Jönnek a nagyok! BERT-Large, GPT-2 és GPT-3 nyelvmodellek magyar nyelvre

Yang Zijian Győző, Dodé Réka, Ferenczi Gergő, Héja Enikő, Jelencsik-Mátyus Kinga, Kőrös Ádám, Laki László János, Ligeti-Nagy Noémi, Vadász Noémi, Váradi Tamás

Nyelvtudományi Kutatóközpont
1068 Budapest, Benczúr u. 33.
{vezetéknév.keresztnév}@nytud.hu

Kivonat Az utóbbi években rendkívüli mértékben felgyorsult a Transformer alapú nyelvmodellek méretének a növekedése. A globális technológiai cégek nagyobbnál nagyobb modelleket tanítanak, amelyek óriási erőforrást és tanítóanyagot igényelnek. Ezekkel a kísérletekkel azt próbálják bebizonyítani, hogy megfelelően nagy méretű modellek, megfelelően sok tanítóanyaggal képesek önmagukban akár finomhangolás nélkül bármilyen nyelvtechnológiai feladatot megoldani. Ebbe a versenybe nem igazán lehetséges beszállni, de arra van lehetőség, hogy az árnyékukban elkezdjünk kísérleteket végezni a nagyobb méretű modellek irányában. Kutatásunkban különböző nagy méretű nyelvmodelleket tanítottunk magyar nyelvre. Betanítottunk egy 6,7 milliárd paraméteres GPT-3, valamint egy GPT-2 és egy BERT-Large modellt magyar nyelvre. A modelleket különböző finomhangolással teszteltük. A BERT-Large modellünk több feladatban is felülmúlta a huBERT modellt, és elsőként hoztunk létre egynyelvű magyar GPT-3 modellt, amelyekkel tudomásunk szerint elsőnek végeztünk prompt kísérleteket *few-shot* tanulással magyar nyelvre.
Kulcsszavak: GPT-3, GPT-2, Megatron BERT, prompt programozás, *few-shot* tanulás

1. Bevezetés

Az utóbbi időkben minden évben jön ki egy újabb nyelvmodell, a nagy kutatóközpontok és cégek versenyt űznek abból, hogy nagyobbnál nagyobb méretű és paraméterszámú nyelvmodelleket tanítsanak. Amikor 2021-ben a Microsoft az NVIDIA-val karöltve létrehozta az 530 milliárd paraméteres Megatron-Turing NLG modellt (Smith és mtsai, 2022), megszületett a cikk[1], amelyik felteszi a kérdést, hogy vajon ez a verseny lenne az új Moore-törvény? Ezek a kutatások

[1] https://www.microsoft.com/en-us/research/blog/using-deepspeed-and-megatron-to-train-megatron-turing-nlg-530b-the-worlds-largest-and-most-powerful-generative-language-model

azt próbálják bebizonyítani, hogy ha elég nagy a modell és több adaton tanult, akkor képes ez az egy nagy modell minden nyelvtechnológiai feladatot megoldani finomhangolás nélkül, csupán prompt programozással. Ehhez a versenyhez óriási erőforrás szükséges, amit csak a legnagyobb cégek és kutatóközpontok engedhetnek meg maguknak.

Jelenleg magyar nyelvre a legjobban teljesítő nyelvmodell a huBERT (Nemeskey, 2021). A HILBERT (Feldmann és mtsai, 2021) teljesítménye, bár méretét tekintve a huBERT-nél nagyobb modell, az eddig rendelkezésre álló teszteken elmaradt a huBERT-étől; ennek oka az, hogy a huBERT-hez képest kevesebb adaton lett betanítva. 2022 júniusában mutatta be a HILANCO[2] konzorcium a HILANCO-GPTX 6,7 milliárd paraméteres angol-magyar kétnyelvű GPT-3 modellt.

Jelen cikkben a huBERT és HILBERT modelleknél nagyobb nyelvmodellek készítéséről számolunk be. A modellek tanításához nagy mennyiségű adatot is gyűjtöttünk. A jelen tanulmányban bemutatjuk az általunk összegyűjtött tanítóanyagot, majd bemutatjuk az ebből tanított modelljeinket. Nevezetesen betanítottunk egy HILANCO-GPTX modellhez hasonló 6,7 milliárd paraméteres GPT-3, egy GPT-2 és egy BERT-Large modellt magyar nyelvre, melyeket PULI névre kereszteltük: **PULI GPT-3SX**, **PULI GPT-2** és **PULI BERT-Large**. Mindhárom modellünk kutatás céljára szabadon elérhető a Hugging Face oldalunkonk[3]: NYTK/PULI-GPT-3SX, NYTK/PULI-GPT-2, NYTK/PULI-BERT-Large

2. Kapcsolódó irodalom

Jelenleg a világ egyik legnagyobb modellje a PaLM (Chowdhery és mtsai, 2022) (Pathways Language Model) a Google-tól, amely 540 millárd paraméteres. A méretek növelése mellett bevezették a Pathways architektúrát, amely azt a célt szolgálja, hogy egyszerre minél több feladatot tudjon tanulni a modell. A Pathway egy hagyományos, csak dekóderrel rendelkező transzformer (Vaswani és mtsai, 2017) architektúrát valósít meg néhány módosítással. Módosításaihoz az utóbbi évek fejlesztéseiből merít, úgy mint a SwiGLU activációs függvény (Shazeer, 2020), a párhuzamos rétegezés (Wang és Komatsuzaki, 2021) a transzformer blokkokban, a RoPE beágyazás (Su és mtsai, 2021) vagy a SentencePiece (Kudo és Richardson, 2018) használata. A PaLM modellhez képest méretben egy kicsivel lemaradva, komoly konkurens a már a bevezetésben is említett Megatron-Turing NLG modell (Smith és mtsai, 2022). Ebben a nagyságrendben elsikkad a figyelmünk a 10 milliárd paraméter különbség felett, de például magyar nyelvre még nem sikerült senkinek 10 milliárd paraméteres modellt tanítani. Paraméterszámát tekintve így is háromszor akkora, mint a mérföldkőnek számító GPT-3. A GPT-3 (Brown és mtsai, 2020) megjelenése óriási visszhangot váltott ki mind a sajtóban, mind a nyelvtechnológiai szakmai közösség körében. Újdonsága abban rejlett, hogy óriási mennyiségű adattal tanították, és az akkori viszonylat-

[2] https://hilanco.github.io
[3] https://huggingface.co/NYTK

ban óriási paraméterszámmal. A modellel olyan szöveget tudtak generálni, ami az emberi íráshoz volt hasonló. Emellett finomhangolás nélkül, az úgynevezett prompt programozással kevés (*few-shot*) bemeneti példával, vagy egyáltalán nem adva példát (*zero-shot*), meg tud oldani nyelvtechnológiai feladatokat. A GPT-3 több változata is elérhető kipróbálásra, mint a Davinci, Curie, Babbage vagy az Ada. Mindegyik változat más-más feladattípusban erős. Az eddig felsorolt modellek mellett érdemes megemlíteni a Wu Dao 2.0[4] modellt. A Wu Dao 2.0 modellt a Beijing Academy of Artificial Intelligence (BAAI) szervezet mutatta be 2021-ben. Ez jelenleg a legnagyobb neurális modell, amely 1750 milliárd paraméterrel rendelkezik. A modell összehasonlítása a többi nyelvmodellel annyiban nehéz, hogy nem csak szövegeken, hanem képeken is tanították. A modellt a Pile angol adathalmaz (Gao és mtsai, 2020) mellett további 1,2TB kínai szövegen és 2,5TB képen tanították. A modellt a FastMoE (He és mtsai, 2021) rendszerben tanították. Több feladatban is 'state-of-the-art' eredményt ért el. Az elmúlt években egymás után mutatták be a modelleket. De érdemes megemlíteni még a Megatron-Turing NLG elődjeit, a 17,2 milliárd paraméteres Turing-NLG-t[5] vagy a 8,3 milliárd paraméteres Megatron-LM (Shoeybi és mtsai, 2019a) modelleket. Ha milliárd paraméterszámról beszélünk, akkor a T5 (Raffel és mtsai, 2020) XXL modellje 13 milliárd, a GPT-2 (Radford és mtsai, 2019) pedig 1,5 milliárd paraméterével éppen belefértek még ebbe az összehasonlításba. Bár paraméterszámban már nagyságrenddel kisebb, de az utolsó említendő modell a BERT-Large (és általában a BERT család), amely a maga már szerénynek mondható 340 millió paraméterével a transzformer alapú nyelvmodellezés alapjait fektette le.

3. A tanítóanyag

Kutatásunk első része a tanítóanyag összeállítása volt. A korábbi kutatások alapján a nagy modellek tanításához nagy mennyiségű adatra is szükség van. Ehhez a feladathoz az alábbi forrásokból származó korpuszokat használtuk fel.

Az 1. táblázatban láthatóak az alkorpuszok főbb tulajdonságai. Nem tokenizáltuk a szöveget, a számokat a nyers szövegre mértük. A modellek tanításához később sem tokenizáltuk a szöveget. A korpuszban egy sor egy bekezdés, a dokumentumokat üres sorok választják el, nem bontottuk a szöveget mondatokra. A korpusz szövegei a következő forrásokból áll össze:

- **Webkorpusz 2.0**: A Webkorpusz 2.0 (Nemeskey, 2020b) korpuszt Nemeskey Dávid Márk gyűjtötte a Common Crawl[6] adattárból 2013 és 2019 április közötti időszakból. A korpusz több mint 9 milliárd tokenből áll. A kutatásunkhoz a nem tokenizált változatot használtuk fel.
- **Wikipédia**: A magyar Wikipédia, a Webkorpusz 2.0 része.

[4] https://towardsdatascience.com/gpt-3-scared-you-meet-wu-dao-2-0-a-monster-of-1-75-trillion-parameters-832cd83db484

[5] https://www.microsoft.com/en-us/research/blog/turing-nlg-a-17-billion-parameter-language-model-by-microsoft

[6] https://commoncrawl.org

- **CC**: A szöveg legnagyobb részét a Common Crawl adattárból gyűjtöttük. Mivel a WebKorpusz 2.0 a 2019 áprilisig gyűjtött anyagot tartalmazza, a kutatásunkban az ezután létrejött adatokat gyűjtöttük össze. Az adatok letöltéséhez és *boilerplate*-tisztításához az Indig (2018) által módosított CC letöltő szkriptet[7] alkalmaztuk. A felhasznált szkript már tartalmaz különböző tisztítási és deduplikációs eljárásokat. Ennek ellenére, a különböző évek gyűjtései között lehetnek még duplikációk, ezeket a folyamat végén kezeltük. Fontos megjegyezni, hogy mind a Webkorpusz 2.0, mind a saját CC gyűjtés csak a '.hu' doménről származó szövegeket tartalmazza.
- **neticle**: A nyilvános közösségi média posztokból és kommentekből származó szöveganyag, melyet a Neticle Kft-től kaptunk meg korábban.
- **sajtó**: Online sajtóoldalakról letöltött tartalmak 2020 és 2021 közötti időszakból.
- **JSI**: A szlovén Jožef Stefan Institute az eventregistry.org címen futó webszolgáltatás céljaira 2013 óta számos nyelven gyűjti a híreket internetes forrásokból (RSS-ből). Ennek a magyar anyagát használtuk fel.
- **araneum**: Araneum Hungaricum Maium[8](Benko, 2014a,b; Rychlý, 2007) korpusz, amelyet Vladimir Benko állított össze.
- **hutenten**: A huTenTen korpusz a Lexical Computing Llc. által kifejlesztett TenTen korpuszcsalád (Jakubíček és mtsai, 2013) része és a SketchEngine platform magyar referenciakorpusza. A korpuszt a Lexical Computing LLC állította össze egy 2013-ban végzett gyűjtés (Suchomel és Pomikálek, 2012) alapján, magyar nyelvi elemzését az MNSZ1 (Oravecz és mtsai, 2014) kódjaival Oravecz Csaba, az emMorph (Novák és mtsai, 2016) kódjaival a Lexical Computing Llc. végezte.
- **hírportálok**: Korábbi kutatásainkhoz, főleg szöveg-összefoglaló generálás feladatához összegyűjtött adathalmaz. Különböző hírportálokról gyűjtött cikkek és azok leadjei. A források: index.hu; nol.hu; HVG. Természetesen lehetnek átfedések a CC-ben gyűjtött adatokkal, a duplikációkat a folyamat végén kezeltük.

A gyűjtés végén összekonkatenáltuk az összes forrásból összegyűjtött szövegeket, majd dokumentumszintű `jsonline` formátumra alakítottuk át, ahol egy sor egy json objektum, benne egy `text` mezővel, amiben egy dokumentum szövegei találhatóak, megőrízve a sortöréseket. Ezen a json fájlon végeztünk *dokumentumszintű* deduplikációt (`uniq`) és véletlenszerű keverést. Az 1. táblázatban az utolsó két sorban az összegzett értékeket látjuk, ahol az első sor az összeadott értéket, míg az utolsó sor a deduplikált végső korpusz értékeit mutatja. Ha összevetjük a csak simán összeadott számokat a deduplikált számokkal, akkor azt láthatjuk, hogy van körülbelül 4 millárd szó eltérés. Ez azt jelenti, hogy főleg a CC éves gyűjtései között nem kevés duplikáció szerepel.

Az adatok gyűjtése több lépésben történt. Először a Webkorpusz 2.0 korpuszból és a Common Crawl gyűjtésekből állítottuk össze az első nagy korpuszunkat. Ily módon első körben létrejött egy körülbelül 25 milliárd szavas korpusz. Ez

[7] https://github.com/DavidNemeskey/cc_corpus
[8] http://ucts.uniba.sk/aranea_about/_hungaricum.html

	Dokumentum	Bekezdés	Szó
Webkorpusz 2.0	9 240 709	171 239 297	8 051 677 190
Wikipédia	418 622	6 804 115	124 982 493
CC 2019	8 259 348	199 368 999	5 994 324 578
CC 2020	7 374 175	174 726 213	5 289 809 348
CC 2021	10 681 529	254 848 762	7 702 038 666
CC 2022	2 586 953	61 817 892	1 874 763 279
neticle	30 471 970	85 351 213	1 112 740 383
sajtó	1 682 151	4 103 234	625 098 614
jsi	4 023 083	32 363 186	1 077 066 597
araneum	3 727 984	31 721 824	1 329 200 470
hutenten	6 447 787	164 654 976	2 670 682 031
híportálok	1 326 922	8 503 669	433 558 050
Összesen	86 241 233	1 195 503 380	36 285 941 699
Összesen (uniq)	78 059 419	1 069 655 352	32 425 610 652

1. táblázat. A tanítókorpusz összetétele

látható az 1. táblázat első kettő blokkjában. Ezzel a korpusszal elkezdtünk különböző kísérletek végezni. Ebben a fázisban jöttek létre a későbbi modellek tanításához használt szótárak (*vocab*) is. Összesen kettő szótárat hoztunk létre:

- **BERT szótár**: Szóelem (WordPiece) szótár; méret: 32 203. Néhány specális karaktert adtunk hozzá manuálisan, valamint a szótár végére manuálisan hozzáadtunk 203 darab tokent, amelyet a huBERT szótárából (utolsó 203 elem) emeltünk át. Erre azért volt szükség, mert korábbi kutatásainkban észleltük azt a jelenséget, hogy ha a tokenizálás folyamatába a hagyományos tokenizáláshoz magyar tokenizáló eszközt használunk, mint például a quntoken (Mittelholcz, 2017) eszközt, akkor bizonyos esetekben az írásjel rajtamarad a szón (ami helyes), amit nem tud kezelni az eredeti szótár, mivel az eredeti BERT tokenizáló minden írásjelet leválaszt a szóról. Ennek az a következménye, hogy mivel nem szerepel az írásjel szóelem részeként (csak külön szóelemként), ismeretlen (`[UNK]`) szóelemként kezeli őket.
- **GPT szótár**: Byte-Pair-Encoding (BPE) szótár; méret: 50 000. Néhány specális karaktert adtunk hozzá manuálisan.

A 2. táblázatban láthatóak a tokenizálók összehasonlítása. Referenciaként a huBERT tokenizálóját adtuk meg. Az összehasonlíthatóság végett a Nemeskey (2021) tanulmányban megadott példákat használtuk. Látható, hogy a mi BERT szótárunk nagyon hasonlít a huBERT szótárához. Majdnem minden szó tokenizálása megegyezik, kivétel az "Andersen" szó, ahol a huBERT egybetartotta, míg a mi szótárunk szétszedte. A GPT szótárunk viszont már "töredezettebb" a BERT szótárakhoz képest.

A szótárak elkészítése után elkezdtük tesztelni a tanítószkripteket, kevés lépéssel sikerült betanítani modelleket. Miután sikerült a tanítás, következő lépésként további adatokat adtunk hozzá a korpuszhoz (lásd 1. táblázat harmadik blokk). Így jött létre a 32,4 milliárd szavas korpuszunk, amellyel végül beta-

	huBERT	BERT szótár	GPT szótár
Nemzeti	Nemzeti	Nemzeti	Nemzeti
Andersen	Andersen	And ers e n	A nder sen
labdarúgó	labdarúgó	labdarúgó	lab darúgó
zambiai	z amb iai	z amb iai	z amb iai
megmaradt	megmaradt	megmarad t	meg maradt
hétfő	hétfő	hétfő	hétfő
keddtől	kedd től	kedd től	ked d től
edényben	edény ben	edény ben	edény ben
Hétfőn	Hétfőn	Hétfőn	Hétfőn
tájékoztatták	tájékoztat ták	tájékoztat ták	tájékozt atták
leggazdagabb	leggazdagabb	leggazdagabb	leg gazdagabb
elpártolt	el párt olt	el párt olt	elp árt olt

2. táblázat. Tokenizálók összehasonlítása

nítottuk a modelljeinket. A szövegek egy részét normalizálni kellett. Bizonyos szövegek nem utf-8 kódolásúak voltak, ezeket kellett utf-8 formátumba konvertálni. Továbbá a *sajtó* szövegein metaadat szűrést kellett végrehajtani, mivel nyelvi annotációkat is tartalmazott a szöveg. A *neticle* szövegei esetén pedig bekezdésszintű deduplikációt is végre kellett hajtani.

4. A modellek bemutatása és előtanítása

Kutatásunkban három modellt tanítottunk: Megatron BERT, Megatron GPT-2 és a GPT-NeoX családhoz tartozó 6,7 milliárd paraméteres GPT-3 modellt.

A **PULI BERT-Large** a Megatron BERT (Shoeybi és mtsai, 2019b) magyar változata. A Megatron BERT a Megatron-DeepSpeed projekt része, ami az NVIDIA Megatron-LM modellek tanítását támogatja DeepSpeed technológiával. A projekt tartalmaz egy BERT-Large méretnek megfelelő (345 millió paraméter) BERT nyelvmodell előtanítási implementációt. Kutatásunkban azt a szkriptet használtuk, amelyik egyetlen GPU-n végzi a tanítást. Így a modelltanításhoz egyetlen NVIDIA A100 (80GB) GPU-t használtunk. A tanítás során a legtöbb hiperparaméteren nem változtattunk, kivéve:

– `split`: 994,5,1; kikapcsoltuk az fp16 kapcsolót; `micro-batch-size`: 40; `global-batch-size`: 320.

A vágáson (`split`) azért módosítottunk, hogy több anyagot használjunk fel a tanításhoz. A micro-batch-size méretet empirikusan választottuk ki, ennyi fért bele a 80 GB-os GPU memóriába. A `global-batch-size` méret választása esetében 8 darab GPU használatát szimuláltuk. A tanítást 750 000 lépésnél állítottuk le, ami körülbelül 2 és fél epoch környékén van. Az eredeti BERT modellek tanítása 256 batch méreten történt (Devlin és mtsai, 2019) körülbelül 40 epoch a 3,3 milliárd szavas korpuszon, ami átszámítva a mi méretünkre körülbelül 4 epochnak felel meg. A végső veszteségi értékek (*loss*):

– Nyelvmodell (*language model* - lm) validációs veszteség (`lm loss value`): 2,32; perplexitás (`lm loss PPL`): 10,19.
– Mondatsorrend predikció (*sentence order prediction* - sop) validációs veszteség (`sop loss value`): 0,14; perplexitás (`sop loss ppl`): 1,15.

A tanítási idő körülbelül két hónap volt. Végül az elkészült modellt a Hugging Face által közzétett szkripttel[9] konvertáltuk a Hugging Face által használt formátumra.

A **PULI GPT-2** a Megatron GPT-2 magyar változata. A Megatron GPT-2 szintén a Megatron-DeepSpeed projekt része, amely egy 345 millió paraméteres GPT-2 típusú modell. Hasonlóan a BERT modell tanításához, az egy GPU-s implementációt választottuk, vagyis egy NVIDIA A100 (80GB) GPU-t használtunk. A hiperparaméter beállításakor hasonlóan jártunk el, mint a BERT tanítása esetén:

– `split`: 994,5,1; kikapcsoltuk az fp16 kapcsolót; `micro-batch-size`: 13; `global-batch-size`: 52.

A tanítást 500 000 lépésnél állítottuk le, ami körülbelül 1 epoch. A végső veszteségi értékek (*loss*):

– Nyelvmodell validációs veszteség (`lm loss value`): 2,80; perplexitás (`lm loss ppl`): 16,46.

A tanítási idő 20 nap volt. Az elkészült modellt a Hugging Face által közzétett szkripttel[10] konvertáltuk Hugging Face által használt formátumra. A konvertáló szkriptben módosítani kellett a `vocab_size`, `bos_token_id` és az `eos_token_id` paramétereket, mivel saját szótárat használtunk.

A **PULI GPT-3SX** a GPT-NeoX (Andonian és mtsai, 2021; Black és mtsai, 2022) magyar változata. A GPT-NeoX az EleutherAI[11] projektje. A projekt célja, hogy nagy paraméterszámú nyelvmodelleket lehessen tanítani, mint a GPT-3. Egyik konkrét céljuk, hogy reprodukálják a GPT-3 modellt. Implementációjuk az NVIDIA Megatron-LM és a DeepSpeed technológián alapszik. Többféle GPT-3 jellegű konfigurációt állítottak össze, kicsi modellektől (pl.: 160 millió paraméter) a közepes méretűeken át (pl.: 20 milliárd paraméter) a nagy modellekig (175 milliárd paraméter). Kutatásunkban egy viszonylag kicsinek számító 6,7 milliárd paraméteres konfigurációt használtunk. A tanításhoz 8 darab NVIDIA A100 (80GB) GPU-t használtunk. A betanítást a hiperparaméterek nagy részének a módosítása nélkül végeztük. Egyedül a *batch* méretet módosítottuk, amely empirikus módon (ennyi fér bele a 80 GB-os GPU-ba) a 16-ra állítottunk. A tanítást 150 000 lépésnél állítottuk meg, ami számításunk alapán egy kicsivel több mint 1 epoch. A tanítási idő 21 nap volt. A végső veszteségi értékek (*loss*):

[9] https://github.com/huggingface/transformers/tree/main/src/transformers/models/megatron_bert
[10] https://github.com/huggingface/transformers/tree/main/src/transformers/models/megatron_gpt2
[11] https://www.eleuther.ai

– Nyelvmodell validációs veszteség (`lm loss value`): 2,17; perplexitás (`lm loss ppl`): 8,76.

A jövőben szeretnénk még további kísérleteket végezni a modellek tovább tanításával.

5. A modellek finomhangolása

A modellek kiértékeléséhez többféle módszert is választottunk. A Megatron BERT és a GPT-2 esetében a finomhangolás módszerével teszteltük. Ehhez először, az összehasonlíthatóság végett, a magyar szakirodalomban (Nemeskey, 2020a; Yang és Váradi, 2021; Yang Zijian Győző, 2022; Yang Zijian Győző, 2022) gyakran használt három tokenszintű osztályozó és kettő mondatszintű osztályozó korpuszt használtunk fel: NYTK-NerKor (NerKor) (Simon és Vadász, 2021), Szeged NER (SzNer) (Szarvas és mtsai, 2006), Szeged Treebank (NP) (Csendes és mtsai, 2005) és Magyar Twitter Szentiment korpuszt (Szabó és mtsai, 2016). A szentiment korpusz esetében a Laki és Yang (2021) által készített két alkorpuszt használtuk fel: 2 osztályos (2-o) és 5 osztályos (5-o). A kiértékelési metrikának az F1 és az abszolút pontosságot választottuk (accuracy - acc). A finomhangoláshoz a Hugging Face által közzétett függvényeket használtuk, ugyanazokat és ugyanolyan beállításokkal, mint Yang és Váradi (2021) a kutatásukban, annyi különbséggel, hogy a mi esetünkben 4 darab NVIDIA A100 GPU állt a rendelkezésünkre.

Következő lépésként a 2022-ben megjelent magyar benchmark korpuszon, a HuLU (Ligeti-Nagy és mtsai, 2022) korpuszon értékeltük ki. Kiértékelési metrikának az abszolút pontosságot választottuk (accuracy - acc), kivétel a HuRC, ahol a pontos egyezés és az F1 metrikákat. Időközben megjelent a HuLu hivatalos kiértékelő oldala[12], ahol némelyik korpusznál az MCC (Matthew's correlation coefficient) (Matthews, 1975) metrikát használják, így az itt elért MCC eredményeket is beillesztettük a tanulmányba. A HuCOLA, HuSST, HuWNLI és HuRTE feladatok finomhangolásához a Hugging Face szövegosztályozó szkriptjét használtuk (ugyanazt, mint amit a szentiment analízis modell finomhangolásához használtuk). A HuRC esetében a Hugging Face kérdés-válasz (question answering) szkriptjét[13], míg a HuCoPA feladathoz a többválasztós (multiple choice) szkriptet[14]. Minden finomhangoláshoz 4 darab NVIDIA A100 GPU állt a rendelkezésünkre.

A HuLU korpuszokon való finomhangolás részletei egy külön tanulmány témáját jelentik. Minden modell és feladat más-más hiperparaméter-beállításokat igényelt. Továbbá a HuWNLI, a HuCoPA, a HuRTE és a HuRC korpuszok esetében is különböző bemeneti adatstruktúrákkal, úgynevezett prompttal kísérleteztünk, amelyek hatással voltak a teljesítményre. A két fő kísérleti irány, hogy

[12] https://hulu.nytud.hu

[13] https://github.com/huggingface/transformers/tree/main/examples/pytorch/question-answering

[14] https://github.com/huggingface/transformers/tree/main/examples/pytorch/multiple-choice

a vizsgálandó szövegeket szeparátor címkével (SEP) válasszuk el, vagy egy egységes folyószöveggé (Szöveg) alakítsuk át őket. Egy-egy példa a két típusra:

- Eredeti:
  - *premise*: A sofőr felkapcsolta az autó fényszóróit.
  - *choice1*: Mennydörgést hallott.
  - *question*: cause
- SEP: A sofőr felkapcsolta az autó fényszóróit. [SEP] Mennydörgést hallott.
- Szöveg: A sofőr felkapcsolta az autó fényszóróit. Mert mennydörgést hallott.

A fenti példában a szeparátor címke lehet akár `[CLS]` vagy a GPT modelleknél `</s>` stb. A folyószöveggé alakítás során a fenti példában két mondatot látunk, de lehet egy mondatba is foglalni őket vesszővel elválasztva stb. Minden modellnél a különböző prompt különböző hatással van a teljesítményre. A GPT-3 modell esetében különösen nagy jelentősége van a megfelelő promptok kiválasztásában.

A legnehezebb feladat a GPT-3 modell kiértékelése volt. Mivel a GPT-3 finomhangolása nagy erőforrást igényelt volna, első körben prompt programozással (Reynolds és McDonell, 2021) végeztünk kísérleteket. A prompt programozás azt jelenti, hogy nincsen finomhangolás, csupán, a predikció során, a bemeneti adat manipulálásával érjük el, hogy a modell a kívánt kimenetet generálja. Az adott feladathoz a megfelelő prompt választása nem könnyű feladat, amelyre külön kutatások vannak (Shin és mtsai, 2020; Alivanistos és mtsai, 2022). A promptok előállítása történhet manuálisan és automatikus módszerekkel. A kutatások alapján az egyes nyelvmodellek eltérő módon reagálnak a különböző promptkészletekre, sőt a különböző feladatokra is (ahogy fent láthattuk). A promptok szerkezeti felépítése és a megfelelő promptok kiválasztása nagyban meghatározza a modell teljesítményét. A jövőben szeretnénk ennek egy külön kutatást szentelni. A mostani kutatásban manuálisan hoztuk létre a promptokat és empirikus módon állítottuk a paramétereket. Többféle prompttal és beállítással kísérleteztünk, végül a legjobbnak tűnő beállítást tartottuk meg. A generált válasz manipulálásában a következő paraméterekkel tudtunk kísérletezni: bemeneti prompt szöveg; hőmérséklet; top-k; top-p és kimeneti szöveg hossza. A kimeneti szöveg hossza minden esetben 3 volt megadva, hiszen mindig csak egy-egy osztálycímke, vagy egy-egy szót vártunk kimenetnek. Azért adtunk meg 1-nél nagyobb értéket, mivel a modell időnként sortöréssel kezdte a szöveget, illetve hogy tudjunk olyan kiemeneti szóval kísérletezni, amit esetleg több szóra darabolhat a tokenizáló. Ahhoz, hogy csökkentsük a modell 'kreativitását' a top-k, top-p és hőmérséklet értékét is alacsonyra kellett állítani. Első tapasztalataink alapján a top-k értékét 10 körüli értékre állítottuk, a top-p értékét 0,1-0,4 közé, a hőmérsékletet pedig 0,1-0,4 környékére. A prompt esetében vettük a tanítóanyag első X szegmensét, ahol minden szegmenshez egy szeparátor karakterrel elválasztva hozzákonkatenáltuk a kimeneti osztálycímkét. Az osztálycímkék esetében kipróbáltuk a szám (0;1) és különböző szöveges (igen; nem; pozitív; negatív; jó; rossz stb.) változatot. Az X kiválasztásánál az volt az elsődleges szempont, hogy lehetőleg lásson minden osztálycímkéből több példát is. Túl sok példát a bemeneti hossz korlátja miatt nem is tudtunk volna megadni. Tapasztalataink alapján 30-nál több példánál nem nagyon tudott jól osztályozni a modell. A modell az utolsó néhány

példára hajlamos jobban odafigyelni, ezért nem mindegy, hogy mi az utolsó példa amit lát. A 3. táblázatban látható egy példa egy lehetséges promptra. A dőlt betűvel szedett sor a tesztelni kívánt mondat. A modell a megadott szeparátor karakterre folytatja a szöveg generálását. Mivel 3 a kimeneti hossz értéke, ezért még folytatja a generálást a modell, így mi csak a generált szöveg első szavát vesszük figyelembe. Ahogy a példában látszik, egy sortörést és egy *A* betűt generált még a modell.

Hatásokkal teli, de túl langyos filmbiográfia. = negatív
Ha szeretsz időnként moziba menni, érdemes a Wasabi-val kezdeni. = pozitív
A szórakoztatás és az oktatás ritka kombinációját kínálja. = pozitív
Azon töpreng, hogy miért van szükségünk annyira a történetekre. = semleges
Szellemtelen és teljesen értelmetlen. =
Kimenet: negatív [sortörés] A

3. táblázat. Példa egy promptra

A következő beállításokat alkalmaztuk a különböző feladatokban:

- HuCOLA: prompt #: 25; hőmérséklet: 0,1; top-p: 0,12; top-k: 10;
- HuSST: prompt #: 29; hőmérséklet: 0,3; top-p: 0,1; top-k: 10;
- HuWNLI: prompt #: 15; hőmérséklet: 0,3; top-p: 0,3; top-k: 10;
- HuRTE: prompt #: 14; hőmérséklet: 0,3; top-p: 0.4; top-k: 10;

6. Eredmények

A 4. táblázatban láthatóak a NER, NP és szentimentanalízis eredmények, a 5. táblázatban pedig a HuLU eredmények. A mostani eredmények a jövőben változhatnak, ha tovább tanítjuk a modelljeinket.

A 4. táblázatban az látható, hogy a PULI BERT-Large modellünk az esetek nagy részében felülmúlta a huBERT modellt. Egyedül az NP feladatban marad le elhanyagolható mértékben. Ahogy az várható volt, az új PULI GPT-2 modellünk felülmúlta a kicsi méretű NYTK-GPT-2 modellt, azonban a BERT modelleket nem, ez sem meglepő, mivel a GPT típusú modelleknek nem az osztályozás az erősségük.

	NerKor (F1)	SzNER (F1)	NP (F1)	2-o (acc)	5-o (acc)
huBERT	90,18	97,51	**96,97**	85,92	68,50
NYTK-GPT-2 small	69,43	88,06	85,02	80,74	61,00
PULI BERT-Large	**91,06**	**97,55**	96,96	**86,29**	**68,99**
PULI GPT-2	71,45	88,25	87,73	81,48	63,75

4. táblázat. NER, NP és szentimentanalízis eredmények

A 5. táblázatban látható a HuLU korpuszain történő kiértékelés, a 6. táblázatban látható a HuCoPA, a HuCOLA és a HuRTE feladatokon mért MCC értékek. A GPT modellek esetében nem mindegyik feladatra tudtunk tesztelni. A HuCoPA egy úgynevezett többválasztásos feladat, amit nem tudtunk első körben finomhangolni a GPT-2 modellel. A GPT-3 esetében viszont még nem sikerült kitapasztalni a promptot erre a típusú feladatra. Hasonlóképpen a HuRC feladatára sem tudtuk most tesztelni a GPT modelleket. A számok alapján a legnehezebb feladat a HuWNLI volt, szinte mindegyik modell 66% alatt teljesített. Ha azt nézzük, hogy ez a feladat tulajdonképpen egy bináris osztályozó feladat, akkor a 66% rendkívül alacsonynak számít, ebből is az következik, hogy ez egy nehéz feladat. A huBERT és a Megatron BERT kimagaslóan teljesít a GPT modellekhez képest, ez várt eredmény, hiszen a GPT modelleknek nem erősségük az osztályozás. Természetesen a GPT-3 a leggyengébben teljesítő modell, mivel külön kutatás szükséges a megfelelő promptok használatára, a mostani kutatás nem erre fókuszált.

A HuLU már jobban árnyékolja a BERT modellek működését. Több feladaton is a huBERT vette át a vezetést. A HuCoPA és a HuRC esetében a Megatron BERT lemarad a huBERT modellhez képest. Ebből arra lehet következtetni, hogy az osztályozási feladatokra jobban be lehet tanítani a Megatron BERT modellt, de a komplexebb nyelvi feladatokban, mint a többválasztásos vagy a kérdés-válasz feladatokban van még hova fejlődnie. Úgy gondoljuk, hogy ez a modell továbbtanításával fejleszthető lehet.

	HuCoPA (acc)	HuCOLA (acc)	HuSST (acc)	HuRC (egyezés/F1)	HuWNLI (acc)	HuRTE (acc)
huBERT	**77,99**	91,43	79,40	**64,50/69,03**	64,93	74,05
PULI BERT-Large	76,59	**91,65**	**79,91**	60,38/65,57	**65,67**	**75,88**
PULI GPT-2	-	85,93	72,79	-	61,94	70,66
PULI GPT-3SX few-shot	-	74,18	68,84	-	63,43	56,33

5. táblázat. HuLU eredmények

	HuCoPA (MCC)	HuCOLA (MCC)	HuRTE (MCC)
huBERT	**0,561**	0,709	0,487
PULI BERT-Large	0,414	**0,711**	**0,517**
PULI GPT-2	-	0,499	0,415

6. táblázat. HuLU MCC eredmények

A GPT-3 teljesítménye rendkívül bizonytalan. Az adott paraméterek mellett minden predikciónál más-más eredményt ad. Valamennyire lehet ezt korlátozni és kontrollálni, de így is rendkívül érzékeny. A különböző promptokkal és

hiperparaméter-beállításokkal végzett többszöri futtatásból a legjobb eredményt írtuk be a 5. táblázatba. A jövőben külön kutatást szeretnénk szentelni a GPT-3 modell kiértékelésére.

Végül a 7. táblázatban látható néhány GPT-3 által generált szöveg különböző hőmérsékleten. A modellnek a következő kérdést tettük fel: 'Mit gondolsz a nyelvtechnológiáról?' A kiemeneti hossz 256 volt, azonban ahol úgy láttuk, hogy a modell témát vált, ott levágtuk a szöveg végét, így a válaszoknak az első néhány mondatai láthatóak csak. Láthatjuk, hogy mindegyik generált bekezdés jól-formált, jól-struktúrált és nyelvileg is helyes.

Mit gondolsz a nyelvtechnológiáról?
1. válasz (0,5 hőmérséklet):
A nyelvtechnológia egy olyan technológia, amely a nyelveket a mesterséges intelligencia segítségével közelíti meg.
2. válasz (0,6 hőmérséklet):
A nyelvtechnológia egy új technológia, ami lehetővé teszi, hogy a számítógépek közvetlenül az emberek agyát használják. Ha az agyat egy számítógéphez kapcsoljuk, akkor képes lesz megérteni a nyelvet. A nyelvtechnológia, a nyelvi feldolgozás és a nyelvtechnológia együtt képesek lesznek arra, hogy a nyelvet az emberek gondolkodásához igazítsák. Nem tudom, hogy hol tart a nyelvtechnológia, de azt tudom, hogy a nyelvtechnológia a következő lépés, és szerintem a nyelvtechnológia a következő nagy lépés.
3. válasz (0,7 hőmérséklet):
A nyelvtechnológia a mesterséges intelligencia egyik ága. A mesterséges intelligencia a gépi tanulással és a nyelvtechnológiával együtt fejlődve alakítja az emberi nyelvet és annak használatát.

7. táblázat. GPT-3 által generált szövegek

7. Összegzés

Kutatásunkban betanítottunk egy 6,7 milliárd paraméteres GPT-3 modellt, egy 345 millió paraméteres GPT-2 modellt és egy 345 millió paraméteres BERT-Large modellt. A GPT-3 modellünk az első egynyelvű GPT-3 modell magyar nyelvre. Több nyelvtechnológiai feladaton is teszteltük modelljeinket. A PULI BERT-Large modellünk a legtöbb feladatban felülmúlta a huBERT modellt. A PULI GPT-3SX modellünkkel magyar nyelvre elsőnek végeztünk prompt programozással, *few-shot* módszerrel kiértékelést. A PULI GPT-3SX számai még messze alulmaradnak a többi modellhez képest, azonban külön kutatást igényel a megfelelő promptok és hiperparaméterek beállítása az adott feladatokra. Azonban képes olyan szöveget generálni, ami hasonlít egy ember által írt szövegre.

Hivatkozások

Alivanistos, D., Santamaría, S.B., Cochez, M., Kalo, J.C., van Krieken, E., Thanapalasingam, T.: Prompting as Probing: Using Language Models for Knowledge Base Construction (2022)

Andonian, A., Anthony, Q., Biderman, S., Black, S., Gali, P., Gao, L., Hallahan, E., Levy-Kramer, J., Leahy, C., Nestler, L., Parker, K., Pieler, M., Purohit, S., Songz, T., Phil, W., Weinbach, S.: GPT-NeoX: Large Scale Autoregressive Language Modeling in PyTorch (8 2021), https://www.github.com/eleutherai/gpt-neox

Benko, V.: Aranea: Yet Another Family of (Comparable) Web Corpora. In: Sojka, P., Horák, A., Kopeček, I., Pala, K. (szerk.) Text, Speech and Dialogue. pp. 247–256. Springer International Publishing, Cham (2014a)

Benko, V.: Compatible Sketch Grammars for Comparable Corpora. In: Abel, A., Vettori, C., Ralli, N. (szerk.) Proceedings of the 16th EURALEX International Congress. pp. 417–430. EURAC research, Bolzano, Italy (jul 2014b)

Black, S., Biderman, S., Hallahan, E., Anthony, Q., Gao, L., Golding, L., He, H., Leahy, C., McDonell, K., Phang, J., Pieler, M., Prashanth, U.S., Purohit, S., Reynolds, L., Tow, J., Wang, B., Weinbach, S.: GPT-NeoX-20B: An Open-Source Autoregressive Language Model. In: Proceedings of the ACL Workshop on Challenges & Perspectives in Creating Large Language Models (2022), https://arxiv.org/abs/2204.06745

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language Models are Few-Shot Learners. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H. (szerk.) Advances in Neural Information Processing Systems. vol. 33, pp. 1877–1901. Curran Associates, Inc. (2020)

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N., Prabhakaran, V., Reif, E., Du, N., Hutchinson, B., Pope, R., Bradbury, J., Austin, J., Isard, M., Gur-Ari, G., Yin, P., Duke, T., Levskaya, A., Ghemawat, S., Dev, S., Michalewski, H., Garcia, X., Misra, V., Robinson, K., Fedus, L., Zhou, D., Ippolito, D., Luan, D., Lim, H., Zoph, B., Spiridonov, A., Sepassi, R., Dohan, D., Agrawal, S., Omernick, M., Dai, A.M., Pillai, T.S., Pellat, M., Lewkowycz, A., Moreira, E., Child, R., Polozov, O., Lee, K., Zhou, Z., Wang, X., Saeta, B., Diaz, M., Firat, O., Catasta, M., Wei, J., Meier-Hellstern, K., Eck, D., Dean, J., Petrov, S., Fiedel, N.: PaLM: Scaling Language Modeling with Pathways (2022)

Csendes, D., Csirik, J., Gyimóthy, T., Kocsor, A.: The Szeged Treebank. In: Matoušek, V., Mautner, P., Pavelka, T. (szerk.) Text, Speech and Dialogue. pp. 123–131. Springer Berlin Heidelberg, Berlin, Heidelberg (2005)

Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). pp. 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota (Jun 2019)

Feldmann, Á., Hajdu, R., Indig, B., Sass, B., Makrai, M., Mittelholcz, I., Halász, D., Yang, Z.Gy., Váradi, T.: HILBERT, magyar nyelvű BERT-large modell tanítása felhő környezetben. In: XVII. Magyar Számítógépes Nyelvészeti Konferencia. pp. 29–36. Szegedi Tudományegyetem, Informatikai Intézet, Szeged, Magyarország (2021)

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)

He, J., Qiu, J., Zeng, A., Yang, Z., Zhai, J., Tang, J.: FastMoE: A Fast Mixture-of-Expert Training System (2021)

Indig, B.: Közös crawlnak is egy korpusz a vége – Korpuszépítés a CommonCrawl .hu domainjából. In: Vincze, V. (szerk.) XIV. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2018). p. 125–134. Szegedi Tudományegyetem Informatikai Intézet, Szegedi Tudományegyetem Informatikai Tanszékcsoport, Szeged (2018)

Jakubíček, M., Kilgarriff, A., Kovář, V., Rychlý, P., Suchomel, V.: The TenTen Corpus Family. In: 7th International Corpus Linguistics Conference CL 2013. pp. 125–127. Lancaster (2013)

Kudo, T., Richardson, J.: SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 66–71. Association for Computational Linguistics, Brussels, Belgium (Nov 2018), https://aclanthology.org/D18-2012

Laki, L., Yang, Z.Gy.: Improving Performance of Sentence-level Sentiment Analysis with Data Augmentation Methods. In: Proceedings of 12th IEEE International Conference on Cognitive Infocommunications (CogInfoCom 2021). pp. 417–422. IEEE, Online (2021)

Ligeti-Nagy, N., Ferenczi, G., Héja, E., Jelencsik-Mátyus, K., Laki, L.J., Vadász, N., Yang, Z.Gy., Váradi, T.: HuLU: magyar nyelvű benchmark adatbázis kiépítése a neurális nyelvmodellek kiértékelése céljából. In: XVIII. Magyar Számítógépes Nyelvészeti Konferencia. p. 431–446. JATEPress, Szeged (2022)

Matthews, B.: Comparison of the predicted and observed secondary structure of T4 phage lysozyme. Biochimica et Biophysica Acta (BBA) - Protein Structure 405(2), 442–451 (1975)

Mittelholcz, I.: emToken: Unicode-képes tokenizáló magyar nyelvre. In: XIII. Magyar Számítógépes Nyelvészeti Konferencia. pp. 61–69. Szegedi Tudományegyetem, Informatikai Intézet, Szeged, Magyarország (2017)

Nemeskey, D.M.: Egy emBERT próbáló feladat. In: XVI. Magyar Számítógépes Nyelvészeti Konferencia. pp. 409–418. Szegedi Tudományegyetem, Szeged (2020a)

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

Nemeskey, D.M.: Introducing huBERT. In: XVII. Magyar Számítógépes Nyelvészeti Konferencia. pp. 3–14. Szegedi Tudományegyetem, Informatikai Intézet, Szeged, Magyarország (2021)

Novák, A., Siklósi, B., Oravecz, Cs.: 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). European Language Resources Association (ELRA), Paris, France (may 2016)

Oravecz, Cs., Váradi, T., Sass, B.: The Hungarian Gigaword Corpus. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14). pp. 1719–1723. European Language Resources Association (ELRA), Reykjavik, Iceland (May 2014)

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language Models are Unsupervised Multitask Learners (2019)

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Journal of Machine Learning Research 21(140), 1–67 (2020)

Reynolds, L., McDonell, K.: Prompt Programming for Large Language Models: Beyond the Few-Shot Paradigm. In: Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems. CHI EA '21, Association for Computing Machinery, New York, NY, USA (2021)

Rychlý, P.: Manatee/Bonito - A Modular Corpus Manager. In: 1st Workshop on Recent Advances in Slavonic Natural Language Processing. pp. 65–70. Masarykova univerzita, Brno (2007)

Shazeer, N.: GLU Variants Improve Transformer (2020)

Shin, T., Razeghi, Y., Logan IV, R.L., Wallace, E., Singh, S.: AutoPrompt: Eliciting Knowledge from Language Models with Automatically Generated Prompts. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 4222–4235. Association for Computational Linguistics, Online (Nov 2020)

Shoeybi, M., Patwary, M., Puri, R., LeGresley, P., Casper, J., Catanzaro, B.: Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism (2019a)

Shoeybi, M., Patwary, M., Puri, R., LeGresley, P., Casper, J., Catanzaro, B.: Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism (2019b)

Simon, E., Vadász, N.: Introducing NYTK-NerKor, A Gold Standard Hungarian Named Entity Annotated Corpus. In: Ekstein, K., Pártl, F., Konopík, M. (szerk.) Text, Speech, and Dialogue - 24th International Conference, TSD 2021, Olomouc, Czech Republic, September 6-9, 2021, Proceedings. Lecture Notes in Computer Science, vol. 12848, pp. 222–234. Springer (2021)

Smith, S., Patwary, M., Norick, B., LeGresley, P., Rajbhandari, S., Casper, J., Liu, Z., Prabhumoye, S., Zerveas, G., Korthikanti, V., Zhang, E., Child, R., Aminabadi, R.Y., Bernauer, J., Song, X., Shoeybi, M., He, Y., Houston, M., Tiwary, S., Catanzaro, B.: Using DeepSpeed and Megatron to Train Megatron-Turing NLG 530B, A Large-Scale Generative Language Model (2022)

Su, J., Lu, Y., Pan, S., Murtadha, A., Wen, B., Liu, Y.: RoFormer: Enhanced Transformer with Rotary Position Embedding (2021)

Suchomel, V., Pomikálek, J.: Efficient Web Crawling for Large Text Corpora. In: Kilgarriff, A., Sharoff, S. (szerk.) Proceedings of the seventh Web as Corpus Workshop (WAC7). pp. 39–43. Lyon (2012)

Szabó, M.K., Vincze, V., Simkó, K.I., Varga, V., Hangya, V.: A Hungarian sentiment corpus manually annotated at aspect level. In: 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)

Szarvas, G., Farkas, R., Kocsor, A.: A Multilingual Named Entity Recognition System Using Boosting and C4.5 Decision Tree Learning Algorithms. In: Todorovski, L., Lavrač, N., Jantke, K.P. (szerk.) Discovery Science. pp. 267–278. Springer Berlin Heidelberg, Berlin, Heidelberg (2006)

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is All you Need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (szerk.) Advances in Neural Information Processing Systems 30, pp. 5998–6008. Curran Associates, Inc. (2017)

Wang, B., Komatsuzaki, A.: GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax (May 2021)

Yang, Z.G., Váradi, T.: Training language models with low resources: RoBERTa, BART and ELECTRA experimental models for Hungarian. In: Proceedings of 12th IEEE International Conference on Cognitive Infocommunications (CogInfoCom 2021). pp. 279–285. IEEE, Online (2021)

Yang Zijian Győző: BARTerezzünk! - Messze, messze, messze a világtól, - BART kísérleti modellek magyar nyelvre. In: XVIII. Magyar Számítógépes Nyelvészeti Konferencia. pp. 15–29. Szegedi Tudományegyetem, Informatikai Intézet, Szeged, Magyarország (2022)

Yang Zijian Győző: "Az invazív medvék nem tolerálják a suzukis agressziót" - Magyar GPT-2 kísérleti modell. In: XVIII. Magyar Számítógépes Nyelvészeti Konferencia. pp. 463–476. Szegedi Tudományegyetem, Informatikai Intézet, Szeged, Magyarország (2022)