítményéről. Másfelől a modell létrehozásához is több mintát tudunk megtartani, ami kevés minta esetén a gépi tanuló eljárások teljesítményében nagy jelentőséggel bír. Természetesen az eljárás teljesíti az a modellek létrehozásának azon alapkövetelményét, miszerint a tanító, az optimalizáló és a teszt halmazok egymástól függetlenek, így a kapott eredmény valóban a módszerünk valós teljesítményéről ad becslést. A hátránya, hogy jelentősen megnövel a vizsgálat idejét, illetve esetlegesen különböző modelleket alkot, így nem egy konkrét modell teljesítményét becsüli, hanem inkább magát az alkalmazott eljárást. Jelen kutatásban a külső keresztvalidációnál 25-ös fold számot választottunk osztályonként kiegyenlítve, így minden fold-ba pontosan egy HC, MCI és MAD személy beszédmintája került, míg a belső keresztvalidáció esetében 24-et választottunk osztályonként kiegyenlítve, így itt is minden fold-ba pontosan egy különböző osztályú személy került.

A modell optimalizálása során a bemeneti jellemzővektort optimalizáltuk inkrementális kiválasztás (forward selection) eljárással. Az eljárás egy jellemző csökkentő eljárás, erre azért volt szükség, mivel az x-vektor 512 dimenziójú volt, míg a teljes adatbázisban felvétel típusonként csupán 75 felvétel állt rendelkezésre. A gépi tanuló eljárások statisztikai alapúak, így, ha kevés tanító minta áll a rendelkezésünkre célszerű lehet a leíró jellemzővektor dimenziójának a csökkentése (Kiss és Vicsi, 2017).

Az eljárás az üres jellemzővektorból indul ki. Az i-dik lépésben rendelkezésére áll az i-1 dimenziójú az eljárás szerint optimális jellemzővektor. Majd egyesével kiértékeli az eljárás, hogy a még fel nem használt jellemzők közül, az i-1 dimenziójú jellemzővektorhoz melyik hozzáadásával lehet a legjobb eredményt kapni, így az i-dik lépés végén megkapjuk az i dimenziójú az eljárás szerint optimális jellemzővektort. Az eljárás esetén lehetséges különböző leállási feltételeket megadni, ezáltal csökkentve a futási időt. Az eljárás előnye, hogy a jellemzők számától függően négyzetes komplexitású, így gyorsnak nevezhető és a gyakorlatban hatékony, képes lehet akár az optimális jellemzővektor megtalálásra. Azonban, ha igaz, hogy a jellemzőknek van

egy olyan részhalmaza, amik együttes alkalmazása szükséges a legjobb eredmény eléréshez, azonban a részhalmazaik semmiben sem javítják a modell teljesítményét, akkor az eljárás képtelen lehet megtalálni az optimális jellemzővektort. Jelen kutatásban az eljárást úgy alkalmaztuk, hogy ha 3 lépésig nem javult az eredmény, akkor álljon le az eljárás.

Az egyes gépi tanuló eljárásokat a Rapidminer Stúdió által javasolt alapértelmezett paraméter értékek mentén alkalmaztuk. SVM esetén SVM-C típust használtunk rbf kernellel és a gamma és cost értékeknek 1-et adtunk meg. RF esetén 100 fát alkalmaztunk, gain ration hasítási kritériumot és 10-es maximális mélységet adtunk meg. KNN esetén k-t 5-nek választottuk.

2.5 Kiértékelési metrikák

Az eljárás tesztelése során megkaptuk az SVM, az RF illetve a KNN tévesztési mátrixát. Ezekből kiszámítottuk a pontosságot (accuracy), az átlagos precizitást (precision), az átlagos felidézést (recall) és az átlagos f1 értéket (f1 score). Elsődleges metrikának a pontosságot választottuk, és az optimalizálás során is azt alkalmaztuk. Mivel az adatbázis osztályonként kiegyenlített volt, így a pontosság egy megfelelő metrika, ugyanakkor ebből következik, hogy az átlagos felidézés és pontosság minden esetben meg fog egyezni. De az átlagos precizitás illetve az átlagos f1 érték további információt közöl arról, hogy az adott modell az osztályonként mennyire egyenletesen teljesít.

3 Eredmények

A kutatás során alkalmazott gépi tanulási modellekkel (SVM, KNN, RF), és a különböző típusú felvételeken azonnali felidézés (AF), előző napi narráció (ENN) és késleltetett felidézés (KF) végeztünk osztályozási vizsgáltokat. Az elemzés célja az volt, hogy azonosítsuk, milyen pontossággal és hatékonysággal lehet megkülönböztetni egymástól az egészséges személyek (HC), az enyhe (időskori) kognitív zavarral diagnosztizált személyek (MCI), valamint az enyhe Alzheimer-kórral diagnosztizált személyek (MAD) beszédfelvételeit. A kapott eredményeket a 1. táblázatban foglaljuk össze.

1. Táblázat: SVM, RF és KNN optimalizált modellekkel kapott eredmények a HC, MCI és MAD elkülönítése esetében a felvétel típusától függően

Gépi tanuló eljárás	Felvétel típus	Pontosság	Átlagos precizitás	Átlagos felidézés	Átlagos f1 érték
SVM	**AF**	56,0%	55,7%	56,0%	55,7%
SVM	**ENN**	50,7%	51,4%	50,7%	50,6%
SVM	**KF**	***58,7%***	***58,7%***	***58,7%***	***58,7%***
RF	**AF**	50,7%	50,0%	50,7%	50,1%
RF	**ENN**	45,3%	45,4%	45,3%	45,3%
RF	**KF**	*53,3%*	*52,6%*	*53,3%*	*52,7%*
KNN	**AF**	46,7%	40,1%	46,7%	40,4%

KNN	**ENN**	40,0%	42,1%	40,0%	39,4%
KNN	**KF**	*50,7%*	*48,1%*	*50,7%*	*47,4%*

A gépi tanuló eljárásokat összehasonlítva (1. táblázat) a legjobb pontossági eredményeket az SVM-mel kaptuk (51-59%), második legjobbakat RF-el (45-53%) majd a legrosszabbakat KNN-nel (40-51%), ugyanakkor elképzelhető, hogy az egyes gépi tanuló eljárások paramétereinek optimalizálásival ezek a különbségek eltűnnének vagy mérséklődnének.

A felvétel típusokat összehasonlítva egyértelmű tendencia látszik. Legjobb eredményeket a késleltetett felidézésen (KF) kaptuk (51-59%), második legjobb eredményeket az azonnali felidézésen (AF) kaptuk (47-56%), míg a legrosszabbakat az előző napi narrációkon (ENN) kaptuk (40-51%). Továbbá érdekes, hogy míg a két felidézés típus (KF-AF) között 3-4% különbség volt gépi tanuló eljárás függvényében, addig az azonnali felidézés és az előző napi narráció (AF-ENN) között 5-7%, ami majdnem a duplája a KF-AF közti különbségnek, míg a késleltetett felidézés és az előző napi narráció (KF-ENN) között 8-11% volt, ami majdnem a háromszorosa a KF-AF közti különbségnek. Ezekből kijelenthető, hogy jelen adatbázis esetében, az általunk alkalmazott módszerekkel, a felidézés típusú feladatból kinyert jellemzőkkel magasabb osztályozási pontosságot lehetett elérni, vagyis ez a típusú beszédminta feltehetőleg hatékonyabb lehet az orvosi diagnózis támogatásához.

Az SVM és az RF esetében a pontosság és az átlagos f1 értékek hasonlóak voltak, amiből arra lehet következtetni, hogy az így kapott modellek hasonló teljesítménnyel képesek felismerni az egyes osztályokat, ezzel szemben a KNN esetén az f1 érték tipikusan alacsonyabb volt a pontosságnál, amiből arra lehet következtetni, hogy az adott modell valamelyik osztályt lényegesen alacsonyabb teljesítménnyel volt képes megtanulni. Ebben az esetben tévesztési mátrixokat tanulmányozva azt kaptuk, hogy a MCI-re való döntés átlagosan csupán 22,3%-os, ami 10%-al elmarad a várt 33%-tól. Ez érthető, hiszen a kognitív hanyatlás egy fokozatos jelenség így az egészséges időseket az enyhe kognitív zavartól szenvedő idősektől nem feltétlenül egyszerű elkülöníteni, illetve maga a beszéd nem feltétlenül képes tökéletesen elkülöníteni az egészséges időseket, az enyhe kognitív zavartól szenvedő idősektől, vagy az enyhe Alzheimer kórtól szenvedő idősektől sem.

Legjobb eredményt SVM gépi tanuló eljárással értük el a késleltetett felidézés típusú beszédmintákat felhasználva. Ebben az esetben a pontosság 58,7% volt, 2,7%-al magasabb, mint az azonnali felidézés típusú beszédmintákat felhasználva, ugyanakkor fontos megjegyezni, hogy ez a jelen adatbázisban azt jelentette, hogy kettő személlyel többet osztályoztunk helyesen. A legjobb eredmény tévesztési mátrixa a 2. táblázaban látható.

2. Táblázat: SVM gépi tanuló eljárással, a késleletet felizés (KF) beszédmintákon tanított optimalizált modell tesztelésének tévesztési mátrixa

	Prediktált HC	Prediktált MCI	Prediktált MAD	felidézés
Valós HC	**15**	7	3	60,0%
Valós MCI	5	**13**	7	52,0%
Valós MAD	4	5	**16**	64,0%

precizitás	62,5%	52,0%	61,5%

A tévesztési mátrixból megállapítható, hogy a modell hasonló teljesítménnyel volt képes felidézni az egyes osztályokat (HC: 60%, MCI: 52%, MAD 64%), legrosszabbul az MCI osztály esetében teljesített. Továbbá a precizitás érték szinte megegyezett a felidézési értékkel osztályonként (HC: 63%, MCI: 52%, MAD: 62%), amiből látható, hogy a modell mindhárom osztályra hasonló eséllyel döntött 32-35%, tehát egyik osztályt sem hagyta el vagy részesítette túlzott előnyben. Ezek a tulajdonságok Fontosak egy a gyakorlatban is használandó modell esetében. Maguk az eredmények nem tekinthetőek túl magasnak ugyanakkor egy előszűrő rendszerhez, egy állapotkövető rendszerhez vagy orvosi diagnosztika támogatásához akár használható is lehetne az általunk fejlesztett modell.

4 Összefoglalás

Az időskori demencia, mint például az enyhe kognitív zavar, vagy az enyhe Alzheimer kór, egyre jelentősebb népegészségügyi problémát jelent, amely a diagnosztika és az ellátás területén is komplex kihívásokat teremt. Bár a betegség jelenleg nem gyógyítható, a korai felismerés és az időben elkezdett kezelés jelentősen javíthatja a betegek életminőségét és lassíthatja a kognitív hanyatlást. Az ettől szenvedő személyek száma nominálisan és a népességen belüli arányaiban évről vére nő, részben a várható élettartam növekedésével részben az elöregedő társadalmak következtében. Így a betegség felismerése egy arányaiban egyre csökkenő rétegre hárul. Fontos lenne a diagnózis illetve az állapot nyomon követesének automatikus támogatása. Mind az enyhe kognitív zavar, mind az Alzheimer kór esetében jellemző a beszéd megváltozása, a beszéd minőségének romlása, így a beszéd egy alkalmas objektív biomarkere lehet a betegség automatikus felismerésének, súlyosság szerinti becslésének.

Kutatásunkban a Szegedi Tudományegyetem Pszichiátriai Klinikáján rögzítettet beszédadatbázissal dolgoztunk, amiben 25 egészséges idős személy (HC), 25 enyhe kognitív zavarral diagnosztizál személy (MCI), illetve 25 enyhe alzheimer kórral diagnosztizált személy (MAD) beszédmintái találhatók. Az adatbázis minden személytől három spontán jellegű beszédmintát tartalmaz, azonnali felidézés (AF), előző napi narráció (ENN) és késleltetett felidézés (KF).

Minden beszédmintát, x-vektor jellemző kinyerő eljárás segítségével, 512 dimenziójú jellemzővektorral (x-vektor) írtunk le. Az egyes jellemzőket z transzformáció segítségével normalizáltuk. Háromféle gépi tanulási modellt: Support Vector Machine (SVM), K-Nearest Neighbors (KNN) és Random Forest (RF) használtunk, hogy elkülönítsük a HC, MCI, illetve a MAD személyeket. Gépi tanuló eljárásonként (SVM/RF/KNN) illetve beszédminta típusonként (AF/ENN/KF) egymásba ágyazott keresztvalidációs (nested cross-validation) eljárást alkalmazva különböző optimalizált modelleket hoztunk létre. A modell optimalizálás során az 512 dimenziójú x-vektorok dimenzióját csökkentettük inkrementális kiválasztást (forward selection) megvalósító jellemzővektor kiválasztó algoritmussal.

A legjobb eredményt az SVM gépi tanuló eljárással, a KF típusú beszédmintákat felhasználva értük el 58,7%-os pontossággal. Az SVM modell stabilabb és pontosabb

osztályozást biztosított az egyes csoportok között, mint a KNN vagy a RF. Egas-López és mtsai. ugyanezen az adatbázison i-vector jellemző kinyerő eljárást alkalmazva, 46,7%-os pontosságot értek el. Így az általunk bemutatott módszer 25,7%-os relatív javulást mutatott az ő módszerükhöz képest, a konkrét adatbázis esetében ez azt jelenti, hogy az általunk bemutatott eljárás segítségével 9-cel több személyt ismertünk fel helyesen. Ugyanakkor mi is azt az eredményt kaptuk, hogy a legnagyobb pontosságot a KF típusú beszédfelvételeken lehet elérni, második legnagyobbat az AF-en, legrosszabbat pedig az ENN-en. Ugyanakkor nagyobb eltérést tapasztaltunk, mint Egaz-López és mtsai. (8% szemben a 4%-al) (Egas- López és mtsai, 2019). Ugyanakkor elképzelhető, hogy az AF és a KF közti különbséget nem a késleltetés okozta, hanem mivel a felvételek sorrendje mindig AF, ENN majd KF voltak, így az általános kimerültség jobban megviselte az MCI illetve az MAD személyeket szemben a HC személyekkel, és ez okozta a beszédnek a nagyobb fokú elváltozását ezáltal a jobb osztályozási eredményeket, de az kijelenthető az eredményeink alapján, hogy a felidézés alkalmasabb beszédtípus az előző napi narrációval szemben.

További két kutatással hasonlítottuk még össze az eredményeinket (Egas- López és mtsai, 2021) illetve (Kiss-Vetráb és mtsai., 2022), ahol szintén ezt az adatbázist alkalmazták, ugyanakkor csak az HC-MCI elkülönítést vizsgálták és csak az ENN beszédtípust használták, így 70-72%-os pontosságot értek el x-vektor illetve szekvenciális autoenkóder használatával. Az általunk bemutatott módszert nehéz egy az egyben összehasonlítani ezekkel az eredményekkel, ugyanakkor az SVM-el, az ENN beszédtípuson létrehozott optimalizált modellen kapott tesztelési eredményből elhagyva azokat a döntéseket amikor a modell MAD-ra döntött, illetve az eredetileg MAD-al diagnosztizált személyeket, mi 63,4%-os pontosságot értünk el, ami átlagosan 10,7%-os teljesítmény csökkenést jelent. Feltehetőleg ez abból adódik, hogy az általunk bemutatott modell három osztályos osztályozásra lett tanítva és optimalizálva.

Mindhárom gépi tanuló modell esetében a legrosszabb eredményt az enyhe kognitív zavar (MCI) esetében kaptuk (mind precizitás mind felidézés értékek esetében).

A vizsgálat kezdetén megfogalmazott kérdéseinkre a válasz, hogy jelen módszer segítségével legjobb eredményt a KF esetében lehet kapni, az 58,7%-os pontossággal. Önmagában ez a teljesítmény feltehetőleg nem elégséges automatikus diagnózis felállítására, ugyanakkor a beszédalapú előszűrésre vagy a diagnózis támogatására ígéretes lehetőséget kínál a demencia korai felismerése esetében. A képet tovább árnyalja, hogy bár a legjobb eredményt a KF esetében kaptuk, ugyanakkor feltehetőleg ez tart a legtovább, kevésbé természetes beszédtípus, így a gyakorlatban kevésbé jól alkalmazható. Ráadásul más betegségek esetében (depresszió, Parkinson kór, skizofrénia, szorongás, ALS, stb.), amik szintén felismerhetők beszédalapú módszerekkel) általában nem alkalmazzák ezt a típusú felvételt az ilyen irányú kutatások. Az ENN egy sokkal természetesebb, egyszerűbben megvalósítható, feltehetőleg rövidebb ideig tartó folyamat, amit ráadásul egy előszűrés (háziorvosi rendelés) vagy egy szakorvosi rendelés esetében be lehet építeni az alapvető protokollba, hiszen a legtöbb esetben amúgy is végeznek egy általános állapot felmérést, az előző napok eseményeinek átbeszélésével. Így az ENN rögzítése és kiértékelés az adott folyamatot nem terhelné extra költségekkel, és ez az előbb felsorolt betegségek esetében is egy gyakran kutatott beszédtípus.

Az x-vektor jellemző kinyerő eljárás elsősorban, a beszéd akusztikai lenyomatát reprezentálja, ugyanakkor a temporális, illetve szemantikai jellemzőket nem, vagy

csak kevéssé veszi figyelembe. Így a továbbiakban tervezzük, hogy ezen jellemzőket is felhasználva újra elvégezzük a vizsgáltot. Az x-vektor módszer másik nagy hátránya, hogy szinte semmit sem mond arról, hogy mi az eltérés a beszédben az egyes osztályok között, így a jövőben tervezzük az akusztikai jellemzők klasszikus vizsgálatát, illetve egy azon alapuló gépi modell tesztelését. A gépi tanuló eljárások statisztikai alapúak így a nagy számú minta elengedhetetlen a robusztus és jól működő modellek létrehozásában, így tervezzük új felvételek rögzítését is. Továbbá tervezzük jelen módszer angol nyelvű beszédmintákon való tesztelését.

Jelen kutatás eredményeinek felhasználásánál érdemes figyelembe venni az adatbázis mennyiségi és minőségi jellemzőit. Továbbá, hogy az egyes gépi modellek paramétereit nem optimalizáltuk, hanem alap értékeket használtunk.

Köszönetnyilvánítás

This work was partly funded by project no. K143075, which has been implemented with the support provided by the National Research, Development, and Innovation Fund of Hungary, financed under the K 22 funding scheme and the CELSA project titled: “Identification of language-independent speech and language biomarkers characteristic of dementia”.

Bibliográfia

Better, M. A.: Alzheimer’s disease facts and figures. *Alzheimers Dement* 19.4, pp.: 1598-1695, (2023)

Breiman, L.: Random forests. *Machine Learning*, 45(1), pp. 5–32, (2001)

Campbell, E. L., Fernández, L. D., Raboso, J. J., & García-Mateo, C.: Alzheimer's Dementia Detection from Audio and Language Modalities in Spontaneous Speech. In IberSPEECH. (2021)

Chang, C.C., Lin, C.J.: LIBSVM: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, *2*(3), pp.1-27. (2011)

Cortes, C., Vapnik, V.: Support vector machine. *Machine learning* 20.3,pp. 273-297, (1995)

Cover T. M., Hart P. E.: Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), pp. 21–27, (1967)

Dwivedi, S., Kasliwal, P. and Soni, S.: Comprehensive study of data analytics tools (RapidMiner, Weka, R tool, Knime). In 2016 Symposium on Colossal Data Analysis and Networking (CDAN) (pp. 1-8). IEEE. (2016)

Egas López, J.V., Tóth, L., Hoffmann, I., Kálmán, J., Pákáski, M. and Gosztolya, G.: Assessing Alzheimer’s disease from speech using the i-vector approach. In *International Conference on Speech and Computer* (pp. 289-298). Cham: Springer International Publishing. (2022)

Egas-López, J.V., Balogh, R., Imre, N., Tóth, L., Vincze, V., Pákáski, M., Kálmán, J., Hoffmann, I. and Gosztolya, G.,: Enyhe kognitív zavar detektálása beszédhangból x-vektor reprezentáció használatával. XVII. Magyar Számítógépes Nyelvészeti Konferencia Szeged, 2021. január 28–29., (2021)

Egas-López, J.V., Kiss, G., Sztahó, D. and Gosztolya, G., 2022, May. Automatic assessment of the degree of clinical depression from speech using x-vectors. In ICASSP 2022-2022 IEEE

International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 8502-8506). IEEE.

Folstein, M., Folstein, S., McHugh, P.: Mini-mental state: a practical method for grading the cognitive state of patients for the clinician. J. Psychiatr. Res. 12(3), pp. 189–198, (1975)

Freedman, M., Leach, L., Kaplan, E., Winocur, G., Shulman, K., Delis, D.: Clock Drawing: A Neuropsychological Analysis. Oxford University Press, New York (1994)

Haulcy, R. M., Glass, J.: Classifying Alzheimer's disease using audio and text-based representations of speech. Frontiers in Psychology, 11, 624137. (2021)

Jenei, A.Z., Kiss, G., Sztahó, D.,: Detection of speech related disorders by pre-trained embedding models extracted biomarkers. In *International Conference on Speech and Computer* (pp. 279-289). Cham: Springer International Publishing. (2022)

Kiss, G. and Vicsi, K.: Mono-and multi-lingual depression prediction based on speech processing. International Journal of Speech Technology, 20(4), pp.919-935, (2017)

Kiss-Vetráb, M., José Vicente, E.L., Balogh, R., Imre, N., Hoffmann, I., Tóth, L., Pákáski, M., Kálmán, J. and Gosztolya, G.,: Enyhe kognitív zavar automatikus felismerése szekvenciális autoenkóder használatával. XVIII. Magyar Számítógépes Nyelvészeti Konferencia Szeged, 2022. január 27–28., (2022)

Low, D. M., Bentley, K. H., Ghosh, S. S.: Automated assessment of psychiatric disorders using speech: A systematic review. laryngoscope investigative otolaryngology, (2020)

Olaszy, G.: A beszéd komplex szerkezete. A magyar beszéd. szerk. Olaszy, G. és Németh, G., Akadémiai kiadó, pp. 9-17, (2010)

Pappagari, R., Cho, J., Joshi, S., Moro-Velázquez, L., Zelasko, P., Villalba, J., & Dehak, N.: Automatic detection and assessment of Alzheimer disease using speech and language technologies in low-resource scenarios. In Interspeech Vol. 2021, pp. 3825-3829. (2021)

Rosen, W., Mohs, R., Davis, K.: A new rating scale for Alzheimer's disease. J. Psychiatry Res. 141(11), pp. 1356–1364, (1984)

Snyder, D., Garcia-Romero, D., Povey, D., Khudanpur, S.: Deep Neural Network embeddings for text-independent speaker verification. In: Intersp ee h. pp. 999-1003. Stockholm, Svédország (2017)

Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., Khudanpur, S.: X-vectors: Robust DNN embeddings for speaker verification. In: ICASSP. pp. 5329-5333. Calgary, Alberta, Kanada (2018)

Subramanian, V., Kwon, N., Brueckner, R., Blaylock, N., & O'Connell, H.: Detecting Mild Cognitive Impairment using Vocal Biomarkers from Spontaneous Speech. In Proc. Workshop on Speech, Music and Mind. Hyderabad, India: ISCA pp. 1-5. (2024)

Tripathi, T., & Kumar, R.: Speech-based detection of multi-class Alzheimer's disease classification using machine learning. International Journal of Data Science and Analytics, 18(1), 83-96. (2024)

Varma, M., Simon, R.: Bias and variance in error estimation with nested cross-validation. *BMC bioinformatics*, 7(1), pp. 1–8. p, (2006)

Vetráb, M. and Gosztolya, G.,: Using hybrid HMM/DNN embedding extractor models in computational paralinguistic tasks. *Sensors*, *23*(11), p.5208, (2023)

Vigo, I., Coelho, L., & Reis, S.: Speech-and language-based classification of Alzheimer's disease: a systematic review. Bioengineering, 9(1), p. 27, (2022)

Vincze, V., Szatlóczki, G., Tóth, L., Gosztolya, G., Pákáski, M., Hoffmann, I., Kálmán, J.: Telltale silence: temporal speech parameters discriminate between prodromal dementia and mild Alzheimer's disease. *Clinical Linguistics & Phonetics*, *35*(8), pp. 727-742. (2021)

Wang, N., Cao, Y., Hao, S., Shao, Z., & Subbalakshmi, K. P.: Modular Multi-Modal Attention Network for Alzheimer's Disease Detection Using Patient Audio and Language Data. In Interspeech Vol. 2021, pp. 3835-3839. (2021)

World Health Organization, Global status report on the public health response to dementia. (2021).

Evaluating and Improving the Intelligibility of Hungarian Dysarthric Speech Using Foundation Models

Árpád Berta[1], Bernadett Dam[2,3], László Tóth[1], Lívia Ivaskó[2]

[1]University of Szeged, Institute of Informatics
[2]University of Szeged, Department of General Linguistics
[3]University of Szeged, Doctoral School of Linguistics
berta@inf.u-szeged.hu

Abstract. The current trend in AI is to base the development of practically any application on so-called foundation models, "a type of AI technology that are trained on vast amounts of data and can be adapted to a wide range of tasks and operations". This paper examines whether this strategy is viable in speech technology when our input is special in at least two senses: first, it is produced by Hungarian speakers, and second, these speakers have dysarthria. Dysarthria is a medical umbrella term that covers neurologic-related speech disorders that affect multiple aspects of speech, resulting in impaired speech intelligibility. Conclusively, dysarthria might quite negatively impact social connections, and thus life quality in general. Our final goal is to improve the intelligibility of the affected people, and of course in their native language, Hungarian. This might lead to further complications, as the applied technologies are mainly developed for English. Due to the difficulties of collecting task-specific data the idea of using general-purpose speech technology tools (that were originally developed for English) arises naturally. The paper discusses the possibilities and pitfalls of this approach, and our main conclusion is that albeit these techniques might basically work, resulting in increased automatic recognition performance, applying them to Hungarian and dysarthric speech requires special attention to avoid any unexpected behavior.
Keywords: dysarthria, intelligibility, foundation model

1 Introduction

In the recent decade, artificial intelligence (AI) has turned from a niche research topic into an ubiquitous development tool. As part of this, the use of 'foundation models' became widespread. According to Bommasani et al. (Bommasani et al., 2021), the expression means "any model that is trained on broad data that can be adapted to a wide range of downstream tasks". From the application developers' point of view, building on them is advantageous because the AI part may be taken for granted. Meanwhile the AI engineer can focus on only the AI issues

- for example, the creation of these models typically requires special hardware and vast amounts of data. In the following, we will use the term 'foundation model' in a broad sense, generally discussing the use of (neural) AI tools that are readily available from the internet without retraining, be of any size.

Unfortunately, language engineers face a further special problem here: big foundation models – for example, large language models – are usually created with the English language in mind, and while they might work for other languages as well, it is not necessarily guaranteed. Even worse, in this paper we try to apply the AI tools originally developed for English not simply to Hungarian speech, but to Hungarian dysarthric speech, which is even more a special case.

Dysarthria is a collective medical term that covers motor speech disorders of neurologic origin. Speakers with dysarthria may exhibit difficulties with any or all components of speech production, e.g., respiration, phonation, resonance formation, and articulation (Duffy, 2013). Any disorder of speech planning and execution – acquired or developmental – poses a serious communication barrier. Phonation and articulation disorders can affect speech production temporarily or permanently, in some cases worsening gradually. The resulting speech – even though it may be well-formed in terms of content and grammar – will be difficult or almost impossible to understand (Horváth and Hirshberg, 2013). Impaired intelligibility can reduce the affected people's ability to live independently. We intend to improve their life quality by exploiting the tools of computational linguistics and artificial intelligence, thus supporting their social reintegration (Tóth et al., 2018; Terbe et al., 2022).

Dysarthria can have multiple causes (Aronson, 1981). Depending on the damaged or dysfunctional areas, individual occurrences can be grouped according to the different characteristics of speech disturbances. Depending on the extent to which speech sounds and suprasegmental features are affected, we can distinguish different types of dysarthria. For example, those whose speech disorder is caused by neurodegenerative diseases or traumatic brain injury will produce speech errors such as monotonic voice or hyperkinetic speech. Speech volume, pitch and rhythm can show a great variation depending on the origin and location of the lesion or disease. Among dysarthria types resulting from motor function impairment, weakened control of the articulatory organs affects the process of producing certain speech sounds, resulting in blurred or distorted phones, like distorted vowels or imprecise consonants. The difficulty of sustaining sounds at a given pitch for a given duration will manifest as a distortion at the suprasegmental level of speech, i.e., prosody. Lastly, dysfunctions of the vocal cord control also deteriorate voice quality, causing a condition known as dysphonia (Markó et al., 2007), which often co-occurs with dysarthria (Camillo and Ortiz, 2007). In summary, dysarthria is a complex clinical condition that may affect several components of speech to varying degrees, with different degrees of severity depending on the underlying clinical condition (e.g., ALS, multiple sclerosis, stroke, traumatic brain injury, cerebellar disorders, Parkinson's disease, and other neurological disorders).

Any neurological disorder, consequently dysarthria is of course not language-specific. However, there might be some special speech properties in which English and Hungarian are basically different, which might hinder the application of an English language technology tool to Hungarian. We think that the two languages are not so basically different regarding articulatory fine movements; that is, the basic sound formation process is the same. The main difference lies in the features that listeners use as keys to differentiate certain speech sounds or larger segments from each other. Thus, from the point of intelligibility, different features may be relevant for speakers of the two languages. We observe the largest differences at the suprasegmental level, in prosody. English is categorized as a stress-timed language (Aubanel and Schwartz, 2020), while Hungarian is definitely not. This may influence both certain timing issues and the reduction of non-stressed syllables (in English), so any tool that offers to manipulate these factors should be used with special care.

2 Approaches to Improving Intelligibility

Our main goal is to support the communication of the affected people by increasing the intelligibility of their speech by computational means. A very general approach might be the application of speech synthesis. The current technology allows very good-quality synthetic speech, along with very quick methods to personalize it (Székely et al., 2025). By "generality" we mean that this approach would work even when the user is not able to speak at all. Meanwhile this feature is a bottleneck at the same time: in practice we observed that for the target group – typically old people with neurological disorders – it is difficult to type the input of a speech synthesizer. If the user is able to speak, automatic speech recognition (ASR) offers a reasonable alternative to typing. However, a general-purpose speech recognizer may not perform well for dysarthric speech, and would require special adjustments (Mihajlik et al., 2025). Moreover, converting the speech to a written form would discard some suprasegmental (e.g. individual and/or prosodic) details. Thus, here we decided to focus on improving dysarthric speech, which implicitly assumes that the subject is able to produce speech (albeit dysarthric). In that case the closest existing technology one might apply is voice conversion (VC). Which is a general speech technology application where the goal is to modify the voice of a (so-called source) speaker so that it became similar to a target speaker, as if by (s)he/he was talking (Mohammadi and Kain, 2017). More generally, it can be considered as a special case of voice transformation, as we want to change certain aspects of the voice (although not necessarily all – see later). Voice conversion is mainly utilized by the entertainment industry (Turk and Arslan, 2002) – for example, it allows the manipulation of the sound track of a movie or a vinyl record retrospectively. Another application area is telecommunication: for example, one might personalize the synthesized voice in a teleconference-application, or even combine it with real-time machine translation. A special medical case is when we seek to reconstruct the original voice of a patient, or at least provide the opportunity to communicate by

synthetic speech – a field known as augmentative and alternative communication (AAC) (Székely et al., 2025).

The concept of voice conversion is not novel (Moulines et al., 1995), but it has became more popular and successful only with the advent of deep learning. The first solutions required a parallel corpus – that is, the same utterances from both the source and the target speaker. Later came the non-parallel methods (Kaneko et al., 2021), and those that are supposed to work for more speakers as well, in a 'many-to-many' or 'many-to-one' fashion (Kaneko et al., 2019).

As you can see, voice conversion is a general technology that was not directly invented for 'repairing' dysarthric speech. Although it might be tried, some aspects need special consideration. The first one is the identity of the target speaker. Voice conversion aims to convert all aspects of a speaker, while in the case of a dysarthric speech we would want to get rid of all dysarthric features while preserving all personal properties. Optimally, we have recordings from the given speaker from before becoming dysarthric, but in general this is not possible (for example, when the dysarthria is not acquired but congenital). In the latter case, it would be good to have a huge 'donor' database of voices, so the patient could select a voice (s)he prefers. Implantation of any user-specific properties (for example, if the patient has pre-morbidity recordings) could then be a second step. In this paper we convert the voice to one healthy speaker from whom we had plenty of samples, and leave the issue of personalization for later.

As we seek to improve the quality and intelligibility of a voice, at first look we would say that the articulatory fine movements of the target speech should be transferred to the source speech, without influencing the voice production component (with a traditional speech technology term, the 'excitation signal'), and this way we could preserve personal components such as the pitch and the personal tone in general. In practice, however, the situation is not this simple, because dysarthria usually affects not only the articulation but the sound production as well. Even worse, usually the prosody is also hurt, which appears as a general slowness in the simplest case, but usually it influences other factors of prosody (e.g. loudness, pitch, timing...) as well, significantly contributing to the unintelligibility of speech. Similarly, a low signal-to-noise ratio may also play a role in decreasing the intelligibility. Consequently, we will also experiment with noise reduction methods, which - together with tempo manipulation - may act as a 'preprocessing' step before voice conversion.

A further issue is the amount of dysarthric samples available. As we will devote the next chapter to the hardships of data collection, we just generally say here that as machine learning requires huge amounts of training data, we will prefer those methods that require the least possible amount of training samples. Machine learning (especially computer vision) introduced the term 'one-shot learning' for basically this sparse-data scenario , which was later extremized by introducing 'zero-shot learning' (Xian et al., 2019). In this paper we focus on such implementations of voice conversion that offer to work in a zero-shot manner, so, practically, do not require a fine-tuning training step (while earlier we experimented with more traditional approaches (Terbe et al., 2022)).

3 Databases - and the difficulties of data collection

An important argument in the favor of foundation models is that they are typically trained on enormous amounts of data – much more than what would be available for us. However, in the case of language technology this normally means English language data, and it is an interesting question in itself whether these models would work for Hungarian. Within language technology, speech processing is a small subfield, even in English. So, Hungarian data is much less available in general. Even worse, here we are targeting dysarthric speech, which is even more sensitive in a data protection sense, as it counts as medical data. Furthermore, speaking can be effortful for the affected people, so we cannot expect large datasets (meaning hundreds or even thousands of hours) from them.

For the current study, we used selected samples from the Hungarian Dysarthria Database[1] (Jenei et al., 2022), which was collected by the Laboratory of Speech Acoustics at the Budapest University of Technology and Economics. This corpus contains speech recordings from 50 adult dysarthric speakers, all of whom are native Hungarian. The materials include read speech: a collection of standalone sentences, as well as a longer passage (approximately 1-minute-long).

For our current purpose we decided to select speakers with different intelligibility levels. The basic demographic data of the used speakers are presented in Table 1. As the group of control people and those with severe dysarthria were underrepresented in the database, the authors of this paper also contributed, providing further voice material for the experiments. We took care to represent both genders in the samples. The etiology of dysarthria was also varied, for example, stroke or Parkinson's disease.

Table 1. Basic information about the speakers (and their voice samples)

Severity	Speakers	Ages	Total Duration
Control	5	36; 37; 42; 52; 66	0.5 hours
Mild dysarthria	7	36; 40; 46; 49; 64; 66; 66	1.9 hours
Moderate dysarthria	4	49; 54; 70; 73	1.6 hours
Severe dysarthria	3	39; 40; 53	1.1 hours

4 What is Intelligibility?

As we already wrote, our main goal is to improve the intelligibility of dysarthric speech, while preserving the personality (e.g. tone) of the original speaker. Unfortunately, both factors are very difficult to formalize and measure.

Here we will focus only on the first one, intelligibility, as we immediately faced a basic question: how do we plan to evaluate our models, that is, can we measure the intelligibility of a speech signal? Can it be expressed as one number?

[1] http://lsa.tmit.bme.hu/databases/dysarthria-hun.html

Subjective evaluation would be an obvious option. That would mean that average human subjects (or even better: experts, e.g. speech therapists) are asked to evaluate the intelligibility of our signals. But for a software developer this sounds troublesome. It would be much simpler if we could automate this step. This is the main reason why in the literature frequently the error rate of an automatic speech recognizer (ASR) is used to estimate the intelligibility[2]. However, this simplification implicitly assumes that artificial intelligence is a good approximation of human intelligence – usually without any direct subjective listening evaluation (with some notable exceptions (Liu et al., 2024; Baas et al., 2023)).

Such an experimental comparison surely does not exist for Hungarian. Thus, in our first experiment, we examined the correlation between speech therapists' evaluation and ASR efficiency. Our data was described in the previous section, while as regards classification, our speakers were grossly classified into the subsets "healthy, mildly, moderately, severily dysarthric", by clinical experts, as the current clinical practice uses these categories (the worst clinical case, "anarthric – not able to speak at all" is missing for obvious reasons) (Al-Ali et al., 2024). While traditional assessments by speech-language pathologists are common, this process is time-consuming, subjective, and can vary between practitioners.

As regards ASR, we needed solutions that work with our Hungarian samples. As a first option, we applied well-known international ASR systems designed for multilingual speech transcription. We report recognition results from OpenAI's Whisper (Large-V3) (Radford et al., 2023) and Meta's Massively Multilingual Speech (MMS) (Pratap et al., 2023) – this latter consists of 1 billion parameters and was trained on 102 languages. Both were trained on Hungarian speech data as well, and we explicitly configured them to provide Hungarian transcriptions.

Apart from these firms, Hungarian researchers also developed solutions for Hungarian speech recognition. From among these, we chose to work with BEAST2 – a recognizer developed for academic purposes (Kádár et al., 2023). Quite recently, even better performance was achieved by the company SpeechTex (Mihajlik et al., 2025), who generously granted us access to their system, which we will refer to as FastConformerHU (or FConfHU) in the following.

It is important to note that none of the ASR models was fine-tuned during any phase of our evaluation. To the best of our knowledge, none of them were trained on dysarthric speech data. This setup simulates real-world scenarios in which an ASR system encounters dysarthric speech for the first time — without any prior exposure or adaptation to such speech patterns.

Table 2 reports the word error rates (WER) obtained , so lower values indicate better intelligibility (for the machine). For each dysarthria severity class, the average WER is the mean of speaker-level average WERs. We also show character error rates (CER), as it offers a more nuanced evaluation for Hungarian than the WER alone. This is due to Hungarian's rich agglutinative morphology: ASR

[2] Speech technology has several methods to evaluate the quality of speech signals, such as the STOI (short-term objective intelligibility), and we did use these earlier (Terbe et al., 2022). However, these metrics were originally developed for speech synthesis or speech enhancement, and their use for the current purpose is at least debatable.

Table 2. ASR error rates (%) for the categories defined by speech-language therapists.

	healthy		**mild**		**moderate**		**severe**	
ASR engine	**WER**	**CER**	**WER**	**CER**	**WER**	**CER**	**WER**	**CER**
Whisper-large-v3	19.3	11.1	70.8	41.5	85.1	48.1	100.8	73.7
MMS-1B-FL102	31.5	8.9	89.4	62.4	94.1	74.0	98.9	78.5
BEAST2	14.3	3.8	62.8	36.6	76.0	49.7	88.3	65.9
FastConformerHU	6.5	2.1	45.2	20.7	55.8	26.4	82.7	51.5

Table 3. ASR error rates (%) after noise reduction.

	healthy		**mild**		**moderate**		**severe**	
ASR engine	**WER**	**CER**	**WER**	**CER**	**WER**	**CER**	**WER**	**CER**
Whisper-large-v3	**19.2**	**10.9**	72.9	43.9	**81.7**	**45.9**	**98.6**	**66.5**
MMS-1B-FL102	33.8	9.0	90.3	63.5	94.2	74.5	**98.8**	**78.2**
BEAST2	15.1	4.1	63.8	37.1	77.3	51.0	89.1	66.2
FastConformerHU	6.8	2.2	45.9	21.3	57.7	27.8	82.9	52.7

models often correctly identify the stem but miss the actual word form. In such cases, WER penalizes the entire word as incorrect, even though most characters are right — whereas CER accounts for the partial correctness.

Generally, ASR error rates indeed correlate with the speech therapists' rating, suggesting that ASR may serve as a gross approximation of the voice intelligibility. Moreover, the Hungarian-specific models consistently outperformed the multilingual ones. In all subsequent tables, WER/CER values lower than those for the original input will be shown in bold, indicating performance improvement.

5 Noise Reduction

Our sound material - collected in real environments - was contaminated by a high amount of background noise. Moreover, individuals with dysarthria would not be able to properly tune their loudness when using some voice processing tool, for example, a cellphone, which may result in too soft recordings. These observations suggest that loudness normalization and noise reduction could improve voice quality, and thus intelligibility. However, these solutions should be used with a special caution, as these algorithms are often built on spectral subtraction, which may introduce artifacts. As most ASR systems are not trained on such a data, noise reduction may be detrimental on their performance (Gerazov et al., 2026).

In our experiments, we used Sepformer (Subakan et al., 2021) for speech enhancement. The model was pre-trained on the Microsoft DNS-4 dataset (Dubey et al., 2022), which includes 150 diverse noise types (e.g., baby crying, heavy breathing, fan, mouse click). While originally it was designed for speaker separation, Sepformer proved effective for speech enhancement as well. We treated it as a foundation model and do not fine-tune it during evaluation.

Table 3 shows the impact of SepFormer on "speech intelligibility" (ASR scores). We found improvements for nearly all severity levels, according to Whisper and MMS, which was notably significant for Whisper in the "severe" case.

Table 4. ASR error rates (%) after applying rhythm conversion (RC).

RC method	Whisper WER	Whisper CER	MMS WER	MMS CER	BEAST2 WER	BEAST2 CER	FConfHU WER	FConfHU CER
healthy								
Urhythmic(Global)	25.1	14.6	35.7	10.4	20.2	6.1	8.4	2.6
Urhythmic(Fine)	35.6	24.1	41.1	12.5	23.9	8.1	11.1	3.4
Syllable(Global)	37.1	27.1	37.0	10.8	21.3	6.4	8.0	2.5
Syllable(Fine)	29.3	17.2	40.9	14.7	23.3	8.3	12.8	4.2
mild dysarthria								
Urhythmic(Global)	78.7	48.9	92.4	69.6	68.1	43.1	52.1	26.9
Urhythmic(Fine)	84.6	55.8	95.0	75.4	74.9	49.2	61.2	33.5
Syllable(Global)	81.5	49.9	93.7	69.5	69.5	43.7	49.1	23.9
Syllable(Fine)	105.0	72.6	95.6	75.6	76.4	51.2	64.0	34.8
moderate dysarthria								
Urhythmic(Global)	92.2	57.2	95.0	78.0	79.1	53.7	60.5	31.2
Urhythmic(Fine)	94.5	64.0	96.2	79.5	82.4	58.0	68.2	38.5
Syllable(Global)	87.9	54.2	95.1	76.6	78.3	52.8	57.3	28.6
Syllable(Fine)	111.4	73.4	95.8	79.6	82.2	56.8	65.9	35.8
severe dysarthria								
Urhythmic(Global)	104.2	76.2	**98.6**	**77.6**	**87.8**	66.1	**81.7**	50.6
Urhythmic(Fine)	120.5	92.3	**98.6**	78.7	88.8	67.8	84.3	53.7
Syllable(Global)	**97.2**	**72.6**	**98.8**	**76.4**	**87.0**	66.2	**81.2**	50.5
Syllable(Fine)	148.4	113.4	**98.5**	**77.4**	88.4	67.5	82.7	51.9

6 Rhythm Reconstruction

Although dysarthria is highly heterogeneous, slow articulation is a common trait, suggesting that rhythm normalization would enhance intelligibility without any further modification. To verify this, we evaluated two unsupervised rhythm conversion methods: Urhythmic (van Niekerk et al., 2023) and a syllable-based method (El Hajal et al., 2025b). Both first segment speech using acoustic features extracted by a foundation encoder, then model the rhythm distribution of these segments, and finally align the source rhythm to that of a target speaker. Urhythmic uses hierarchical clustering to divide the signal into segments based on sonority. While it was originally developed for general-purpose rhythm conversion, it was also tried in the recognition of dysarthric speech (El Hajal et al., 2025a). The syllable-based approach focuses on syllable nuclei, estimating peak-to-peak syllable durations per speaker. Both variants model speaking rate at two levels: fine-grained and global. In the fine-grained scenario a gamma distribution is fitted on speech segment durations to learn a speaker-specific duration model. In global rhythm modeling, a speaking rate — such as syllables per second or sonorants per second - is approximated over all speech segments. The rhythm model of a dysarthric source speaker is mapped onto that of a healthy target speaker, transmitting the rhythmic properties of the target to the source.

Table 5. ASR error rates (%) after applying voice conversion.

VC method	Whisper WER	Whisper CER	MMS WER	MMS CER	BEAST2 WER	BEAST2 CER	FConfHU WER	FConfHU CER
	healthy							
Seed-VC	23.3	11.0	41.6	13.4	21.3	6.6	10.6	3.4
kNN-VC	30.3	18.5	36.7	10.8	18.7	5.8	8.4	2.7
Urhythmic(Global)+kNN-VC	29.6	17.0	36.7	10.9	18.6	5.5	8.6	2.8
Urhythmic(Fine)+kNN-VC	34.2	21.9	40.8	12.6	23.8	8.4	11.6	3.7
Syllable(Global)+kNN-VC	34.2	22.3	36.8	10.6	20.2	6.1	8.7	2.7
Syllable(Fine)+kNN-VC	33.7	20.3	41.7	15.1	24.3	8.5	12.6	4.4
	mild dysarthria							
Seed-VC	81.7	51.1	95.7	71.9	73.2	44.8	63.2	33.6
kNN-VC	75.9	45.3	94.4	69.8	70.9	42.4	56.3	28.4
Urhythmic(Global)+kNN-VC	77.0	46.8	92.0	68.3	69.4	42.3	57.0	30.2
Urhythmic(Fine)+kNN-VC	80.9	51.8	94.7	73.9	74.6	47.2	63.4	35.2
Syllable(Global)+kNN-VC	78.0	47.6	92.8	68.2	71.0	43.2	56.4	28.7
Syllable(Fine)+kNN-VC	94.8	58.9	95.3	74.7	76.4	49.6	67.3	37.3
	moderate dysarthria							
Seed-VC	94.4	60.3	97.4	76.5	80.9	51.8	75.2	40.1
kNN-VC	94.3	57.5	97.3	76.6	82.6	54.8	71.1	38.5
Urhythmic(Global)+kNN-VC	98.6	63.6	96.9	76.6	79.3	51.6	70.7	38.3
Urhythmic(Fine)+kNN-VC	109.9	73.9	97.0	78.1	82.9	55.3	75.1	42.0
Syllable(Global)+kNN-VC	104.2	62.8	95.4	74.0	79.2	51.7	70.2	36.9
Syllable(Fine)+kNN-VC	123.4	86.0	96.4	79.0	82.1	55.1	75.2	41.5
	severe dysarthria							
Seed-VC	116.3	88.1	99.1	79.2	91.9	67.4	88.0	56.7
kNN-VC	113.1	85.8	100.0	**78.2**	91.4	69.3	88.5	58.2
Urhythmic(Global)+kNN-VC	113.9	80.9	99.8	79.6	90.0	66.5	86.8	55.0
Urhythmic(Fine)+kNN-VC	112.5	83.6	99.5	79.7	90.0	67.3	88.1	56.5
Syllable(Global)+kNN-VC	115.8	88.7	99.4	**78.4**	88.5	66.6	87.2	55.2
Syllable(Fine)+kNN-VC	159.2	115.1	99.0	**78.2**	89.0	67.6	87.9	56.5

The Szindbád Hungarian single-speaker audio book (Tóth, 2009) was used in our evaluations both to re-train the speech segmenter and to define the target rhythm model — leveraging the healthy prosody of the narrator as reference. Speech volume was normalized to -20dB, following the preprocessing procedure suggested by the inventors[3]. Acoustic features were extracted using a pre-trained WavLM Large encoder (Chen et al., 2022), and waveforms were reconstructed with a HiFi-GAN vocoder specifically adjusted to WavLM features, as proposed by the authors of the method (Baas et al., 2023).

The WER and CER scores for the examined rhythm conversion methods are summarized in Table 4. The syllable-based (global) method reduced error rates for severe dysarthria for all evaluated ASR models. Similarly, global Urhythmic reduces the error rates on more than one ASR model. For severe dysarthric speech, both methods proved effective in enhancing intelligibility.

[3] https://github.com/idiap/RnV

7 Experiences with Voice Conversion

As the amount of available data was very limited – and keeping in mind a possible future application – we preferred those approaches that required no or minimal training data (in an earlier work we tried more traditional approaches (Terbe et al., 2022)). So we focused on those voice conversion methods that promised to work in a 'zero-shot' manner, or at least, with minimal retraining. In the following we introduce the two methods that we experimented with.

7.1 K-Nearest Neighbor Voice Conversion

The k-nearest neighbor voice conversion (kNN-VC) method (Baas et al., 2023) transforms source-speaker acoustic features into target-speaker features by finding the k nearest neighbors of each source frame within the target speaker's feature space. Acoustic features are extracted using a pre-trained WavLM Large encoder (Chen et al., 2022), which was trained in a self-supervised manner on large-scale English speech data. Feature similarity is measured via cosine similarity. The encoder remains frozen (so it requires no fine-tuning) during use, k-NN is non-parametric, so the whole approach is entirely zero-shot.

In their follow-up studies (El Hajal et al., 2025a,b) the inventors demonstrated that combining kNN-VC with rhythm conversion can improve the intelligibility of dysarthric speech. In this setup, the source was a dysarthric speaker, while the target was a healthy speaker, and rhythm conversion was followed by kNN-based feature matching. Both methods operate using the same acoustic feature representation, with waveform synthesis applied only once, at the end.

As described in Section 6, the pre-matched vocoder was employed for waveform synthesis, using the Szindbád audio book's reader as the target speaker with volume normalization. Table 5 shows the results both for kNN-VC and kNN-VC combined with rhythm conversion. A slight improvement was observed in the severe dysarthria class by the MMS ASR. Otherwise, however, the error rates increased for all other categories.

7.2 Zero-Shot Voice Conversion with Diffusion Transformers

The next examined zero-shot VC method is called Seed-VC (Liu, 2024). In transforms the speech of an unseen source speaker so as to match the timbre of a reference speaker, while preserving linguistic content. According to the authors, "it enhances timbre representation by employing a diffusion transformer that leverages the full context of the reference utterance, enabling a precise capture of fine-grained speaker characteristics". Furthermore, "an external timbre shifter is applied during training to deliberately perturb the source speech's timbre, forcing the content extractor to learn robust, timbre-invariant representations of linguistic information and thereby minimizing timbre leakage at inference time".

Here, we apply the pretrained Seed-VC model[4] to transfer the intended speech information while converting the acoustic characteristics to match those

[4] https://github.com/Plachtaa/seed-vc

of the target speaker. Here, we again used the Szindbád audiobook corpus as samples for the target speaker. The pretrained Seed-VC model is built upon a U-ViT diffusion transformer architecture (Bao et al., 2023). The authors applied Whisper-Small (Radford et al., 2023) as the semantic encoder to extract linguistic features from input speech, which were used as input to the U-ViT during training. The training was performed on the Emilia-101k dataset (He et al., 2024), which contains 101k hours of spontaneous speech data across six languages — though Hungarian was not among these. This framework also employs a pretrained BigVGANV2 (Lee et al., 2023) vocoder for waveform synthesis.

Table 5 summarizes the results with voice conversion (sometimes in combination with rhythm adjustment). Seed-VC performed competitively with kNN-VC, a method specifically proposed for dysarthric speech conversion. However, the error rates overall remained higher than those for the original speech.

8 Discussion

We conclude that although the examined general-purpose speech processing tools can be applied to dysarthric speech, any such development should take care of the specialties of these signals. Even the fact alone that the input is Hungarian introduces peculiarities that a system developed for English might not handle well. Among others, such an observation was that CER provides critical insights beyond WER in the case of Hungarian. For example, we saw cases where the modified speech showed an improved average WER, but this was not accompanied by a corresponding decrease in average CER. The improvement in WER indicates that some previously mis-recognized words was finally hit — necessarily implying a reduction in character errors for those specific words as well. However, for some other words the character-level error count might have increased, leading to an overall stagnation or even degradation of the average CER.

As regards dysarthric speech, we were surprised to see how quickly the ASR performance degraded with the severity of dysarthria. Upon a closer inspection, we found that the high WER is mostly caused by insertion errors – due to the slowness of dysarthric speech that general-purpose ASR systems may not handle well, as simply they were not trained with such a data. A similar problem is the case of stuttering pronunciation, or when the speaker performs sound substitution – putting an existing sound in place of another that (s)he cannot produce. The application of ASR tools presumes that the transcript reflects *what should have been said* instead of what was actually said. This is a special design question that needs the contribution of a speech-language therapist.

Ideally, voice conversion of dysarthric speech should not only enhance clarity but also reduce speech duration, compensating for the common slow articulation. While the best-performing syllable-based (global) rhythm converter reduced the overall duration of severe dysarthric speech by 28%, kNN-VC reduced it by only 6%, and Seed-VC by just 3%. Thus, these two VC methods offered negligible benefits in terms of tempo reconstruction. That is, while both methods do alter the speaker's vocal quality, they barely modify their prosodic properties like

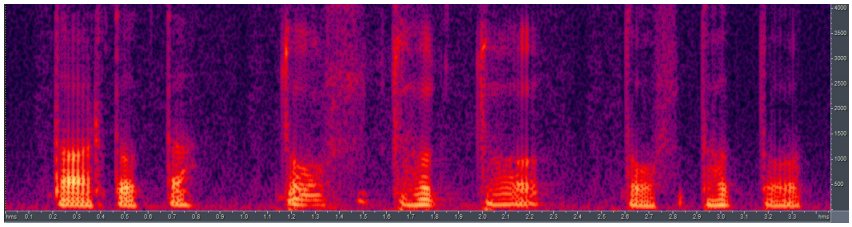


Fig. 1: Spectrograms for the word ‘tartotta’. The pronunciation variants: a) healthy b) dysarthric c) the dysarthric recording processed by Urhythmic.

timing and rhythm. This behavior perhaps can be expected for kNN-VC, as it operates as a local, frame-level algorithm with no explicit modeling of global temporal structure. However, Seed-VC, which is based on a transformer architecture and thus theoretically capable of capturing long-range dependencies, also failed to produce reasonable tempo adjustments. A direct rhythm transformation should be robust; however, its fine-grained variant failed to outperform the simpler global approach. This suggests that the underlying segmentation process fails — particularly for Hungarian. Methods such as peak-to-peak syllable detection or sonorant-obstruent classification, which were developed for English, may not transfer well to Hungarian.

We argue that effective rhythm normalization would be essential -— not only to mitigate insertion errors in ASR, but also to enhance the intelligibility of dysarthric speech. While the evaluated VC method variants are zero-shot, they both incorporate modules that may be language-specific in their design. As a demonstration of language-specific behavior, let us examine an example that combines kNN-VC with the global syllable-based rhythm transformation.

Fig. 1 shows spectrograms of the word ’tartotta’ ([tɑrtotːɑ]), from a speaker from whom we fortunately had pre-illness recordings as well. The image first shows the healthy recording from the speaker, while the second one is the same word with a dysarthric pronounciation. This recording is clearly much longer, but let us focus on the [t] and [tː] pairs. While originally the duration of the latter was more than twice of the former, after dysarthria this contrast became much smaller. The third spectrogram shows the output after applying kNN-VC with global rhythm transformation. While the speech became much ‘faster’, the word is still longer than in the first case. Even worse, the contrast between the short and long variants of [t] did not improve at all. We think that this is a great example of such special properties of Hungarian that would require dedicated attention, as a tool developed for English would never handle such an issue without adjustment. Hence, in future work, we intend to further refine these approaches by adapting them specifically to Hungarian (dysarthric) speech.

Acknowledgments

We thank to the Technical University of Budapest for their dysarthric database and to the SpeechTex Ltd. for providing access to their recognition system.

Bibliography

Al-Ali, A., Al-Maadeed, S., Saleh, M., Naidu, R.C., Alex, Z.C., Ramachandran, P., et al.: Classification of dysarthria based on the levels of severity: A systematic review. IEEE Access 12, 48223–48238 (2024)

Aronson, A.: Motor speech signs of neurologic disease. In: Darby, J.K. (ed.) Speech Evaluation in Medicine, pp. 159–180. Grune and Stratton (1981)

Aubanel, V., Schwartz, J.L.: The role of isochrony in speech perception in noise. Scientific Reports 10(1), 2045–2322 (2020)

Baas, M., van Niekerk, B., Kamper, H.: Voice conversion with just nearest neighbors. In: Proc. Interspeech 2023. pp. 2053–2057 (2023)

Bao, F., Nie, S., Xue, K., Cao, Y., Li, C., Su, H., Zhu, J.: All are worth words: A ViT backbone for diffusion models. In: Proc. CVPR. pp. 22669–22679 (2023)

Bommasani, R., Hudson, D.A., Adeli, E., Altman, R.B., Arora, S., von Arx, S., et al.: On the opportunities and risks of foundation models. CoRR abs/2108.07258 (2021), https://arxiv.org/abs/2108.07258

Camillo, L., Ortiz, K.: Vocal analysis (auditory-perceptual and acoustic) in dysarthrias. Pro-Fono Revista de Atualizacao Cientifica 19(4), 381–386 (2007)

Chen, S., Wang, C., Chen, Z., Wu, Y., Liu, S., Chen, Z., et al.: WavLM: Large-scale self-supervised pre-training for full stack speech processing. IEEE Journal of Selected Topics in Signal Processing 16(6), 1505–1518 (2022)

Dubey, H., Gopal, V., Cutler, R., Matusevych, S., Braun, S., Eskimez, et al.: Deep noise suppression challenge. In: Proc. ICASSP. pp. 9271–9275 (2022)

Duffy, J.R.: Motor speech disorders: third edition. Elsevier (2013)

El Hajal, K., Hermann, E., Hovsepyan, S., Magimai-Doss, M.: Unsupervised rhythm and voice conversion of dysarthric to healthy speech for ASR. In: Proc. Workshop on Speech Pathology Analysis and DEtection (SPADE) (2025a)

El Hajal, K., Hermann, E., Hovsepyan, S., Magimai-Doss, M.: Unsupervised rhythm and voice conversion to improve ASR on dysarthric speech. In: Proc. Interspeech 2025. pp. 2760–2764 (2025b)

Gerazov, B., Politi, M., Bratières, S.: Arabic ASR on the SADA large-scale arabic speech corpus with transformer-based models. In: Proc. SPECOM 2025. pp. 70–84 (2026)

He, H., Shang, Z., Wang, C., Li, X., Gu, Y., Hua, H., et al.: Emilia: An extensive, multilingual, and diverse speech dataset for large-scale speech generation. In: Proc. SLT (2024)

Horváth, S., Hirshberg, J.: Diszartria/diszartrofónia. In: Foniátria és társtudományok II. (in Hungarian), pp. 80–86. Eötvös Kiadó (2013)

Jenei, Z.A., Kiss, G., Sztachó, D.: Detection of speech related disorders by pretrained embedding models extracted biomarkers. In: Proc. SPECOM 2022. pp. 279–289 (2022)

Kádár, S.M., Dobsinszki, G., Mády, K., Mihajlik, P.: „Feeding the BEAST" – extensions to the BEA speech transcriber, including its integration with a neural language model (in Hungarian). In: Proc. MSZNY. pp. 135–143 (2023)

Kaneko, T., Kameoka, H., Tanaka, K., Hojo, N.: StarGAN-VC2: Rethinking conditional methods for StarGAN-based voice conversion. arXiv preprint arXiv:1907.12279 (2019)

Kaneko, T., Kameoka, H., Tanaka, K., Hojo, N.: MaskCycleGAN-VC: Learning non-parallel voice conversion with filling in frames. In: Proc. ICASSP. pp. 5919–5923 (2021)

Lee, S., Ping, W., Ginsburg, B., Catanzaro, B., Yoon, S.: BigVGAN: A universal neural vocoder with large-scale training (2023)

Liu, D., Lin, Y., Bu, H., Li, M.: Two-stage and self-supervised voice conversion for zero-shot dysarthric speech reconstruction. In: Proc. International Conference on Asian Language Processing (IALP) 2024. pp. 423–427 (2024)

Liu, S.: Zero-shot voice conversion with diffusion transformers. arXiv preprint arXiv:2411.09943 (2024)

Markó, A., Gráczi, T., Bajnócziné, S.K.: The efficiency of dysphonia therapy as a function of the patient's former vocal performation experience (in Hungarian). Alkalmazott nyelvtudomány 12(1–2), 83–103 (2007)

Mihajlik, P., Székely, É., Barta, P., Kádár, S.M., Dobsinszki, G., Tóth, L.: Improved dysarthric speech to text conversion via TTS personalization. In: Proc. EUSIPCO. pp. 521–525 (2025)

Mohammadi, S., Kain, A.: An overview of voice conversion systems. Speech Communication 88, 65–82 (2017)

Moulines, E., Sagisaka, Y., (eds.): Voice conversion: state of the art and perspectives. Speech Communication (special issue) 16(2) (1995)

van Niekerk, B., Carbonneau, M.A., Kamper, H.: Rhythm modeling for voice conversion. IEEE Signal Processing Letters 30, 1297–1301 (2023)

Pratap, V., Tjandra, A., Shi, B., Tomasello, P., Babu, A., Kundu, S., et al.: Scaling speech technology to 1,000+ languages. arXiv:2305.13516 (2023)

Radford, A., Kim, J.W., Xu, T., Brockman, G., Mcleavey, C., Sutskever, I.: Robust speech recognition via large-scale weak supervision. In: Proc. ICML. pp. 28492–28518 (2023)

Subakan, C., Ravanelli, M., Cornell, S., Bronzi, M., Zhong, J.: Attention is all you need in speech separation. In: Proc. ICASSP 2021 (2021)

Székely, É., Mihajlik, P., Kádár, S.M., Tóth, L.: Voice reconstruction through large-scale TTS models: Comparing zero-shot and fine-tuning approaches to personalize TTS in assistive communication. In: Proc. Interspeech. pp. 2735–2739 (2025)

Terbe, D., Tóth, L., Ivaskó, L.: Applying voice conversion to improve the voice quality of dysarthric patients (in Hungarian). In: Proc. of MSZNY 2022. pp. 161–174 (2022)

Turk, O., Arslan, L.: Subband based voice conversion. In: Proc. ICSLP (2002)

Tóth, L.: Speech recognition experiments with audio books (in Hungarian). In: Proc. MSZNY 2009. pp. 206–216 (2009)

Tóth, L., Kovács, G., Ivaskó, L., Tóth, A., Jakab, K., Vécsei, L.: Automatic analysis of the speech of post-stroke dysarthric patients - initial results (in Hungarian). In: Orvosi Informatika. Neumann Kollokvium. pp. 43–49 (2018)

Xian, Y., Lampert, C.H., Schiele, B., Akata, Z.: Zero-shot learning — a comprehensive evaluation of the good, the bad and the ugly. IEEE Trans. PAMI 41(09), 2251–2265 (2019)

Face mask Detection from Speech Using Deep Speaker Embeddings

Mohammed Hamzah Alsalihi[1] and David Sztaho[1⋆]

Department of Telecommunications and Artificial Intelligence, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, 1111 Budapest, Hungary
m.abed@edu.bme.hu, sztaho.david@vik.bme.hu

Abstract. This paper investigates the feasibility of detecting face mask usage and coverage area from speech signals using deep speaker embeddings. We evaluate the performance of three pre-trained speaker embedding models: X-vector, ECAPA-TDNN, and ResNet-TDNN on the MASCFLICHT corpus, a publicly available dataset designed for automatic mask classification tasks. The classification is carried out using three common back-end classifiers: logistic regression, multilayer perceptron, and linear support vector machines. Experiments are conducted across three tasks: face mask type classification, mask coverage area classification, and a combined task involving both. Results show that X-vector consistently outperforms other embeddings, achieving the highest unweighted average recall (UAR) in most settings. In contrast, ECAPA-TDNN exhibits lower performance across tasks. While prototypical encoders with feature transformations achieve strong results in isolated tasks, they generalize poorly to combined conditions. These findings indicate that speaker embeddings, particularly X-vector, can effectively capture mask-induced spectral variations in speech and serve as robust features for mask detection tasks.
keywords: face mask , deep speaker embedding , machine learning, classification

1 Introduction

Artificial intelligence (AI) tools, especially deep learning models, have achieved significant progress in speech-related tasks such as automatic speech recognition(Yu et al. 2016; Mihajlik et al. 2024), speaker verification (Abed et al. 2024), and emotion recognition (Chowdhury et al. 2025; Abed et al. 2023). These systems can capture subtle acoustic variations that are difficult for human listeners to detect, enabling applications such as identifying respiratory conditions, detecting stress, or estimating recording environments from speech alone.

One challenging application is detecting whether a speaker is wearing a face mask and, if so, identifying the mask type (Mallol-Ragolta, Spiesberger, et al. 2024). Face masks introduce acoustic changes by attenuating higher frequencies,

⋆ Corresponding author

altering formant structures, and reducing speech amplitude (Shekaraiah et al. 2024). However, these changes are often subtle and vary across speakers, speaking styles, and environments, making automatic detection difficult (Fiorella et al. 2023). The problem becomes more complex when distinguishing between mask types such as surgical and filtering face-piece (FFP2) masks, which differ in material, fit, and acoustic impact. Even advanced deep learning models can struggle to consistently discriminate these variations due to their small magnitude and overlap with natural speech variability.

In forensic voice comparison, this challenge has practical implications. The reliability of speaker classification depends on matching the conditions between questioned and known samples. A mismatch caused by mask usage can degrade system accuracy and influence evidential weight. Detecting the presence and type of mask before performing speaker comparison allows for condition matching, compensation, or case flagging, reducing the risk of misclassification.

Several studies have investigated face mask classification using facial images and acoustic signals, applying different artificial intelligence (AI) techniques (Habib et al. 2022; Su et al. 2022). During the COVID-19 pandemic, researchers also examined the effect of face masks on speaker verification, with a focus on forensic voice comparison. Mallol-Ragolta et al. (Mallol-Ragolta, Spiesberger, et al. 2024) studied face mask type and coverage area recognition from speech using the MASCFLICHT corpus, combining formant-related features, spectrograms, and enriched spectrograms with overlaid formant traces in prototypical networks. Similarly, in (Markitantov et al. 2022), the authors presented the BRAVE-MASKS corpus, a multimodal database of Russian speakers, and developed audio, visual, and audio-visual systems for mask type recognition using deep neural networks and transfer learning.

This topic has gained strong attention in forensic voice comparison, since protective masks act as voice filters and may compromise the reliability of forensic analyses. In (Saraiva et al. 2024) the authors analyzed the effect of surgical and FFP2 masks on acoustic-phonetic parameters across several languages, showing that masks can alter fundamental frequency, intensity, jitter, shimmer, and harmonics-to-noise ratio depending on mask type, language, and sex. Fejes et al. (Fejes et al. 2024) extended this work to forensic automatic speaker recognition systems using the Forensic Multilingual Voices Database (FMVD), reporting performance degradation with both surgical and FFP2 masks, with variation across languages and genders. Earlier studies also contributed to this line of research: Fiorella et al. (Fiorella et al. 2023) found that wearing surgical masks generally did not significantly change acoustic parameters but could reduce vocal intensity in some speakers. Taken together, these studies highlight that the presence of face masks is a critical factor for forensic voice comparison, motivating the development of both acoustic-phonetic and automatic approaches that can reliably handle such conditions.

This study investigates face mask classification from speech, focusing on both mask type and coverage area. We utilise the Mask Augsburg Speech Corpus using FFP2 and surgical masks with the Airways Covered Halfway and Com-

pletely (MASCFLICHT) (Mallol-Ragolta, Urbach, et al. 2023). These conditions enable the joint classification of mask presence, type, and coverage area. We employ three deep speaker embedding approaches: X-vector (Snyder, Garcia-Romero, Sell, et al. 2018; Snyder, Garcia-Romero, Povey, et al. 2017), ECAPA-TDNN (Desplanques et al. 2020), and ResNet-TDNN (Villalba et al. 2020) combined with machine learning classifiers, including multilayer perceptron (MLP), linear support vector classifier (SVC), and logistic regression (LR). The objective is to evaluate whether speech-based systems can reliably detect mask presence, type, and coverage area despite subtle acoustic differences, and to assess the potential of such detection as a pre-processing step in forensic voice comparison and other speech-processing applications.

The paper is organized as follows, Section 2 describes the methodology, including details on the dataset and applied machine learning models. Section 3 reports the performance outcomes across models and evaluation measures. Section 4 offers a detailed discussion of the results and their significance. Section 5 closes the paper with concluding remarks and possible directions for future research.

2 Materials and Methods

The proposed methodology follows the workflow illustrated in Figure 1. Speech signals from the MASCFLICHT dataset are initially processed using pre-trained speaker embedding models to extract distinct feature vectors. Three embedding architectures are evaluated: X-vector, ECAPA-TDNN, and ResNet-TDNN. These features are then fed into backend classifiers, including logistic regression (LR), multilayer perceptron (MLP), and linear support vector classifier (SVC). The classifiers are trained for three classification tasks: face mask types classification, detecting coverage areas, and a combined task that integrates both type and coverage face mask classification.

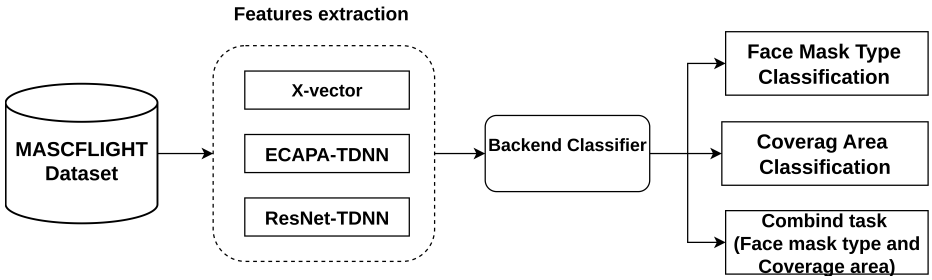


Fig. 1: Face mask classification workflow in the current study

2.1 Dataset

This study utilises the Mask Augsburg Speech Corpus using FFP2 and surgical masks with the Airways Covered Halfway and completely (MASCFLICHT) (Mallol-Ragolta, Urbach, et al. 2023), a publicly released dataset developed for automatic face mask type and coverage area recognition from speech. The corpus comprises 2 hours, 27 minutes, and 55 seconds of audio recorded from 30 native German speakers (15 female, 15 male), aged between 19 and 55. Recordings were conducted in a quiet environment using the built in microphone of a Xiaomi Mi 10 smartphone, with consistent speaker to microphone distance. Speech was collected under five mask-related conditions:

- **NM (No Mask)**: No face covering
- **SP (Surgical Partial)**: Surgical mask covering only the mouth
- **ST (Surgical Total)**: Surgical mask covering mouth and nose
- **FP (FFP2 Partial)**: FFP2 mask covering only the mouth
- **FT (FFP2 Total)**: FFP2 mask covering mouth and nose

Participants completed two types of speech tasks under all conditions: a free speech task involving picture description and a guided speech task consisting of reading phonetically balanced German sentences from the PD1 corpus. Five distinct picture sets and five sentence lists were rotated to ensure variability across participants and conditions. Preprocessing and data partitioning procedures were conducted by the dataset authors as described in (Mallol-Ragolta, Urbach, et al. 2023). The corpus is provided with a speaker-independent split into training, development, and test sets, balanced for gender and classification difficulty. The final partitioning includes: 18 speakers (9 female, 9 male) in the training set, 6 speakers (3 female, 3 male) in the development set, and 6 speakers (3 female, 3 male) in the test set.
The resulting dataset contains a total of 8,875 one-second speech segments, approximately evenly distributed across the five mask conditions. Figure 2 shows the Complete distribution of female and male 1-second audio samples in the MASCFLICHT corpus across training, development, and test partitions for each recording condition.

2.2 Deep Speaker Embedding Models

This study utilizes three advanced deep learning models to derive speaker embeddings from speech data: X-vector, ECAPA-TDNN, and ResNet-TDNN.

X-vector The X-vector model[1] is based on a Time Delay Neural Network (TDNN), originally introduced by Snyder et al. (Snyder, Garcia-Romero, Povey, et al. 2017), and designed to produce fixed-size speaker embeddings from variable-length audio. It includes five frame-level TDNN layers, where the first three layers handle short temporal contexts, and the last two capture broader segment-level

[1] `https://huggingface.co/speechbrain/spkrec-xvect-voxceleb`

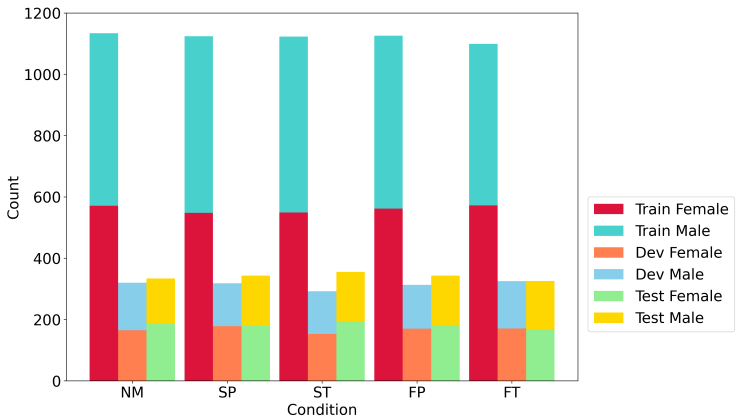


Fig. 2: Complete distribution of female and male 1-second audio samples in the MASCFLICHT corpus across training, development, and test partitions for each recording condition. Conditions include: NM (No Mask), SP (Surgical Partial – mouth only), ST (Surgical Total – mouth and nose), FP (FFP2 Partial – mouth only), and FT (FFP2 Total – mouth and nose).

information. Following these, a statistics pooling layer computes the mean and standard deviation of frame-level outputs, forming a fixed-length vector. This is followed by two fully connected layers, each with 512 units. The resulting 512-dimensional vector serves as the speaker embedding. The model consists of approximately 4.2 million parameters. We used a pre-trained version trained on VoxCeleb1 and VoxCeleb2, as described in (Snyder, Garcia-Romero, Sell, et al. 2018), and available via the SpeechBrain toolkit (Ravanelli et al. 2021). The model takes 24 mel-frequency band energies as input and outputs robust speaker embeddings suitable for classification tasks.

ECAPA-TDNN The ECAPA-TDNN model[2] (Desplanques et al. 2020) builds upon the X-vector TDNN structure with enhancements aimed at boosting embedding quality. These include channel-aware attention mechanisms, SE-Res2Blocks for modeling inter-channel relationships, and multi-layer feature aggregation. These additions allow the model to better capture both local and global information while improving resilience to noise and input variability. We employed a pre-trained version with 22.3 million parameters, available on Hugging Face. The model uses 80 mel-frequency bands as input and produces 192-dimensional embeddings.

ResNet-TDNN The ResNet-TDNN model[3] (Villalba et al. 2020) integrates Residual Network (ResNet) layers with TDNN components. This combination

[2] `https://huggingface.co/speechbrain/spkrec-ecapa-voxceleb`
[3] `https://huggingface.co/speechbrain/spkrec-resnet-voxceleb`

helps capture both short-term and long-term speech features. The model processes 80-band mel-spectrograms and outputs 256-dimensional speaker embeddings. The version used in our experiments was pre-trained on VoxCeleb1 and VoxCeleb2 and made available by the SpeechBrain project. It provides an effective balance between complexity and performance for speaker verification.

2.3 Machine learning models

To evaluate the effectiveness of speaker embeddings for face mask classification, three supervised machine learning models were employed: Logistic Regression (LR), Multi-Layer Perceptron (MLP), and Support Vector Classifier (SVC). All models were trained on embeddings extracted from the speech data, with the task of predicting face mask coverage area, face mask type, and the joint classification of coverage area and face mask type. Prior to training, the embeddings were normalized using Min–Max scaling to rescale each feature into the $[0, 1]$ interval, ensuring stable optimization across classifiers. The Logistic Regression model was configured in the multinomial setting to handle multi-class classification. The `lbfgs` solver was selected due to its efficiency with medium-sized datasets and good convergence properties, while the maximum number of iterations was set to 1000 to guarantee stability. A fixed random seed (42) was applied for reproducibility. This model serves as a linear baseline for performance comparison. To capture non-linear relationships in the embeddings, a Multi-Layer Perceptron (MLP) was implemented using two hidden layers with 256 and 128 neurons, respectively, each activated by the Rectified Linear Unit (ReLU) function. Training was performed with the Adam optimizer for up to 1000 iterations, balancing efficiency and generalization while avoiding overfitting. A Support Vector Classifier (SVC) with a linear kernel was also applied. The linear kernel was chosen due to the high-dimensional nature of x-vector embeddings, where linear separation often provides competitive results with lower computational cost compared to non-linear kernels. Probability estimates were enabled to facilitate probabilistic evaluation of predictions, and a fixed random seed ensured reproducibility.

2.4 Evaluations metrics

For the face mask type classification task, Unweighted Average Recall (UAR) was used as the primary evaluation metric. This metric accounts for class imbalance by computing the average recall across all classes. The confusion matrix was also used to analyze misclassifications between mask categories (e.g., surgical, cloth, N95). UAR is defined as follows:

$$\text{UAR} = \frac{1}{C} \sum_{i=1}^{C} \frac{TP_i}{TP_i + FN_i} \tag{1}$$

where C is the number of classes, TP_i is the number of true positives for class i and FN_i is the number of false negatives for class i.

3 Results

Table 1 presents the Unweighted Average Recall (UAR) for the face mask classification task. As baseline, results in (Mallol-Ragolta, Spiesberger, et al. 2024) are included. On the test set, the X-vector combined with logistic regression outperformed all other systems, achieving a UAR of 55.86%. Among the proposed systems, both ResNet-TDNN and X-vector models consistently outperformed ECAPA-TDNN across all evaluated classifiers.

The results for the coverage area classification task are shown in Table 2. The highest UAR values for both development and test sets were obtained using the X-vector model trained with logistic regression, with 57.32% and 54.80%, respectively. ResNet-TDNN models also demonstrated competitive performance, especially when paired with linear SVC or logistic regression. In contrast, ECAPA-TDNN based systems consistently yielded the lowest performance across all classifier combinations.

Table 1: Performance (UAR %) on the face mask classification task using different feature sets and models.

Method	**Features**	**Model**	**Dev**	**Test**
(Mallol-Ragolta, Urbach, et al. 2023)	eGeMAPS	SVCrbf	49.2	49.3
(Mallol-Ragolta, Spiesberger, et al. 2024)	S	$Proto.Enc_{S}^{FT}$	**55.1**	49.9
Ours	ResNet-TDNN	LR	53.75	50.42
		MLP	54.30	51.68
		SVC (Linear)	53.03	51.26
Ours	ECAPA-TDNN	LR	45.27	49.12
		MLP	44.39	44.05
		SVC (Linear)	45.19	47.08
Ours	X-vector	LR	54.21	**55.86**
		MLP	51.51	51.37
		SVC (Linear)	54.62	54.05

Table 3 summarizes the performance on the combined face mask and coverage area classification task. This task appears to be more challenging, with all systems showing lower UAR values compared to the individual classification tasks. The ResNet model combined with logistic regression achieved the highest UAR on the development set (40.93%), while the X-vector model with logistic regression attained the highest test performance (41.58%). As observed in the previous tasks, ECAPA-TDNN based systems continued to underperform relative to the other embedding methods. Overall, the results demonstrate that the X-vector embedding, particularly when paired with logistic regression, provides the most robust performance across all classification tasks. The ECAPA-TDNN embedding was consistently the least effective. While prior approaches such as

Table 2: Performance (UAR %) on the coverage area classification task using different feature sets and models.

Method	**Features**	**Model**	**Dev**	**Test**
(Mallol-Ragolta, Urbach, et al. 2023)	eGeMAPS	NNC	53.8	47.8
(Mallol-Ragolta, Spiesberger, et al. 2024)	S ⊕ FC ⊕ FB	$Proto.Enc^{\mathrm{F.S}}_{\mathrm{CNN.FT}}$	50.4	45.0
Ours	ResNet-TDNN	LR	51.87	53.89
		MLP	50.17	50.56
		SVC (Linear)	52.39	52.25
Ours	ECAPA-TDNN	LR	46.65	49.63
		MLP	47.82	49.31
		SVC (Linear)	47.06	49.44
Ours	X-vector	LR	**57.32**	**54.80**
		MLP	54.51	53.13
		SVC (Linear)	57.22	54.31

Table 3: Performance (UAR %) on the face mask and coverage area classification task using different feature sets and models.

Method	**Features**	**Model**	**Dev**	**Test**
(Mallol-Ragolta, Urbach, et al. 2023)	eGeMAPS	NNC	31.90	35.00
(Mallol-Ragolta, Spiesberger, et al. 2024)	S	$Proto.CIF^{\mathrm{S}}_{\mathrm{PT}}$	36.00	31.60
Ours	ResNet-TDNN	LR	**40.93**	37.04
		MLP	39.19	40.16
		SVC (Linear)	37.60	39.11
Ours	ECAPA-TDNN	LR	34.96	34.42
		MLP	34.72	31.30
		SVC (Linear)	34.60	33.55
Ours	X-vector	LR	38.36	**41.58**
		MLP	37.70	40.51
		SVC (Linear)	38.06	40.85

prototypical encoders with feature transformations showed strong results on specific tasks, they did not generalize as well across all conditions.

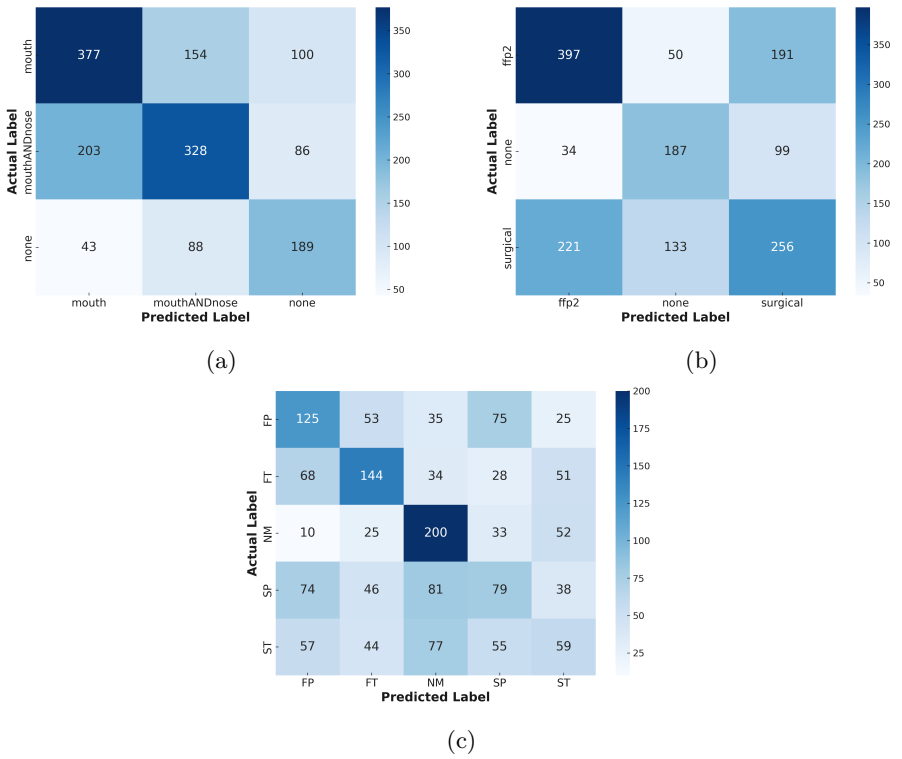


Fig. 3: Confusion matrices of the best-performing models (X-vector and LR) on the validation set for (a) face mask type and coverage area recognition, (b) face mask type recognition, and (c) face mask type and coverage area recognition.

As illustrated in Figure 3, the confusion matrices summarize the best classification performance obtained with X-vector embeddings and logistic regression across the three tasks. Panel (a) presents the results for face mask coverage area, panel (b) shows mask type classification, and panel (c) depicts the combined task. Compared to other embedding approaches, this setup provides better recognition of mask type and coverage. Surgical and FFP2 masks are generally separated more effectively, while most errors arise in distinguishing partial from total coverage within the same mask type (e.g., *SP* vs. *ST*, *FP* vs. *FT*). These outcomes suggest that embeddings capture material-related differences more consistently than coverage differences, highlighting the need for refined modeling to improve coverage detection.

4 Discussion

The experimental results across the three classification tasks highlight consistent trends in the performance of different speaker embedding models and classifiers.

First, the X-vector embedding demonstrated the most reliable performance overall. When paired with logistic regression, it achieved the highest UAR values on both the mask and coverage area tasks, as well as on the combined task. This suggests that X-vector embeddings, originally designed for speaker verification, provide a robust feature representation that transfers effectively to related classification problems. Logistic regression also proved to be a strong and stable classifier in this context, likely because of its ability to handle high-dimensional embeddings without overfitting.

Second, ResNet-TDNN models also produced competitive outcomes, particularly on the coverage area task. Their performance with linear SVC and logistic regression indicates that convolutional and time-delay architectures can extract meaningful representations for classification tasks of this type. However, compared to X-vectors, their performance was slightly less consistent across datasets and tasks.

Third, ECAPA-TDNN embeddings consistently underperformed relative to both X-vector and ResNet-TDNN. This result is notable because ECAPA-TDNN models often achieve state-of-the-art performance in speaker verification tasks. The weaker performance observed here suggests that the fine-grained channel attention mechanisms in ECAPA-TDNN may not generalize well to face mask and coverage area classification tasks, possibly due to differences in acoustic variability and task-specific feature requirements.

Another important observation is that the combined face mask and coverage area classification task proved more challenging than the individual tasks. Across all models, UAR values dropped, indicating that learning to jointly classify both attributes increases task complexity. This performance decline highlights the difficulty of multi-attribute classification using current embeddings and suggests that specialized multi-task learning approaches may be required to improve generalization.

Finally, while prior approaches based on prototypical encoders with feature transformation modules achieved strong results on individual tasks, their effectiveness did not extend to broader conditions. In contrast, the X-vector embedding showed more stable performance across all experimental setups. This finding emphasizes the value of embedding generalization and the importance of selecting models that balance task-specific accuracy with robustness.

Overall, the study demonstrates that embedding choice and classifier pairing strongly influence performance. X-vector embeddings combined with logistic regression offer the most balanced and reliable results, while ECAPA-TDNN embeddings require further investigation to understand their limitations in this classification setting

5 Conclusion

This study explored the use of deep speaker embeddings for face mask detection from speech, focusing on mask type, coverage area, and their combination. Experiments were performed using the MASCFLICHT corpus, with X-vector, ECAPA-TDNN, and ResNet-TDNN models as feature extractors. Results demonstrated that X-vector embeddings consistently outperform other approaches across all tasks and classifiers, with logistic regression providing the most stable back-end performance. The ECAPA-TDNN model showed limited utility in this context, underperforming in all evaluated scenarios. While prototypical encoder models from prior work yielded competitive results in individual classification tasks, they did not generalize well to more complex, joint classification tasks. These results suggest that speech-based mask detection is feasible using speaker embeddings, particularly X-vectors. **Future work** will focus on fine-tuning speaker embeddings on mask-related data to capture subtle acoustic changes more effectively. Multi-task learning will also be explored, training mask type and coverage detection jointly with related tasks such as speaker verification to improve generalization. Domain adaptation and temporal modeling will be considered to enhance robustness under real-world conditions.

Acknowledgements

This work was partly funded by project no. K143075, which has been implemented with the support provided by the National Research, Development, and Innovation Fund of Hungary, financed under the K_22 funding scheme and the CELSA project titled: “Identification of language-independent speech and language biomarkers characteristic of dementia”.

References

Abed, M. H. and D. Sztahó (2024). “Deep Speaker Embeddings for Speaker Verification of Children”. In: *International Conference on Text, Speech, and Dialogue*. Springer, pp. 58–69. DOI: 10.1007/978-3-031-70566-3_6.

Abed, M. H. and D. Sztahó (2023). “Effects of emotional speech on forensic voice comparison using deep speaker embeddings”. In: *Hungarian Computational Linguistics Conference 19th*, pp. 159–170. DOI: http://acta.bibl.u-szeged.hu/78411.

Chowdhury, J. H., S. Ramanna, and K. Kotecha (2025). “Speech emotion recognition with light weight deep neural ensemble model using hand crafted features”. In: *Scientific Reports* 15.1, p. 11824.

Desplanques, B., J. Thienpondt, and K. Demuynck (Oct. 2020). “ECAPATDNN Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification”. In: *Interspeech2020*. interspeech_2020. ISCA. DOI: 10.21437/interspeech.2020-2650.

Fejes, A., A. Saraiva, and J. Devenson (2024). "Assessing the Impact of Different Face Masks on Results of Forensic Automatic Speaker Recognition Systems". In: *Preprints*. DOI: 10.20944/preprints202410.0067.v1. URL: https://doi.org/10.20944/preprints202410.0067.v1.

Fiorella, M. L., G. Cavallaro, V. Di Nicola, and N. Quaranta (2023). "Voice differences when wearing and not wearing a surgical mask". In: *Journal of Voice* 37.3, 467–e1.

Habib, S. et al. (2022). "An efficient and effective deep learning-based model for real-time face mask detection". In: *Sensors* 22.7, p. 2602.

Mallol-Ragolta, A., A. Spiesberger, and B. Schuller (2024). "Face Mask Type and Coverage Area Recognition from Speech with Prototypical Networks". In: *Proc. IberSPEECH 2024*, pp. 131–135.

Mallol-Ragolta, A., N. Urbach, S. Liu, A. Batliner, and B. W. Schuller (2023). "The MASCFLICHT Corpus: Face Mask Type and Coverage Area Recognition from Speech". In: *Proc. Interspeech*, pp. 2358–2362. DOI: 10.21437/Interspeech.2023-1438.

Markitantov, M., E. Ryumina, D. Ryumin, and A. Karpov (2022). "Biometric Russian Audio-Visual Extended MASKS (BRAVE-MASKS) Corpus: Multimodal Mask Type Recognition Task." In: *INTERSPEECH*, pp. 1756–1760.

Mihajlik, P. et al. (2024). "On disfluency and non-lexical sound labeling for end-to-end automatic speech recognition". In: *Interspeech 2024*, pp. 1270–1274.

Ravanelli, M., T. Parcollet, P. Plantinga, et al. (2021). "SpeechBrain: A General-Purpose Speech Toolkit". In: *arXiv preprint arXiv:2106.04624*.

Saraiva, A., A. Fejes, and J. Devenson (2024). "Impact of Face Masks on Acoustic Parameters for Forensic Speaker Recognition". In: *Preprints*. DOI: 10.20944/preprints202409.1261.v1. URL: https://doi.org/10.20944/preprints202409.1261.v1.

Shekaraiah, S. and K. Suresh (2024). "Effect of face mask on voice production during COVID-19 pandemic: a systematic review". In: *Journal of Voice* 38.2, pp. 446–457.

Snyder, D., D. Garcia-Romero, D. Povey, and S. Khudanpur (2017). "Deep neural network embeddings for text-independent speaker verification". In: *Interspeech*, pp. 999–1003. DOI: DOI:10.21437/Interspeech.2017-620.

Snyder, D., D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur (Apr. 2018). "X-Vectors: Robust DNN Embeddings for Speaker Recognition". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 5329–5333. DOI: DOI:10.1109/ICASSP.2018.8461375.

Su, X. et al. (2022). "Face mask detection and classification via deep transfer learning". In: *Multimedia Tools and Applications* 81.3, pp. 4475–4494.

Villalba, J. et al. (2020). "State-of-the-art speaker recognition with neural network embeddings in NIST SRE18 and Speakers in the Wild evaluations". In: *Computer Speech & Language* 60, p. 101026. DOI: https://doi.org/10.1016/j.csl.2019.101026.

Yu, D. and L. Deng (2016). *Automatic speech recognition*. Vol. 1. Springer.

Poszter, laptopos bemutató

A férfi és női nyelvhasználat jellegzetességei a spontán beszédben

Vincze Veronika

HUN-REN–SZTE Mesterséges Intelligencia Kutatócsoport
Szeged, Árpád tér 2.
vinczev@inf.u-szeged.hu

Kivonat Ebben a cikkben feltérképezzük a nők és férfiak nyelvhasználatának jellegzetességeit egy nagyméretű, spontán beszédet tartalmazó adatbázis segítségével. A StaffTalk korpuszban kézzel vannak annotálva különféle pragmatikai és szemantikai információk, melyeket szintén górcső alá veszünk a nemek közti nyelvhasználati különbségek felderítése céljából. Eredményeink alapján kijelenthetjük, hogy vannak szignifikáns különbségek a férfi-női nyelvhasználat terén, ugyanakkor az egyéni nyelvhasználat szerepe sem hanyagolható el: a nők például bonyolultabb mondatszerkesztést alkalmaznak általában, míg a férfiak több indulatszót használnak.
Kulcsszavak: férfi–női nyelvhasználat, pragmatika, szemantika, spontán beszéd

1. Bevezetés

A szociolingvisztikában már régóta kutatott terület a nemek közti különbségek a nyelvhasználatban (Labov, 1972). Eleinte a szociolingvisztika interjú módszertanát használva állapítottak meg eltéréseket a két nem nyelvhasználatában (lásd pl. Lakoff (1975) és Trudgill (1972)). Ezek alapján a nők többet beszélnek, nyelvük finomkodó, udvarias, kommunikációjukra a bizonytalanság jellemző (Milroy és Milroy, 1992). A magyarra is számos szociolingvisztikai fókuszú kutatás született (Huszár, 2009), korpusznyelvészeti alapokon azonban kevéssé vizsgálták meg a kérdést a magyar nyelv vonatkozásában, hiszen viszonylag kevés felhasználható korpusz áll rendelkezésre a témában.

Ebben a munkában a nők és férfiak nyelvhasználatának jellegzetességeit vizsgáljuk egy nagyméretű, spontán beszédet tartalmazó adatbázis, a StaffTalk (Szabó és mtsai, 2021) segítségével. Alapvetően az alábbi kutatási kérdésekre keressük a választ:

- Vannak-e érdemi /szignifikáns különbségek a női-férfi nyelvhasználat között?
- Mennyire erős az egyének idiolektusának hatása?
- Milyen egyéni és csoportos jellegzetességek mutathatók ki az adatok alapján?

A cikkben először röviden ismertetjük a szakirodalmi hátteret, majd a StaffTalk korpuszt mutatjuk be tömören. Ezután rátérünk vizsgálatunkra, mely során

kitérünk morfológiai, szintaktikai, szemantikai és pragmatikai különbségekre is. A cikket egy összefoglalóval és kitekintéssel zárjuk.

2. Kapcsolódó irodalom

A nők és férfiak nyelvhasználata közti eltéréseket régóta vizsgálják sokféle szempontból, többek között fonetikai és szociolingvisztikai kutatások sokasága foglalkozik a kérdéssel (Huszár, 2009).

Angol nyelvre például Newman és mtsai (2008) írott szövegben vizsgálta részletesen a férfiakra és nőkre jellemző sajátságokat. Azt találták, hogy a nők több névmást és pszichológiai igét, valamint több tagadást használnak, ugyanakkor a férfiak általában hosszabb szavakat, több névelőt és prepozíciót, valamint több indulatszót használnak. Az udvariassági stratégiák eltérő használatait pedig többek között Holmes (1995) tárgyalja részletesen.

A magyar nyelv vonatkozásában is a férfi és női nyelvhasználat különbségeire már több munka is felhívta a figyelmet. Huszár Ágnes átfogó elemzést nyújt a témáról (Huszár, 2009), illetve emellett számos kutatás fókuszál egy-egy kisebb problémakörre (Huszár, 2013). Például, Porkoláb (2017) blogbejegyzéseket vizsgál, ahol kitér a két nem nyelvhasználati különbségeire is, vagy egy másik tanulmány a „nőies" munkahelyen dolgozó férfiak nyelvhasználati sajátságaira világít rá (Kovács és Sümegi, 2012). Huszár (2020) szintén a munkahelyi nyelvhasználatot kutatja kommunikációs szempontból, hasonlóan Schleicher (2003) munkájához, Háhn (2008) pedig a munkahelyi levelezésre fókuszálva állapít meg nemi különbségeket.

A hazai spontánbeszéd-kutatások eddig jobbára a fonetikai, illetve akusztikus sajátságok elemzésére irányulnak (Deme és Markó, 2013). Az általunk is használt StaffTalk korpusz udvariassági és bizonytalansági annotációit ugyan részletesen elemzi Vincze és mtsai (2021), azonban a nemek közti összevetésre nem terjed ki. Kutatásunk tehát újdonságot képvisel ilyen téren.

3. Módszerek

A StaffTalk korpusz hanganyagát munkahelyi beszélgetések képezik (Szabó és mtsai, 2021). Egy középiskolában 21 tanár vállalta, hogy a kollégáikkal folytatott beszélgetéseiket néhány hétig okosóra segítségével rögzítsék a kutatók. Ezen beszélgetéseket később nyelvészek leiratozták, és a létrejött leiratokat szemantikai és pragmatikai kategóriákkal annotálták. A felvételeket minőségi ellenőrzésnek vetették alá a korpusz készítői, a rosszul hallható, túlságosan zajos felvételeket nem dolgozták fel, valamint az esetlegesen többször is rögzített beszélgetéseket igyekeztek kiszűrni. A korpusz végső változata így összesen 105 órányi hanganyagot tartalmaz.

Mivel a korpuszban meg vannak jelölve a beszélők és a beszélőváltások is, lehetőségünk volt beszélők szerint feldarabolni az anyagot. Az így beszélőkre bontott anyag képezi jelen cikk vizsgálatának tárgyát. Összesen 6 férfi és 15 nő

beszédanyagát elemezzük a továbbiakban. Az egyes beszélők neveit adatvédelmi okok miatt lecseréltük, így senki sem szerepel beazonosíthatóan a példákban.

A férfiak és nők nyelvhasználatának különbségét több szemszögből is vizsgáltuk. Ehhez lefuttattuk a magyarlanc elemzőt (Zsibrita és mtsai, 2013) a szöveges átiratokon, így megkapva a morfológiai és szintaktikai elemzéseket. Az adatok és a korpuszhoz tartozó annotációk alapján lehetőségünk nyílt az alábbiak vizsgálatára:

- szó- és mondatszám;
- szófaji eloszlások;
- szintaktikai viszonyok eloszlása;
- udvariassági stratégiák;
- bizonytalanságjelző elemek.

A következőkben ezeket az eredményeket mutatjuk be.

4. Eredmények

Az alábbiakban bemutatjuk a beszélőkre bontott korpusz statisztikáit, adatait és azok részletes elemzését.

4.1. Szó- és mondatszám

A korpusz összesen 53 903 mondatot és 647 504 tokent, illetve 468 250 szót (írásjelek nélkül) tartalmaz. Ebből 22 292 mondatot és 194 563 szót a férfiak produkáltak, míg 31 611 mondatot és 273 687 szót a nők. Feltűnő azonban, hogy nem egyenletes a beszélőnkénti beszédmennyiség eloszlása, amit jól szemléltet az 1. táblázat és az 1. ábra.[1] Az eltérő beszédmennyiségből levonható következtetések pontosságát emiatt statisztikai tesztekkel szeretnénk igazolni.

Amint a fentiek mutatják, a korpusz sajátságos összetételű: egyetlen beszélőtől, Ivántól származik a korpusz körülbelül 1/3 része. Ő valószínűleg központi szerepet tölthet be a munkahelyi közösség életében, mivel sok párbeszédben részt vett, és igen kommunikatív. Feltűnő az is, hogy 7 beszélő felelős a korpusz méretének több mint 70%-áért, azaz a beszélők harmadától származik a beszélt nyelvi adatok több mint kétharmada.

Ami a nemi különbségeket illeti, a női adatközlők beszélnek többet. Ez nem meglepő, mivel jóval több a női résztvevők aránya a korpuszban, viszont megjegyzendő, hogy összességében egy férfi (Iván) beszél a legtöbbet. Emiatt kérdéses, hogy Iván idiolektusa mennyire befolyásolja az adatokat. A túláltalánosítások elkerülésére a következőkben vizsgálati eredményeinket Iván adataival és azok nélkül is bemutatjuk, kiemelve ugyanakkor, hogy egy egyén nyelvhasználata is komoly befolyással lehet a korpuszra.

[1] Megjegyezzük, hogy ismereteink szerint nem áll rendelkezésre olyan, magyar nyelvű, teljes mértékben spontán beszédet tartalmazó korpusz, ahol az egyes beszélők közel azonos mennyiségű beszédet produkálnak. Ugyanakkor kérdéses, hogy egy teljesen spontán környezetben rögzített anyag esetében egyáltalán lehetséges-e ilyen elvárásnak megfelelni.

1. táblázat. Szó- és mondatszám beszélőre bontva.

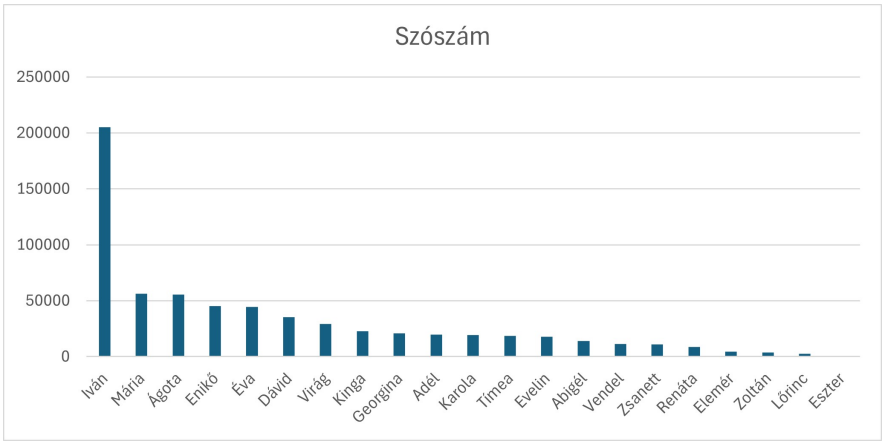
Beszélő	**Szószám**	%	**Mondatszám**	%	**Átlagos mondathossz**
Iván	155392	33,19	17210	31,93	9,03
Mária	43495	9,29	4291	7,96	10,14
Ágota	42177	9,01	4812	8,93	8,76
Enikő	31129	6,65	3770	6,99	8,26
Éva	26645	5,69	2683	4,98	9,93
Dávid	22455	4,80	2868	5,32	7,83
Virág	21175	4,52	3047	5,65	6,95
Kinga	15812	3,38	1835	3,40	8,62
Gina	15411	3,29	1723	3,20	8,94
Adél	15312	3,27	1556	2,89	9,84
Karola	12954	2,77	1625	3,01	7,97
Tímea	12210	2,61	1519	2,82	8,04
Evelin	11895	2,54	1698	3,15	7,01
Abigél	11007	2,35	1205	2,24	9,13
Vendel	8516	1,82	1160	2,15	7,34
Zsanett	7747	1,65	1044	1,94	7,42
Renáta	6027	1,29	763	1,42	7,90
Elemér	3464	0,74	406	0,75	8,53
Zoltán	2770	0,59	408	0,76	6,79
Lőrinc	1966	0,42	240	0,45	8,19
Eszter	691	0,15	40	0,07	17,28
Összesen	468250	100	53903	100	8,69
Férfiak	194563	41,55	22292	41,36	8,73
Nők	273687	58,45	31611	58,64	8,66

4.2. Szófaji elemzések

A szófaji eloszlásokat a nemek viszonylatában a 2. táblázat mutatja be, mind darabszám, mind az összes szóhoz viszonyított százalékos érték tekintetében. A különbségek statisztikailag szignifikánsak (χ^2-próba, $p < 0{,}05$). Ez alapján a nők arányaiban szignifikánsan több kötőszót használnak, mint a férfiak (t-próba, $p < 0{,}05$), illetve a nők átirataiban több írásjelet találunk, mint a férfiakéban. Ez utóbbi azzal lehet összefüggésben, hogy a nőknél magasabb a tagmondatok száma, illetve az összetett mondatok (egynél több tagmondatot tartalmazó mondatok) száma is magasabb. Mivel a tagmondatok közé írásjelet teszünk írásban, ez nem kifejezetten a spontán beszéd jellemzője, ugyanakkor mégis rámutat arra, hogy a nők valamivel bonyolultabb mondatszerkesztést alkalmaznak. További érdekesség az egyéni nyelvhasználatban, hogy Iván nélkül vizsgálva a férfiakat, nagyobb arányban használják az indulatszavakat, mint a nők.

4.3. Szintaktikai viszonyok

A 3. táblázat szemlélteti a függőségi viszonyok eloszlását a nemek viszonylatában, szintén darabszám és az összes szóhoz viszonyított arány szerint. A különbségek statisztikailag szignifikánsak (χ^2-próba, $p < 0{,}05$).



1. ábra: Szavak száma egyénenként.

2. táblázat. Szófaji eloszlások.

	Nő	**Ffi**	**Ffi Iván nélkül**	**Nő %**	**Ffi %**	**Ffi Iván nélkül %**
ADJ	13375	9294	1843	4,89	4,78	4,71
ADP	873	624	112	0,32	0,32	0,29
ADV	39857	27367	5106	14,56	14,07	13,04
AUX	33	27	3	0,01	0,01	0,01
CONJ	15754	10380	2035	5,76	5,34	5,20
DET	16057	11713	2092	5,87	6,02	5,34
INTJ	10759	6915	2128	3,93	3,55	5,43
NOUN	24767	18695	3555	9,05	9,61	9,08
NUM	3850	2561	486	1,41	1,32	1,24
PART	1345	963	178	0,49	0,49	0,45
PRON	25906	18726	3351	9,47	9,62	8,55
PROPN	2910	2124	472	1,06	1,09	1,20
PUNCT	70062	49791	11353	25,60	25,59	28,98
SCONJ	11435	7645	1454	4,18	3,93	3,71
SYM	6	1	0	0,00	0,00	0,00
VERB	35329	26928	4608	12,91	13,84	11,76
X	1369	809	395	0,50	0,42	1,01

Különbséget mutat a mellérendelő viszonyok használata a két nem között: a férfiaknál kevesebb a mellérendelő kötőszó, ugyanakkor több a mellérendelés. Ha pedig a mondatszerkesztést vizsgáljuk, a férfiak több egyszerű mondatot használnak, mint a nők: a férfiak esetében a mondatok 67%-a egyszerű mondat, a nők esetében 62%. Ez utóbbi statisztikailag is szignifikáns eltérés (t-próba, $p < 0{,}05$).

3. táblázat. Szintaktikai viszonyok eloszlása a nemek között.

	Nő	**Ffi**	**Ffi Iván nélkül**	**Nő %**	**Ffi %**	**Ffi Iván nélkül %**
APPEND	3450	2325	648	1,26	1,19	1,65
ATT	26819	18926	3500	9,80	9,73	8,94
CONJ	26464	17610	3356	9,67	9,05	8,57
COORD	14356	11267	2357	5,25	5,79	6,02
DAT	1906	1274	240	0,70	0,65	0,61
DEP	18	11	4	0,01	0,01	0,01
DET	17082	12443	2210	6,24	6,40	5,64
FROM	124	118	25	0,05	0,06	0,06
INF	3061	2130	356	1,12	1,09	0,91
LOCY	2274	1629	310	0,83	0,84	0,79
MODE	18721	12857	2461	6,84	6,61	6,28
NE	314	149	35	0,11	0,08	0,09
NEG	6551	4155	867	2,39	2,14	2,21
NUM	70	64	19	0,03	0,03	0,05
OBJ	9217	6778	1182	3,37	3,48	3,02
OBL	8727	6918	1113	3,19	3,56	2,84
PRED	2125	1494	273	0,78	0,77	0,70
PREVERB	4074	2879	447	1,49	1,48	1,14
PUNCT	69777	49642	11298	25,50	25,51	28,84
QUE	474	348	81	0,17	0,18	0,21
ROOT	31409	22118	5024	11,48	11,37	12,83
SUBJ	15089	11042	2017	5,51	5,68	5,15
TFROM	150	123	21	0,05	0,06	0,05
TLOCY	10510	7629	1208	3,84	3,92	3,08
TO	593	430	81	0,22	0,22	0,21
TTO	329	203	38	0,12	0,10	0,10

4.4. Udvariassági viszonyok

A korpusz az alábbi udvariassági kategóriákra van kézzel annotálva (Vincze és mtsai, 2021):

- Beszédaktusok:
  - ígéret / ajánlat (jövőbeli pozitív cselekedetre utalás)
  - figyelmeztetés / fenyegetés (jövőbeli negatív cselekedetre utalás)
  - kérés / parancs / kívánság
  - panasz / vád / kritika / sértés (negatív vélemény kifejezése negatív jelentéstartalmú szavakkal)
  - dicséret / bók (pozitív vélemény kifejezése pozitív jelentéstartalmú szavakkal)
  - bocsánatkérés
  - köszönetnyilvánítás
- Reakciók:
  - elfogadás / egyetértés
  - visszautasítás / egyet nem értés

  - hárítás
- Irónia:
  - irónia (pozitív jelentéstartalmú szavakkal kifejezett negatív tartalom)
  - antiirónia (negatív jelentéstartalmú szavakkal kifejezett pozitív értékelés)
- Interakciós elemek:
  - figyelem felhívása (fontos vagy érdekes mondandó jelzése a partner felé)
  - üdvözlés / elköszönés

Az udvariassági viszonyokat egyénre lebontva a 4-6. táblázat szemlélteti, nemekre lebontva pedig a 7. táblázat. A nemek közti különbségek szignifikánsak (χ^2-próba, $p < 0{,}05$).

4. táblázat. Udvariassági viszonyok eloszlása egyénenként 1.

Típus	**Iván**	**Dávid**	**Elemér**	**Lőrinc**	**Vendel**	**Zoltán**	**Ágota**
elfogadás/egyetértés	2408	625	48	29	257	30	638
antiirónia	8						
bocsánatkérés	205	42	15	3	9	6	36
figyelem felhívása	512	22	20	4	11	8	106
hárítás	86	12	5	1	2	6	13
panasz/vád/kritika	845	69	25	8	29	14	311
dicséret/bók	251	23	7	7	14	9	104
üdvözlés/elköszönés	375	49	4	1	14	16	49
irónia	177	7	4	5	2	3	15
ígéret/ajánlat	538	60	10	2	20	9	73
visszautasítás	217	54	10	7	20	13	70
kérés/parancs	513	45	9	10	17	13	123
köszönetnyilvánítás	206	38	8	3	9	7	43
figyelmeztetés	48	5	2		6	7	12

Az adatok azt mutatják, hogy az egyetértés a leggyakoribb kategória mindkét nem esetén. Ez nem meglepő, hiszen egy munkahelyi kollektíva tagjai beszélgetnek egymással, így a közös munka érdekében fontos az együttműködés a partnerekkel, kollégákkal. Azt is láthatjuk, hogy a férfiaknál gyakoribb az üdvözlés/elköszönés, illetve a figyelem felhívása. Ez utóbbi főleg Ivánnak köszönhető, az ő egyéni nyelvhasználatában gyakori ez a beszédaktus (az összes ilyen kategória 43%-a köthető a nevéhez). A nők több bókot és dicséretet alkalmaznak, mint a férfiak, ez is a pozitív udvariasság része, mely a közösséget, összetartozást erősíti a csoporttagok között. Az iróniáról pedig elmondható, hogy nagy arányban Iván alkalmazza, az összes ironikus megjegyzés fele tőle származik a korpuszban. Ugyanez igaz az antiiróniára is.

4.5. Nyelvi bizonytalanság

A nyelvi bizonytalanság annotációja az alábbiakban foglalható össze (vö. Vincze és mtsai (2021); Vincze (2014)):

5. táblázat. Udvariassági viszonyok eloszlása egyénenként 2.

Típus	Abigél	Adél	Eszter	Enikő	Gina	Éva
elfogadás/egyetértés	123	275	12	599	281	735
antiirónia		1		1	3	
bocsánatkérés	10	23		60	10	24
figyelem felhívása	63	22		58	21	54
hárítás	8	3		24	8	7
panasz/vád/kritika	59	44	1	260	93	93
dicséret/bók	20	15		115	33	18
üdvözlés/elköszönés	7	17	1	23	17	44
irónia	15	20	1	15	17	5
ígéret/ajánlat	38	20		132	58	91
visszautasítás	28	40	3	127	63	108
kérés/parancs	23	41	1	123	52	117
köszönetnyilvánítás	10	13	2	45	14	54
figyelmeztetés	4	4		11	7	4

6. táblázat. Udvariassági viszonyok eloszlása egyénenként 3.

Típus	Kinga	Karola	Evelin	Mária	Renáta	Tímea	Virág	Zsanett
elfogadás/egyetértés	289	271	329	706	151	367	455	203
antiirónia			1	3	1	1		1
bocsánatkérés	12	13	8	46	7	11	26	2
figyelem felhívása	28	40	35	114	9	14	30	15
hárítás	4	5	7	17	2	4	13	3
panasz/vád/kritika	91	25	31	546	26	65	89	12
dicséret/bók	52	16	15	184	11	18	98	8
üdvözlés/elköszönés	20	42	9	39	5	15	86	15
irónia	2	8	1	27	12	9	6	2
ígéret/ajánlat	62	43	25	51	29	55	70	18
visszautasítás	64	68	45	120	21	56	43	22
kérés/parancs	56	59	18	60	15	46	74	28
köszönetnyilvánítás	10	16	4	23	9	15	41	6
figyelmeztetés	10	1	1	6	3	6	7	2

– Szemantikus bizonytalanság:
  - episztemikus: a világtudásunk alapján nem tudjuk eldönteni, hogy igaz-e vagy hamis az állítás (*talán, valószínűleg, lehetséges*)
  - doxasztikus: hiedelmek, vélemény kifejezése (*hisz, gondol, vél, szerint*)
  - feltételes: egy adott feltételhez kötött az állítás igazságértéke (*ha... akkor*)
  - vizsgálat: pl. kutatási kérdés egy tudományos cikkben (*megvizsgál, elemez*)
– Diskurzusszintű bizonytalanság:
  - weasel: bizonytalan információforrás vagy szereplő a cselekvésben (*valaki, egyesek*)
  - hedge: mennyiségek vagy minőségek homályos jelölése (*sok, gyakori*)

7. táblázat. Udvariassági viszonyok eloszlása a nemek között.

	Nő	Ffi	Ffi Iván nélkül	Nő %	Ffi %	Ffi Iván nélkül %
elfogadás / egyetértés	5434	3397	989	44,11	41,24	53,49
panasz / vád / kritika / sértés	1746	990	145	14,17	12,02	7,84
visszautasítás / egyet nem értés	878	321	104	7,13	3,90	5,62
kérés / parancs / kívánság	836	607	94	6,79	7,37	5,08
ígéret/ajánlat	765	639	101	6,21	7,76	5,46
dicséret/bók	707	311	60	5,74	3,78	3,24
figyelem felhívása	609	577	65	4,94	7,00	3,52
üdvözlés / elköszönés	389	459	84	3,16	5,57	4,54
köszönetnyilvánítás	305	271	65	2,48	3,29	3,52
bocsánatkérés	288	280	75	2,34	3,40	4,06
irónia	155	198	21	1,26	2,40	1,14
hárítás	118	112	26	0,96	1,36	1,41
figyelmeztetés / fenyegetés	78	68	20	0,63	0,83	1,08
antiirónia	12	8	0	0,10	0,10	0,00

- peacock: bizonyít(hat)atlan állítás vagy túlzás (*gyönyörűszép*, *botrányos*)

A bizonytalanságtípusokat egyénre lebontva a 8. táblázat szemlélteti, nemekre lebontva pedig a 9. táblázat. A nemek közti különbségek ebben az esetben is szignifikánsak (χ^2-próba, $p < 0{,}05$).

Összességében azt láthatjuk, hogy a nők beszédében több diskurzusszintű bizonytalanság fordul elő, azaz arányaiban több weasel, peacock és hedge jelenik meg náluk, mint a férfiaknál, Iván nélkül tekintve. Ebből az következik, hogy Iván beszédében a bizonytalanság nagyobb arányban van jelen, mint átlagosan a férfiaknál. Mindemellett, az egyéni nyelvhasználat jellegzetességeit is tetten érhetjük: Elemér az átlaghoz képest sokkal nagyobb arányban használ doxasztikus kulcsszavakat (a korpusz összes doxasztikus kulcsszava közül körülbelül 12% nála fordul elő).

5. Az eredmények megvitatása

A fenti eredmények több érdekes következtetésre is utalnak. Először is, a korpusz felépítésében láthatjuk, hogy a tanári közösségben van egy központi alak, Iván, aki nagyon sok párbeszédben vesz részt, így a tőle származó adatok erősen dominálnak az adatbázisban. A korpusz tervezésekor és a hangfelvételek készítésekor ezt természetesen nem lehetett előre látni, azonban az adatok felhasználásánál és értelmezésénél mindenképp érdemes figyelembe venni ezt a tényezőt.

8. táblázat. Nyelvi bizonytalanság eloszlása egyénekre lebontva.

Név	feltételes	doxaszt.	episzt.	hedge	vizsgálat	peacock	weasel	összesen
Iván	1581	1329	698	2404	9	866	2610	9497
Dávid	216	205	81	196	2	54	231	985
Elemér	51	655	20	43		14	43	826
Lőrinc	19	19	20	43		6	23	130
Vendel	75	92	52	86		42	116	463
Zoltán	8	17	7	22		21	35	110
Ágota	474	479	160	505		167	582	2367
Abigél	80	85	24	102	1	60	118	470
Adél	139	120	65	202	1	70	170	767
Eszter	6	3	3					12
Enikő	480	533	167	565	2	141	449	2337
Gina	161	172	64	225		94	257	973
Éva	281	310	86	262	1	93	372	1405
Kinga	205	188	80	210		91	249	1023
Karola	99	150	50	171	5	68	164	707
Evelin	131	182	32	183		91	243	862
Mária	378	488	181	678		243	712	2680
Renáta	63	58	36	84		53	67	361
Tímea	81	117	51	162		70	156	637
Virág	123	238	87	387	1	115	203	1154
Zsanett	43	83	19	79		23	94	341

9. táblázat. Nyelvi bizonytalanság eloszlása a nemek között.

	Nő	Ffi	Ffi Iván nélkül	Nő %	Ffi %	Ffi Iván nélkül %
feltételes	2744	1950	369	17,05	16,24	14,68
doxasztikus	3206	2317	988	19,92	19,29	39,30
episztemikus	1105	878	180	6,87	7,31	7,16
vizsgálat	11	11	2	0,07	0,09	0,08
hedge	3815	2794	390	23,70	23,26	15,51
peacock	1379	1003	137	8,57	8,35	5,45
weasel	3836	3058	448	23,83	25,46	17,82
szemantikus	7066	5156	1539	43,90	42,93	61,22
diskurzus	9030	6855	975	56,10	57,07	38,78

A szófaji és mondattani sajátosságokból az látszik kitűnni, hogy a nők általában hosszabb és összetettebb mondatokat használnak. Mivel azonban beszélt nyelvről van szó, az élőbeszédben pedig nem mindig egyértelműek a mondathatárok, csak a hangsúlyozás és szünetek alapján lehet erre következtetni. Mivel írásban a szöveg szerzője jelöli ki a mondathatárokat, érdemes lenne ezt a megállapítást összevetni egy írott anyagon készült felméréssel is, hogy a női és férfi fogalmazók között is látszik-e hasonló különbség.

Az udvariassági annotációkból azt szűrhetjük le, hogy a pozitív udvariasság esetei (pl. elfogadás, egyetértés, dicséret) gyakrabban fordulnak elő, azaz az olyan aktusok, ahol a közösségi érzés erősödik. Ez nem meglepő annak fényében, hogy

egy munkahelyi közösségben készültek a felvételek, ahol az együttműködés, jó légkör kialakítása elsődleges szereppel bír.

Érdemes azt is megfigyelni, hogy a nők általánosságban több bizonytalanságjelölőt használnak. Ennek főleg pragmatikai funkciói lehetnek az élőbeszédben, például kérések enyhítése, csoporthoz tartozás kifejezése.

Végezetül arra is térjünk ki, hogy az egyéni nyelvhasználat (valakinek a „szava járása") is kihathat az eredményekre. Például Iván gyakran alkalmazza az iróniát, míg másoknál ez kevéssé figyelhető meg, az adatbázis alapján legalábbis. Ugyanakkor Elemér sokszor használja az *azt hiszem, gondolom* kifejezéseket, így a doxasztikus bizonytalanság eredményeit befolyásolhatja ez a tendencia.

6. Összegzés

Ebben a cikkben a StaffTalk korpusz alapján vizsgáltuk meg a női és férfi beszéd néhány jellegzetességét. Találtunk szignifikáns eltéréseket a szófaji és mondattani sajátosságok terén, valamint az udvariasság és bizonytalansági stratégiák területén is. Néhány esetben arra is rámutattunk, hogy az egyéni nyelvhasználatot is figyelembe kell venni az eredmények értelmezésénél.

A kutatás eredményeit jól hasznosíthatja a szociálpszichológia, a szociopragmatika, valamint a szociológia is. További céljaink között szerepel a férfi-női nyelvhasználati jellegzetességek feltérképezése más adatbázisok segítségével, valamint egyéb nyelvi, például szemantikai jellemzők részletesebb vizsgálata a nemek tükrében.

Hivatkozások

Deme, A., Markó, A.: Lengthenings and filled pauses in Hungarian adults' and children's speech. In: Proceedings of DiSS 2013, The 6th Workshop on Disfluency in Spontaneous Speech. TMH-QPSR. vol. 54, pp. 21–24. KTH Royal Institute of Technology (2013)

Holmes, J.: Women, men and politeness. Longman, Harlow (1995)

Huszár, Á.: Bevezetés a gendernyelvészetbe. Miben különbözik és miben egyezik a férfiak és a nők nyelvhasználata és kommunikációja? Tinta, Budapest (2009)

Huszár, Á. (szerk.): A gendernyelvészet horizontja. Pécsi Tudományegyetem Nyelvtudományi Doktori Iskola, Pécs (2013)

Huszár, Á.: A gender és a kommunikáció összefüggései: kommunikáció a munkahelyen. a különbségek, avagy ki a jobb? Economica 7, 7–12 (2020)

Háhn, J.: Nyelvhasználati különbségek férfiak és nők munkahelyi e-mailjeiben. Alkalmazott Nyelvészeti Közlemények 3(1), 85–96 (2008)

Kovács, Zs., Sümegi, M.: A férfiak nyelvi viselkedése egy "nőies" munkahelyi sztenderd nyelvhasználat tükrében. Társadalmi Nemek Tudománya Interdiszciplináris eFolyóirat 2(3), 102–121 (2012)

Labov, W.: Sociolinguistic Patterns. University of Pennsylvania Press, Philadelphia (1972)

Lakoff, R.: Language and woman's place. Harper Colophon Books, New York (1975)

Milroy, L., Milroy, J.: Social network and social class: Toward an integrated sociolinguistic model. Language in Society 21, 1–26 (1992)

Newman, M.L., Groom, C.J., Handelman, L.D., Pennebaker, J.W.: Gender Differences in Language Use: An Analysis of 14,000 Text Samples. Discourse Processes 45, 211–236 (2008)

Porkoláb, Á.: Webnapló és blogbiznisz. A magyar blogoszféra netnyelvészeti vizsgálata tartalomtípusok és nemek alapján. Ph.D.-értekezés, Pécsi Tudományegyetem (2017)

Schleicher, N.: Kommunikációs stratégiák a munkahelyi alkalmazkodásban. Másképp beszélnek-e a nők és a férfiak? In: Kötő-jelek 2002 : Az Eötvös Loránd Tudományegyetem Szociológia Doktori Iskolájának Évkönyve. pp. 31–60. ELTE, Budapest (2003)

Szabó, M.K., Vincze, V., Ring, O., Üveges, I., Vit, E., Samu, F., Gulyás, A., Galántai, J., Szvetelszky, Zs., Bodor-Eranus, E.H., Takács, K.: StaffTalk: magyar nyelvű spontán beszélgetések korpusza. In: XVII. Magyar Számítógépes Nyelvészeti Konferencia. Szegedi Tudományegyetem, Szeged (2021)

Trudgill, P.: Sex, covert prestige and linguistic change in the urban British English of Norwich. Language in Society 1, 179–195 (1972)

Vincze, V.: Uncertainty detection in Hungarian texts. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. pp. 1844–1853. Dublin City University and Association for Computational Linguistics, Dublin, Ireland (Aug 2014), https://www.aclweb.org/anthology/C14-1174

Vincze, V., Üveges, I., Szabó, M.K.: Spontán beszéd szemantikai–pragmatikai sajátságainak elemzése. In: XVII. Magyar Számítógépes Nyelvészeti Konferencia. Szegedi Tudományegyetem, Szeged (2021)

Zsibrita, J., Vincze, V., Farkas, R.: magyarlanc: A toolkit for morphological and dependency parsing of Hungarian. In: Proceedings of RANLP. pp. 763–771 (2013)

Követik-e a betegek az utasításokat?

Nyerges Bálint[1], Czimbalmos Olivér[1], Tóth Gábor[1], Pálfi Áron[1], Szűcs Tamás[2], Rafael Beatrix[2], Bán János[1], Jánki Zoltán Richárd[1], Iglói Gábor[2], Farkas Richárd[1], Kőrösi Gábor[1], Szántó Zsolt[1], Kósa István[2]

[1]Szegedi Tudományegyetem, Informatikai Intézet
Szeged, Árpád tér 2.
[2]Szegedi Tudományegyetem, Szent-Györgyi Albert Klinikai Központ, Belgyógyászati Klinika, Kardiometabolikus Rehabilitációs Osztály, Szeged
6725 Szeged, Kálvári sgt 57.
{nyerges, korosig, szantozs}@inf.u-szeged.hu
kosa.istvan@med.u-szeged.hu

Kivonat A telemedicina programok sikerének egyik kulcsfontosságú eleme a betegek adherenciája, vagyis annak mértéke, hogy mennyire követik az egészségügyi szakemberek ajánlásait. Vizsgálatunk egy olyan telemedicina programra fókuszál, amelyben szív- és érrendszeri betegeket, illetve fokozott kockázatú személyeket követnek nyomon. A program résztvevői egy 5 napos intézeti ellátást követően 90 napos, dietetikusok és gyógytornászok által támogatott telemedicina követésben vesznek részt, mely során a szakemberek kéthetente telefonon konzultálnak a betegekkel, amiből rövid pár mondatos összefoglalót készítenek. Vizsgálatunk célja ezen összefoglalók alapján a betegek adherenciájának meghatározása volt. A probléma nem hagyományos értelemben vett vélemény detekciós feladat, mégis nagyon hasonlít rá, mert az adherencia az összefoglalókban gyakran valamilyen vélemény formájában jelenik meg. Ebből a gondolatból kiindulva cikkünkben megvizsgáltunk több vélemény detekciós módszert és összehasonlítottuk a teljesítményüket nagy nyelvi modellek eredményeivel. A feladatra – az aspektus orientált vélemény detekcióból motiválódva – nagy nyelvi modellek segítségével, aspektusok mentén elemeztük az összefoglalókat és a végső címkéket ezen aspektusokra épülő szabályrendszerrel határoztuk meg. Az aspektusokra építő, nagy nyelvi modell alapú rendszerrel minden másik megoldásnál jobb eredményt értünk el.
Kulcsszavak: adherencia, vélemény detekció, nagy nyelvi modellek

1. Bevezetés

Az egészségügyi ellátórendszer leterheltsége miatt különösen fontosak azok az egészségügyi megoldások és telemedicina programok, amelyek nem személyes rendelői találkozásokról, hanem távoli szakmai támogatásokról szólnak. Ezeknek a folyamatos távoli szakmai segítségnyújtó programoknak egy nagyon kritikus eleme az, hogy a páciensek mennyire követik az egészségügyi szakemberek utasításait. Ezt a tulajdonságot adherenciának nevezzük.

A mi cikkünk szív- és érrendszeri betegek, illetve fokozott kockázatú személyek (Kosa és mtsai, 2025) nyomkövetésére fókuszál egy telemedicina program keretében. Ennek a programnak a célja, hogy életmódváltással javítsák a betegek egészségügyi állapotát. A résztvevők először 5 napos intézeti ellátásban vesznek részt, utána 90 napos telemedicina szakasz következik. A program teljes időtartama alatt gyógytornászok és dietetikusok folyamatos támogatást nyújtanak a betegek számára, a telemedicina időszakában egy-két hetente telefonos konzultációk során követik a betegek fejlődését, melyekről rövid, pár mondatos összefoglalót készítenek.

A program nagy figyelmet fordít a személyre szabott kezelésnek, így az ellátás legelején pszichológiai tesztekkel felmérik a betegek mentális állapotát. A későbbi telemedicina szakasz támogatásához számos digitális eszköz áll rendelkezésre, amelyek segítségével a páciensek naplózniuk kell a betegségükre releváns adatokat. A betegek nyomkövetése érdekében ezek az adatok a következők: testsúly, táplálkozás, fizikai aktivitás és pulzus. Problémát okoz, hogy ezek az adatok sokszor hiányosak és zajosak, a páciensek különböző okok miatt nem tudják követni a szakemberek ajánlásait. Szerencsére a dietetikusok és gyógytornászok a telefonos konzultációk összefoglalóiban külön kiemelik, ha a betegek nem együttműködőek, illetve ha a hiányosságok külső tényezőkből vagy technikai okokból adódnak.

Vizsgálatunk célja ezen összefoglalók alapján a betegek adherenciájának meghatározása volt. A meghatározott adherencia értékek és a program kezdetén elvégzett pszichológiai tesztek segítségével egy pontosabb képet kaphatunk arról, hogy milyen pszichológiai jellemzők befolyásolják azt, hogy az egyes betegek mennyire tudnak bevonódni a programba, és ez alapján kik azok, akik speciális támogatást fognak igényelni, hogy el tudjanak indulni a számukra szükséges életmódváltásban.

Az adherencia nem hagyományos értelemben vett vélemény detekciós feladat, mégis nagyon hasonlít rá, mert az adherencia az összefoglalókban gyakran valamilyen vélemény formájában jelenik meg. Cikkünkben megvizsgáljuk, hogy a nagy nyelvi modellek korában mit érhetünk el hagyományos vélemény detekciós módszerekkel egy kvázi-vélemény detekciós feladaton. Az így kapott eredményeket nagy nyelvi modellek few-shot promptolt eredményeivel hasonlítjuk össze és, az aspektus orientált vélemény detekcióból motiválódva, nagy nyelvi modellek segítségével aspektusok mentén értékeljük ki az összefoglalókat, végül az aspektusokra kapott válaszok alapján hozunk döntést az adherencia címkék értékéről. A számítógépes nyelvfeldolgozás segítségével megkönnyíthetjük az egészségügyi dolgozók munkáját és kulcsfontosságú információkat tudhatunk meg a betegek adherenciájára kiható tényezőkről.

2. Kapcsolódó munkák

A betegek adherencia mérésének kiterjedt irodalma van (Martin és mtsai, 2005; Iuga és McGuire, 2014), de ez jellemzően nem a szövegfeldolgozás eszköztárára épít. Hagyományosan az adherenciát gyógyszer kiváltási szokások (Lam és

Fresco, 2015), bizonyos betegségek esetén a súlyvesztés (Szalka és mtsai, 2024), vagy akár szenzorok adataira támaszkodva, okos eszközökkel (Pal és mtsai, 2021) szokták meghatározni. A szenzori adatok mellett lehetőség van, a betegek naplózási szokásait is felhasználni, mint ebben a projektben az élelmiszer fogyasztás naplózása alapján.

A pszichológiai kutatások szerint az adherencia nemcsak klinikai, hanem pszichés és szociális tényezők függvénye is, amelyek jelentős hatást gyakorolnak a páciensek motivációjára és elköteleződésére a távgyógyászati beavatkozásokkal kapcsolatban. Míg a tartósan fennálló betegség gyakran aláássa a betegek motivációját és ezáltal fokozza a adherencia csökkenésének a kockázatát, addig a megfelelő szociális támogatás jelentős mértékben elősegíti az adherenciát. Ez utóbbi ugyanis érzelmi és gyakorlati támaszt nyújt, amely megkönnyíti a kezelés következetes betartását (Leiz és mtsai, 2022).

A program kezdetén a betegek részletes pszichológiai tesztet töltenek ki. A pszichológiai eredmények és a betegek program során tapasztalt adherenciájának a segítségével lehetőségünk van megvizsgálni, hogy az egyes pszichológiai jellemzők hogyan függnek össze azzal, hogy a betegek mennyire követik az orvosok javaslatait.

Mivel a gyógytornászok és dietetikusok összefoglalóiban az adherencia jellemzően vélemény formájában fogalmazzák meg (például: *Jól tartaja a diétát – ügyes volt*), ezért megvizsgáltuk, hogy a hagyományos vélemény detekciós módszerek mennyire működnek az adherencia kinyerésére. A szöveg szintű vélemény osztályozás mellett aspektus orientált (Nazir és mtsai, 2020) megközelítéseket is vizsgáltunk, itt a vélemény detekció célja nem a szöveg általános polaritásának a meghatározása, hanem a véleményt kifejezetten egy adott aspektus szemszögéből vizsgáljuk.

A magyar nyelvű vélemény detekció elterjedéséhez elengedhetetlen volt a jó minőségű, kézzel annotált, magyar nyelvű korpuszok megléte. Az első nagyobb adatbázis az OpinHuBank (Miháltz, 2013) volt, ami összesen 10000 példát tartalmazott hírekből és blogbejegyzésekből. Ezekben a példákban a polaritás személyekhez társítva került meghatározásra, ami nagyon hasonlít az aspektus orientált vélemény detekció logikájához. Teljesen aspektus szintű annotációkat tartalmaz a Hungarian sentiment corpus (HuSent) (Szabó és mtsai, 2016), ami 154 termék összehasonlító cikket tartalmaz, összes 17000 mondat hosszban. Kevésbé szerkesztett közösségi médiából származó tartalmakra pedig elérhető a Hungarian Twitter Sentiment Corpus (HTS) (Precognox Ltd., 2017), ami összesen 4000 magyar nyelvű tweet-et és hozzájuk tartozó üzenet szintű polaritás címkét tartalmaz. A címkék 5 fokozatot különböztetnek meg a nagyon pozitívtól a nagyon negatívig. A legújabb nagy méretű vélemény korpusz a HuLU (Ligeti-Nagy és mtsai, 2022) részeként elkészült HuSST, ami a Stanford Sentiment Treebankben (Socher és mtsai, 2013) mondatainak magyarra fordított változatát tartalmazza.

Magyar nyelvű vélemény detekcióra az évek során rengeteg eltérő módszert alkalmaztak a szótár alapú megközeletísektől kezdve (Hangya és Farkas, 2015) a modern előretanított kontextuális beágyazásokra építő mélytanulási megoldásokon (Laki és Yang, 2023; Yang és Laki, 2023) és a generatív nagy nyelvi

modelleken (Yang és mtsai, 2023) át a kiegyensúlyozatlan adatbázison történő adataugmentációig (Üveges és mtsai, 2022).

3. Adherencia a leírásokban

A gyógytornászok és a dietetikusok 1-2 heti rendszerességgel beszélnek telefonon a páciensekkel, és erről jellemzően pár mondatos összefoglalót írnak. Az összefoglalókban általában kitérnek arra, hogy a beteg mennyire követte az egészségügyi alkalmazottak utasításait, mennyire tartotta a diétát, illetve mennyit mozgot a legutóbbi beszélgetés óta. Az összefoglalók adherencia tartalma sokszor emberi szemmel sem egyértelmű, az egyik gyakori ilyen eset, hogy nincs leírva konkrétan, hogy tartotta a diétát, vagy sokat mozgott-e a páciens, csak valamilyen indirekt utalás erre, például egy egyszerű dicséret, hogy "ügyes volt".

A szenzor alapú adherencia vizsgálattal szemben az összefoglalók elemzésének az egyik legnagyobb előnye, hogy részletesebb képet kapunk, ha a betegek az előírt utasításokat valamilyen külső probléma miatt nem tudják betartani vagy nem látszik az adatokban, hogy betartották. Sok esetben a betegek betegség, műtét vagy egyéb körülmények miatt nem tudtak elég aktív életet élni, de a lehetőségeikhez mérten mindent megtettek. Technikai problémák esetén pedig, mint például probléma az internetelérással, eszközök meghibásodása vagy egyéb nehézségek a használatukkal kapcsolatban, nem látszódik az elvégzett munka, pedig a betegek lehet, hogy betartották az előírtakat. A célunk az volt, hogy ezekből az összefoglalókból kinyerjük, hogy mennyire adherensek a betegek.

3.1. Felhasznált adatok

Ahhoz, hogy ezt mérni tudjuk összesen 400 példát annotáltunk kézileg. Az annotáció során 5 kategóriát különböztettünk meg. 3 kategória (alacsony, közepes és magas adherencia) jelölte, hogy ha volt az összefoglalóban adherenciára utaló információ, akkor az alapján milyen volt a beteg adherenciája. Egy külön kategóriát létrehoztunk arra a gyakori esetre, amikor az egészségügyi dolgozók nem tudták elérni a betegeket. A maradék, adherenciára utaló információt nem tartalmazó összefoglalót pedig egy ötödik osztályba soroltunk be. Ezek alapján az adherencia meghatározására az alábbi kategória rendszert alkalmaztuk:

- magas adherencia: a beteg követi az orvos által javasolt feladatokat
- közepes adherencia: a beteg részben követi az orvosi javaslatok, egyes utasításokat betart, másokat nem
- alacsony adherencia: a beteg egyáltalán nem követi az utasításokat
- nem elérhető: a coach nem tudta elérni a beteget
- nincs adherencia információ: a beteg elérhető volt, de a szöveg nem tartalmaz adherenciára utaló információt

Az annotációt két független annotátor végezte, minden összefoglalót mindketten felcímkéztek és ahol nem értettek egyet, ott utólag közösen hoztak döntést. A címkék gyakoriságát az alábbi 1. táblázat tartalmazza:

1. táblázat. Adherencia címkék mennyisége a két halmazon.

Címkék	Validációs	Kiértékelő
Alacsony	22	30
Közepes	25	45
Magas	98	65
Nem elérhető	38	32
Nincs	17	28

4. Módszerek és eredmények

Az adherencia minél pontosabb előrejelzésének érdekében megvizsgáltuk, hogy különböző nyelvfeldolgozási módszertanok mennyire hatékonyan tudják meghatározni egy összefoglaló esetén az adherencia címkét. Ebben a fejezetben összehasonlítjuk a szózsák alapú megoldásokat, különböző vélemény detekciós módszereket és nagy nyelvi modelleket is. A vélemény detekciós modellek és a nagy nyelvi modellek esetén is vizsgáljuk, hogy a döntés aspektusokra bontása tud-e javulást hozni az eredményekben.

4.1. Kísérleti környezet

Az elkészült 400 elemes korpuszt egy 200 elemes validációs és egy 200 elemes kiértékelő halmazra bontottuk. A tanítást tartalmazó kísérletek esetén a tanítást a validációs halmazon végeztük, a prompt alapú módszerek esetén a prompt fejlesztésére és few-shot példákra szintén a validációs halmazt alkalmaztuk. A módszerek teljesítményét minden esetben a kiértékelő halmazon vizsgáltuk.

Más adatbázisokon előretanított megoldások esetén nincs lehetőségünk mind az öt osztály vizsgálatára, így ezekben az esetekben csak az alacsony / közepes / magas adherencia osztályokat különböztetjük meg.

4.2. Hagyományos felügyelt tanulás

Bár az adathalmaz mérete erősen limitálja a hagyományos felügyelt tanulási módszerek lehetőségeit, viszonyítási alapként megnéztük, hogy egy szózsák modell alapú hagyományos osztályozó mit ér el a feladaton. A szózsák modell esetén a szavakat a HuSpaCy keretrendszerrel (Orosz és mtsai, 2023, 2022) lemmatizáltuk és uni-, illetve bigramokat használtunk jellemzőnek. A tanításhoz logisztikus regressziót használtunk. Ezen felül megvizsgáltuk, hogy encoder-alapú nyelvi modellek finomhangolásával mit tudunk elérni, itt HuBERT (Nemeskey, 2020, 2021) modellt finomhangoltunk. A kísérletek eredményeit a 2. táblázat tartalmazza.

A két módszer között nem tapasztalható nagy eltérés, de a limitált tanító adatbázis mellett egyik módszernek sem sikerült hatékonyan elkülöníteni az adherencia osztályokat.

2. táblázat. Hagyományos módszerek eredményeinek összevetése Accuracy és F1-score alapján a kiértékelő halmazon.

Modellnév	Accuracy	F1-score
Szózsák modell	0,505	0,446
HuBERT	0,485	0,424

4.3. Magyar vélemény detekció

A magyar nyelvű vélemény detekciós adatbázisoknak (Miháltz, 2013; Precognox Ltd., 2017) köszönhetően lehetőségünk van arra, hogy kifejezetten vélemény detekcióra tanított modelleket azonnal meg tudjunk hívni egy új feladatra. Ehhez a OpinHuBank (OHB) és Hungarian Twitter Sentiment Corpus (HTS) adatbázisokon tanított HuBERT és XLM-RoBERTa modelleket (Laki és Yang, 2023; Yang és Laki, 2023) vizsgáltunk meg vélemény detekcióra. Az OpinHuBank bár személyneveket alkalmaz a vélemény aspektusaként, az általunk választott modelleknél lehetőségünk van tetszőleges szöveget megadni az aspektusnak. Ezt kihasználva megvizsgáltuk, hogy az OpinHuBank-on tanított modell mennyire alkalmas olyan köznevekkel kapcsolatos vélemény meghatározására, mint a diéta és a mozgás.

Ezek a modellek csak a vélemény polaritásának a kinyerésére alkalmasak, ezért itt csak az alacsony, közepes és magas osztályba tartozó példákat vizsgáltuk a kiértékelés során. Az OpinHuBank 3 címkét tartalmaz, így itt a pozitív - magas adherencia, semleges - közepes adherencia, negatív - alacsony adherencia megfeleltetést alkalmaztuk. Ezzel szemben a Hungarian Twitter Sentiment Corpus 5 címkét tartalmaz, tapasztalataink szerint viszont a validációs halmazon sosem predikálta a nagyon pozitív és a nagyon negatív címkéket, ezért itt is az OpinHuBank-on használt megfeleltetéseket alkalmaztuk.

3. táblázat. Magyar nyelvű előretanított vélemény detekciós modellek eredményei a kiértékelő halmazon.

Modellnév	Accuracy	F1-score
NyTK - HuBERT - HTS	0,621	0,578
NyTK - XLMR - HTS	0,507	0,471
NyTK - HuBERT - OHB	0,514	0,468
NyTK - HuBERT - OHB - diéta	0,514	0,404
NyTK - HuBERT - OHB - mozgás	0,514	0,434
NyTK - XLM-RoBERTa - OHB	0,514	0,481
NyTK - XLM-RoBERTa - OHB - diéta	0,414	0,477
NyTK - XLM-RoBERTa - OHB - mozgás	0,450	0,468

A kísérlet eredményeit a 3. táblázat tartalmazza. A legjobb eredményt a Hungarian Twitter Sentiment Corpus-on tanított HuBERT modell érte el, aminek

az egyik tanulsága lehet, hogy a dietetikusok és a gyógytornászok összefoglalói jobban hasonlítanak a Twitter üzenetekre, mint a híroldalak tartalmára. Ezt a megfigyelést az is erősítő, hogy a 400 példából 5 esetben smiley is találhatók a szövegekben. Ezzel szemben az XLM-RoBERTa alapú megoldások egyik esetben sem voltak hatékonyabbak, mint a HuBERT alapúak és OpinHuBank esetén az aspektusként használt köznevek nem javítottak az eredményeken.

4.4. Angol vélemény detekció

A magyar nyelvű modellek használata mellett lehetőség van angol és más nyelvű adatbázisok (Barbieri és mtsai, 2022) felhasználására is olyan többnyelvű modelleken keresztül, mint az XLM-RoBERTa. Ebben a fejezetben egy többnyelvű (Camacho-Collados és mtsai, 2022; Antypas és mtsai, 2022) (CardiffNLP) és egy angol adaton tanított modellt [1] (Citizenlab) teszteltünk. Mindkét esetben egy XLM-RoBERTa modellt finomhangoltak. A korábbi alfejezethez hasonlóan itt is csak az alacsony / közepes / magas adherencia értékeket vizsgáltuk.

Mindkét modell esetén megvizsgáltuk, hogy az általuk adott értékek mennyire vethetők össze az adherencia kategóriáinkkal. És mindkét modell esetén megvizsgáltuk azt is, hogyha a modelleket a 200 elemű validációs halmazunkon finomhangoljuk, akkor jobb eredményt tudunk-e elérni.

4. táblázat. Nem magyar nyelven tanított modellek előre betanított és finomhangolási eredményei.

Modellnév	Accuracy	F1-score
Citizenlab	0,436	0,380
CardiffNLP	0,529	0,504
Citizenlab - finomhangolt	0,625	0,542
CardiffNLP - finomhangolt	0,655	0,624

Az eredményeket a 4. táblázat tartalmazza. Mindkét modell esetén igaz, hogy a predikcióik elég rossz eredményt értek el, viszont a 200 példán történő finomhangolás után jelentősen javult a teljesítményük.

4.5. Nagy nyelvi modellek

2025-ben az elsőszámú gondolat egy ilyen feladat megoldására a nagy nyelvi modellek alkalmazása. Ezeknél a modelleknél az eredmény nagyban függhet a prompt megfogalmazásán. Mi egy few-shot promptot készítettünk, ahol az instrukciókban a kategóriák definíciója mellett kiemeltük olyan speciális tulajdonságokat, mint, hogy ha a beteg technikai vagy humán problémák miatt – mint a betegség – nem tudott a programmal foglalkozni, vagy naplózni az eredményeit

[1] https://huggingface.co/citizenlab/twitter-xlm-roberta-base-sentiment-finetunned

akkor azt ne vegye negatív adherenciának. A promptok fejlesztéséhez a validációs adathalmazt használtuk, a few-shot példákat is innen válogattuk. Az adatok szenzitivitása miatt nem használhattunk felhő alapú megoldásokat, így a kísérleteinkben három lokálisan futtatható nagy nyelvi modellt alkalmaztunk: Llama 3.3 70B (Grattafiori és mtsai, 2024), gpt-oss 120B (OpenAI, 2025), Qwen3 80B (Team, 2025; Yang és mtsai, 2025).

A nagy nyelvi modellek eredményeit a 5. táblázat tartalmazza. A 3 nyelvi modell alapú megoldás közül a LLama 3.3 70B teljesített a legjobban.

5. táblázat. Nagy nyelvi modellek összehasonlítása mind az 5 címkén és az adherenciára utaló 3 címkén is.

Modellnév	5-címke		3-címke	
	Accuracy	F1	Accuracy	F1
Qwen3 Next 80B A3B Instruct	0,665	0,640	0,679	0,468
Llama3.3 70B	0,710	0,687	0,729	0,512
GPT-oss 120B	0,68	0,659	0,729	0,523

4.6. Nagy nyelvi modellek aspektus alapon

A páciensek adherenciájának vizsgálatához a feljegyzéseket több, előredefiniált aspektus mentén elemeztük. Ezek a tulajdonságok a beteg-orvos kommunikáció különböző dimenzióit ragadják meg, beleértve a beteg elérhetőségét, az életmódbeli változásokat és az adherenciát támogató vagy gyengítő viselkedési mintázatokat. Az egyes kategóriák elkülönített kiértékelése lehetővé teszi, hogy a nagy nyelvi modellek értékelése interpretálható legyen:

- Elérhetőség: Azt vizsgálja, hogy a beteget mennyire sikerült elérni és mennyire volt kommunikált telefonon vagy más médiumon keresztül.
- Utasítás: Megmutatja, hogy a páciens kapott-e utasítást a coach-tól.
- Naplózás: A beteg mennyire vezeti jól a naplóját.
- Dicséret: A coach az üzenetben pozitív vagy negatív jelzőkkel jellemezheti a pácienst (pl. ügyes).
- Testmozgás: Ha szó esett a páciens mozgásáról, akkor megmutatja, hogy ezzel kapcsolatban mennyire követi az utasításokat.
- Súly: Testsúly esetleges változására ad egy pontszámot.
- Étrend: Azt mutatja, hogy a páciens tartja-e a felírt étrendet.
- Kifogás: Megmutatja, hogy amennyiben nincs teljes együttműködés, úgy a páciens részéről találunk-e valami magyarázatot.
- Adherencia: Közvetlenül eldöntjük, hogy a páciens adherens-e.

Az aspektusok felhasználásával interpretálhatóbbá válik az feladat, triviálisabb példák kézzel is osztályozhatók lesznek. Ha az Elérhetőség aspektus negatív, akkor a páciens egyértelműen nem volt elérhető. Hasonlóan hatékony a Súly és

Dicséret aspektus negatív értéke, ebben az esetben biztosan alacsony a páciens adherenciája. Az előző fejezetben a legsikeresebb modellel (Llama3.3 70b) kinyertük az aspektusokat, majd törekedtünk arra, hogy minden aspektus a lehető legkisebb korrelációban legyen egymással. Így 200 példából keresztvalidációval Naive-Bayes módszerrel közelítettük meg a problémát.

6. táblázat. Aspektusok felhasználásával elért eredmények LLM-ek használata során.

Modellnév	5-címke		3-címke	
	Accuracy	F1	Accuracy	F1
Aspekt	0,745	0,592	0,725	0,709
Aspekt+LLM	0,765	0,713	0,765	0,713

Magában ezzel a módszerrel hasonló eredményeket érünk el, mintha közvetlenül nagy nyelvi modellekkel osztályoznánk az adherenciát; azonban egyrészt interpretálhatóbbá válnak a döntéseink. Ezen felül, ha az aspektusokat kombináljuk a korábbi teljesen generatív rendszer eredményeivel további 2 százalékpontos javulás érhető el.

5. Disszkusszió

5.1. Nem egyértelmű esetek

A feljegyzések nem szolgáltak egyértelmű információval az adherencia mértékére vonatkozóan, ami több alkalommal is nehézségeket okozott az annotáció során. Az alábbi két példa jól szemlélteti az ilyen típusú kihívásokat.

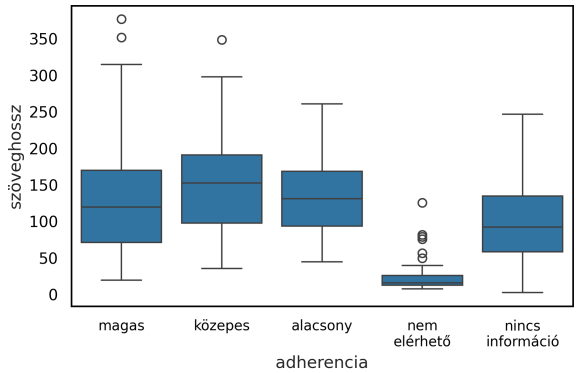
Az egyik tipikus példa a következő bejegyzés:

> *„Aktívkodik napi szinten, ügyes. Testsúlya stagnál, diétát nem tart, és nem is szándékozik. Nem tudok erre mit mondani."*

A szöveg első része a fizikai aktivitás terén mutatott pozitív erőfeszítésre utal. Ugyanakkor a feljegyzés második fele arra utal, hogy a páciens nem követi a diétás ajánlásokat, és nem is mutat hajlandóságot ennek megváltoztatására. Ezeket az eseteket az annotátorok a közepes kategóriába sorolták, viszont ez az ellentmondás a különböző viselkedési területek (aktivitás vs. diéta) eltérő teljesítése miatt nehezen teszi kategorizálhatóvá az esetet a gépi megoldások számára.

A második példa szintén olyan narratív bejegyzést mutat be, amely többértelműsége miatt kihívást jelentett az annotáció során:

> *„Csökken a testsúlya, a kerékpározással küzd (mentálisan), nagyon szenvedős neki. Mondtam, hogy esetleg beszéljünk egyéb mozgásformákról; gondolkodik, hogy mire lenne még lehetőség. Amúgy ügyes, megdicsértem."*



1. ábra: Szöveghossz és az adherencia címkék közötti összefüggés.

A bejegyzés alapján a páciens számára az ajánlott mozgásforma jelentős nehézséget okoz, ugyanakkor a szövegben motivációra utaló jelek is megjelennek: az alternatív mozgásformák keresése és a folyamatos próbálkozás, ezért az annotátorok a magas adherencia osztályba sorolták az esetet. Viszont ez a kettősség — a küzdelem és az együttműködésre való hajlandóság egyidejű jelenléte — ebben az esetben is megnehezítette az adherenciaszint automatikus besorolását.

5.2. Szöveghossz mint jellemző

Az annotált feljegyzések karakterszáma önmagában is informatív jellemzőnek bizonyult. A szövegek hosszának címkénkénti eloszlását a 1. ábra tartalmazza. A "nem elérhető" kategóriába tartozó esetekben nem történik tényleges kommunikáció, így a nagyon rövid bejegyzések döntő többsége ebbe az osztályba sorolható. Emellett a szövegek láthatóan rövidebbek akkor is, ha a feljegyzés nem tartalmaz elegendő információt az adherencia meghatározásához.

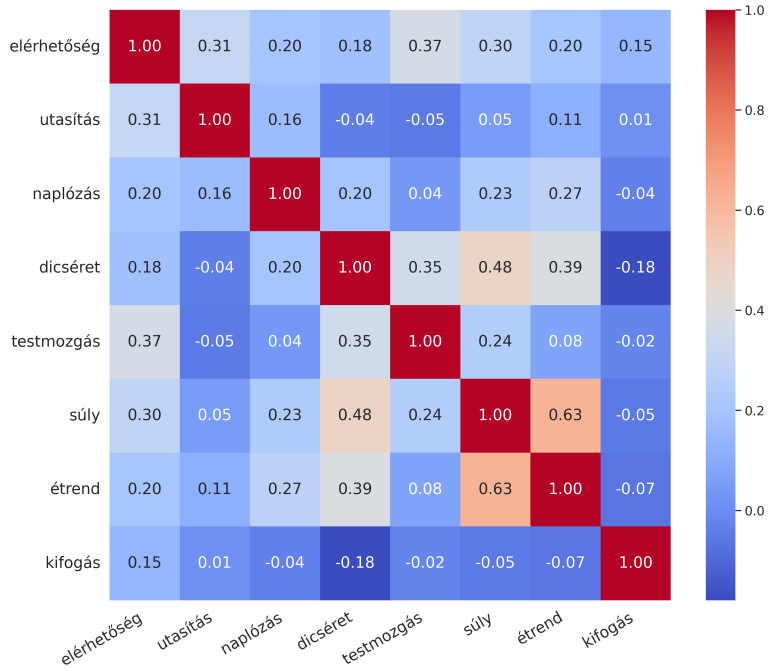
7. táblázat. Aspektusok fontossága Naive Bayes tanításnál.

Aspektus	Fontosság
dicséret	0,1827
elérhetőség	0,1500
naplózás	0,1369
súly	0,1324
testmozgás	0,1292
étrend	0,1053
kifogás	0,0957
utasítás	0,0678

5.3. Aspektusok

Megvizsgáltuk, hogy melyik jellemzőknek volt a legnagyobb fontossága az osztályozás során, ezt a 7. táblázat tartalmazza. Bár a jellemzőknek a fontossága viszonylag kiegyensúlyozott. A dicséret az átlagnál sokkal jobban szeparálja az osztályokat, ezzel szemben a szakértői utasítás kevésbé. A táblázatban az elérhetőség nem tűnik kiemelkedően jelentősnek, hiszen négy címkére semmilyen hatással nincs. Azonban ez az a jellemző, ami teljesen lefedi a nem elérhető adherencia címkét.

Naïve-Bayes osztályozásnál fontos, hogy az aspektusok korrelációja minimális legyen. Az egyes aspektusok korrelációját a 2. ábra tartalmazza. Így megpróbáltuk úgy kiválasztani az aspektusokat és a promptokat, hogy teljesen elkülönüljenek egymástól. Azonban az adherenciához képest még így is jelentős korreláció van a testmozgás, a testsúly és az étrend aspektusai között. Ennek ellenére a modell pontosságán ezek a jellemzők javítottak, mert fontosak az adherencia leírásában.



2. ábra: Aspektusok közötti korreláció.

6. Összegzés

Tanulmányunkban azt vizsgáltuk, hogy a telemedicina programok során keletkező, rövid szakmai összefoglalók alapján milyen mértékben határozható meg a betegek adherenciája. A feladat sajátosságából adódóan az adherencia gyakran implicit módon, véleményszerű megfogalmazásokon keresztül jelenik meg, ezért több hagyományos vélemény detekciós megközelítést és modern nagy nyelvi modelleket hasonlítottunk össze a probléma megoldására.

Összességében a vizsgálat rámutat arra, hogy a nagy nyelvi modellek – különösen aspektusalapú elemzéssel kombinálva – hatékony eszközt jelentenek a telemedicina összefoglalók automatikus feldolgozására. A jövőben ezek az eszközök hozzájárulhatnak ahhoz, hogy az egészségügyi szakemberek gyorsabban és pontosabban azonosítsák azokat a betegeket, akik fokozott támogatást igényelnek az életmódváltás sikeréhez.

Köszönetnyilvánítás

A kutatás támogatói:

A Szegedi Tudományegyetem Interdiszciplináris Kutatásfejlesztési és Innovációs Kiválósági Központ (IKIKK). A szerzők az Értéknövelési

Egészségügyi Modellek Fejlesztési Program, AI által támogatott betegútmenedzsment rendszer fejlesztése program kutatócsoport tagjai.

Az Európai Unió RRF-2.3.1-21-2022-00004 azonosítójú, Mesterséges Intelligencia Nemzeti Laboratórium projektje.

Hivatkozások

Antypas, D., Ushio, A., Camacho-Collados, J., Neves, L., Silva, V., Barbieri, F.: Twitter Topic Classification. In: Proceedings of the 29th International Conference on Computational Linguistics. International Committee on Computational Linguistics, Gyeongju, Republic of Korea (Oct 2022)

Barbieri, F., Espinosa Anke, L., Camacho-Collados, J.: XLM-T: Multilingual language models in Twitter for sentiment analysis and beyond. In: Proceedings of the Thirteenth Language Resources and Evaluation Conference. pp. 258–266. European Language Resources Association, Marseille, France (Jun 2022), https://aclanthology.org/2022.lrec-1.27

Camacho-Collados, J., Rezaee, K., Riahi, T., Ushio, A., Loureiro, D., Antypas, D., Boisson, J., Espinosa Anke, L., Liu, F., Martínez Cámara, E., és mtsai: TweetNLP: Cutting-edge natural language processing for social media. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 38–49. Association for Computational Linguistics, Abu Dhabi, UAE (Dec 2022), https://aclanthology.org/2022.emnlp-demos.5

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., és mtsai: The llama 3 herd of models. arXiv preprint arXiv:2407.21783 (2024)

Hangya, V., Farkas, R.: Doménspecifikus polaritáslexikonok automatikus előállítása magyar nyelvre pp. 15–16 (2015)

Iuga, A.O., McGuire, M.J.: Adherence and health care costs. Risk management and healthcare policy pp. 35–44 (2014)

Kosa, I., Molnar, B., Sepp, K., Pesei, F., Varkonyi, T., Nemes, A., Lengyel, C.: One and a half year's experience with a telemedicine integrated hybrid cardiometabolic rehabilitation service. European Journal of Preventive Cardiology 32(Supplement_1), zwaf236–501 (2025)

Laki, L.J., Yang, Z.G.: Sentiment analysis with neural models for hungarian. Acta Polytechnica Hungarica 20(5), 109–128 (2023), https://acta.uni-obuda.hu/Laki_Yang_134.pdf

Lam, W.Y., Fresco, P.: Medication adherence measures: an overview. BioMed research international 2015(1), 217047 (2015)

Leiz, M., Pfeuffer, N., Rehner, L., Stentzel, U., van den Berg, N.: Telemedicine as a tool to improve medicine adherence in patients with affective disorders–a systematic literature review. Patient preference and adherence pp. 3441–3463 (2022)

Ligeti-Nagy, N., Ferenczi, G., Héja, E., Jelencsik-Mátyus, K., Laki, L.J., Vadász, N., Yang, Z.G., Vadász, T.: Hulu: magyar nyelvű benchmark adatbázis kiépítése a neurális nyelvmodellek kiértékelése céljából. In: XVIII. Magyar Számítógépes Nyelvészeti Konferencia. pp. 431–446 (2022)

Martin, L.R., Williams, S.L., Haskard, K.B., DiMatteo, M.R.: The challenge of patient adherence. Therapeutics and clinical risk management 1(3), 189–199 (2005)

Miháltz, M.: Opinhubank: szabadon hozzáférhető annotált korpusz magyar nyelvű véleményelemzéshez. In: IX. Magyar Számítógépes Nyelvészeti Konferencia. pp. 343–345. Szeged (2013)

Nazir, A., Rao, Y., Wu, L., Sun, L.: Issues and challenges of aspect-based sentiment analysis: A comprehensive survey. IEEE Transactions on Affective Computing 13(2), 845–863 (2020)

Nemeskey, D.M.: Natural Language Processing Methods for Language Modeling. Ph.D.-értekezés, Eötvös Loránd University (2020)

Nemeskey, D.M.: Introducing huBERT. In: XVII. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY2021). p. TBA. Szeged (2021)

OpenAI: gpt-oss-120b and gpt-oss-20b model card (2025), https://arxiv.org/abs/2508.10925

Orosz, G., Szabó, G., Berkecz, P., Szántó, Z., Farkas, R.: Advancing Hungarian Text Processing with HuSpaCy: Efficient and Accurate NLP Pipelines. In: Text, Speech, and Dialogue. pp. 58–69. Springer Nature Switzerland, Cham (2023)

Orosz, G., Szántó, Z., Berkecz, P., Szabó, G., Farkas, R.: HuSpaCy: an industrial-strength Hungarian natural language processing toolkit. In: XVIII. Magyar Számítógépes Nyelvészeti Konferencia. pp. 59–73 (2022)

Pal, P., Sambhakar, S., Dave, V., Paliwal, S.K., Paliwal, S., Sharma, M., Kumar, A., Dhama, N.: A review on emerging smart technological innovations in healthcare sector for increasing patient's medication adherence. Global Health Journal 5(4), 183–189 (2021)

Precognox Ltd.: Hungarian twitter sentiment corpus (2017), https://opendata.hu/dataset/hungarian-twitter-sentiment-corpus

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C.D., Ng, A., Potts, C.: Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. pp. 1631–1642. Association for Computational Linguistics, Seattle, Washington, USA (oct 2013), https://www.aclweb.org/anthology/D13-1170

Szabó, M.K., Vincze, V., Simkó, K.I., Varga, V., Hangya, V.: A Hungarian sentiment corpus manually annotated at aspect level. In: Calzolari, N., Choukri, K., Declerck, T., Goggi, S., Grobelnik, M., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S. (szerk.) Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16). pp. 2873–2878. European Language Resources Association (ELRA), Portorož, Slovenia (May 2016), https://aclanthology.org/L16-1459/

Szalka, B., Vassanyi, I., Máthéné Köteles, É., Szabo, L.A., Lada, S., Bolgar, T., Korom, A., Abraham, J., Bilicki, V., Barnai, M., és mtsai: Factors of weight loss for telemedically supported metabolic syndrome patients in a controlled trial. Applied Sciences 14(22), 10179 (2024)

Team, Q.: Qwen3 technical report (2025), https://arxiv.org/abs/2505.09388

Üveges, I., Csányi, G.M., Ring, O., Orosz, T.: Szövegaugmentálási módszerek összehasonlítása politikai szövegek szentimentanalízise során pp. 521–534 (2022)

Yang, A., Yu, B., Li, C., Liu, D., Huang, F., Huang, H., Jiang, J., Tu, J., Zhang, J., Zhou, J., Lin, J., Dang, K., Yang, K., Yu, L., Li, M., Sun, M., Zhu, Q., Men, R., He, T., Xu, W., Yin, W., Yu, W., Qiu, X., Ren, X., Yang, X., Li, Y., Xu, Z., Zhang, Z.: Qwen2.5-1m technical report. arXiv preprint arXiv:2501.15383 (2025)

Yang, Z.G., Dodé, R., Ferenczi, G., Héja, E., Jelencsik-Mátyus, K., Kőrös, Á., Laki, L.J., Ligeti-Nagy, N., Vadász, N., Váradi, T.: Jönnek a nagyok! bert-large, gpt-2 és gpt-3 nyelvmodellek magyar nyelvre. In: XIX. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2023). pp. 247–262. Szegedi Tudományegyetem, Informatikai Intézet, Szeged, Hungary (2023)

Yang, Z.G., Laki, L.J.: Neurális entitásorientált szentimentelemző alkalmazás magyar nyelvre. In: XIX. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2023). pp. 107–117. Szegedi Tudományegyetem, Informatikai Intézet, Szeged, Hungary (2023)

Szerzői index, névmutató