BARTerezzünk! Messze, messze, messze a világtól, BART kísérleti modellek magyar nyelvre

Yang Zijian Győző

Nyelvtudományi Kutatóközpont
1068 Budapest, Benczúr u. 33.
yang.zijian.gyozo@nytud.hu
MTA-PPKE Magyar Nyelvtechnológiai Kutatócsoport
1083 Budapest, Práter u. 50/a.
yang.zijian.gyozo@itk.ppke.hu

Kivonat A BART autoregresszív típusú modell, amely elsősorban szöveggenerálási feladatokra alkalmas. A kutatásomban különböző BART modelleket tanítottam magyar nyelvre és azokat finomhangoltam különböző szöveggenerálási feladatokra. A kísérleteimben BART base és large modelleket tanítottam magyar és angol-magyar nyelvekre. Az előtanított BART modelleket szövegosztályozás, absztraktív szövegösszefoglaló generálás, gépi fordítás és versgenerálás feladatokra finomhangoltam. Az eredmények alapján a BART kevésbé teljesít jól szövegosztályozás feladatára, de absztraktív szövegösszegzés feladatában „state of the art" eredményeket értem el. Érdekességként a kutatásom végén egy Petőfi versgenerátort mutatok be.
Kulcsszavak: BART, absztraktív összefoglaló generálás, szentiment analízis, szövegosztályozás, gépi fordítás, szöveggenerálás, versgenerálás

1. Bevezetés

Kutatásomban különböző BART modellekkel kísérleteztem. A BERT alapú modellek, dekóder hiányában, kevésbé alkalmasak szöveggenerálás feladataihoz, mint például a szövegösszefoglaló generálás vagy gépi fordítás. A BART egy enkóder-dekóder architektúrájú modell, ezért alkalmas szöveggenerálásra. A kutatásomban különböző BART modelleket tanítottam be magyar és angol-magyar nyelvekre. Kísérleteztem base és large modellekkel egyaránt, majd az előtanított modelleket különböző nyelvtechnológiai feladatokra finomhangoltam. Kutatásom során kísérleteket végeztem a szövegosztályozás területén, továbbá létrehoztam különböző absztraktív összefoglaló generáló modelleket, gépi fordító modelleket és egy versgenerátort. A címben szereplő „Messze, messze, messze a világtól," sort a versgenerátor generálta a „BARTelezzünk!" folytatásaként.

Modelljeim és szkriptjeim megtalálhatóak a Github[1] és Hugging Face[2] oldalakon.

2. Kapcsolódó irodalom

Jelen nyelvtechnológiai feladatok megoldásához az egyik alapvető megközelítés a nyelvi modellek előtanítása, majd azok tovább finomhangolása az adott specifikus feladatra. A konkrét természetes nyelvi feldolgozással kapcsolatos feladatok megoldására a különböző architektúrájú nyelvi modellek teljesítménye különböző. Az utóbbi években a nyelvtechnológia területén a transzformer (Vaswani és mtsai, 2017) architektúrájú modellek dominálnak. A BERT (Devlin és mtsai, 2019; Conneau és mtsai, 2020) típusú modellek bemutatták a maszkolt nyelvi modellezést, amelyek rendkívül magas pontosságot értek el a különböző token és mondatszintű osztályozásos feladatokban. Azonban ezek a modellek kevésbé alkalmasak szöveggenerálásra, mint például a szövegösszefoglalásra. Magyar nyelvre két BERT modell érhető el, a huBERT (Nemeskey Dávid Márk, 2021) és a HILBERT (Feldmann és mtsai, 2021).

A szöveggenerálás feladatára fejlesztették ki az autoregresszív típusú modelleket, mint például a GPT (Radford és Narasimhan, 2018), amelyek „balról jobbra" (left-to-right) modellek, azaz csak a szöveg bal oldalát látják a tanítás során, így rendkívül erősek abban, hogy kitalálják a még hiányzó részeket, a szöveg folytatását.

Az ELMo (Peters és mtsai, 2018) egy bal és egy jobb oldali reprezentációt konkatenál össze, azonban a bennük lévő jegyeket a tanítás során nem hangolja össze, így kevesebb összefüggést tudnak megtanulni.

A MASS (Song és mtsai, 2019) modell rendkívül hasonlít a BART modellhez. A bemeneti szövegből folytonosan kimaszkolnak tokeneket, majd ehhez a szöveghez a kimeneten hozzárendelik a hiányzó tokeneket. Ez a módszer azonban diszkriminatív feladatokra kevésbé hatékony.

Absztraktív összefoglalás területén a BART egyik legnagyobb ellenfele, a PEGASUS (Zhang és mtsai, 2020), amely az előtanítás során a fontosnak vélt egész mondatokat lemaszkolja a bementi dokumentumban.

Gépi fordítás területén a M2M100 (Fan és mtsai, 2020) modell egyetlen modellel képes 100 nyelvről 100 nyelvre fordítani. Tanításakor csak olyan párhuzamos korpuszokat használtak, ahol az angol a forrás vagy a célnyelv.

A Google a transzfer tanulás területén végzett kísérletet a T5 (Raffel és mtsai, 2020) modellel, amely egy nagy korpuszon tanított sztenderd enkóder-dekóder architektúrájú transzformer modell. A különbség más modellek finomhangolásától, hogy a T5 sokféle specifikus feladatot egy modellel tanít be, méghozzá szövegből szöveg (text-to-text) feladatként, legyen az gépi fordítás, vagy osztályozás.

[1] `https://github.com/nytud/neural-models`
[2] `https://huggingface.co/NYTK`

3. BART modell

A BART modell egy enkóder-dekóder architektúrán alapuló transformer modell, amelyet a Facebook fejlesztett[3]. Az enkóder kétirányú (Bidirectional), a dekóder autoregresszív (Autoregressive). A BART gyakorlatilag ötvöz egy BERT és egy GPT típusú modellt. A BART enkóder tanítása abban különözik a BERT-től, hogy amíg a BERT veszteségfüggvényét arra optimalizálták, hogy megtanulja a kimaszkolt tokenek visszaállítását és azt, hogy két mondat egymást követőeke, addig a BART csak olyan feladatokat tanult, amelyek „zajtalanítanak". A BART enkóder az alábbi feladatok alapján optimalizálja a veszteségfüggvényt: *token maszkolás* (Véletlenszerűen kimaszkolt tokenek visszaállítása), textittoken törlés (Véletlenszerűen kitörölt tokenek helyének meghatározása), *Szöveg kitöltés* (A SpanBERT (Joshi és mtsai, 2020) módszerén alapszik, azonban itt Poisson eloszlás alapján számolják ki a hosszakat, majd a hossz alapján kerülnek kimaszkolásra szövegrészletek. Ezzel azt tanulja meg a modell, hogy hány tokent maszkoltak ki. Az eredeti BERT-el ellentétben, nem önálló szóelemeket maszkol ki, hanem egész szavakból álló folytonos szövegrészeket. Ezzel azt érik el, hogy a modell a szövegkörnyezet alapján nagyobb összefüggő szövegrészeket tud megtanulni.), *mondat permutáció* (Pont írásjel alapján mondatokra bontja a szöveget, majd véletlenszerűen megkeveri. A modell ezzel megtanulja, hogy milyen sorrendben voltak eredetileg a mondatok) és *dokumentum rotáció* (Véletlenszerűen kiválasztanak egy tokent, majd úgy forgatják a szöveget, hogy ez a kiválasztott token legyen az első token. A modell ezzel azt tanulja meg, hogy melyik tokenek lehetnek dokumentumkezdők.). A BART az előtanítás során egy dokumentumot ellát zajokkal, majd áteresztve az enkóder-dekóder architektúrán, a dekóder kimenetére és az eredeti dokumentumra számolja ki a veszteséget. A BART egy nyelvi modellnek felel meg.

4. Előtanítás

A Facebook nem tette közzé az előtanítás szkriptjét, de a Hugging Face könyvtárai[4] tartalmaznak előtanítási kódokat. A BART előtanításához a BartForCausalLM függvényt használtam. A BartForCausalLM a BART modell dekóder önálló része, melynek a tetején egy nyelvmodell réteg (gyakorlatilag egy softmax, ami segít a következő token kiválasztásában) található. Ez alkalmas a következő szó prediktálására (causal language modeling). A modell tovább finomhangolható. A kutatásom során öt különböző BART modellt tanítottam elő:

- **BART-base-512**: Egynyelvű magyar BART base modell, 512 bemeneti hosszal.
- **BART-base-1024**: Egynyelvű magyar BART base modell, 1024 bemeneti hosszal.

[3] `https://github.com/pytorch/fairseq/tree/master/examples/bart`
[4] `https://huggingface.co/transformers/model_doc/bart.html`

– **BART-large**: Egynyelvű magyar BART large modell. Erőforrás hiányában, csak részleges kiértékelés történt ezzel a modellel.
– **BART-base-enhu**: Angol-magyar kétnyelvű BART base modell.
– **BART-large-enhu**: Angol-magyar kétnyelvű BART large modell. Erőforrás hiányában, csak részleges kiértékelés történt ezzel a modellel.

4.1. Felhasznált korpuszok

Az egynyelvű BART modellek tanításához a Webcorpus 2.0-t (Nemeskey, 2020) használtam. Az eredeti BART kutatás (Lewis és mtsai, 2020) alapján a korpuszból bekezdéseket nyertem ki, amelyek legalább egy darab pont írásjellel rendelkeztek.

Az angol-magyar kétnyelvű BART modell tanításához az angol WikiText-103 (Merity és mtsai, 2017) és a Webcorpus 2.0 magyar Wikipédia részét használtam. Hasonlóan az egynyelvű korpuszhoz, azokat a bekezdéseket hagytam meg, amelyek legalább egy darab pont írásjellel rendelkeztek.

Mind a három korpusz alapból tokenizálva volt. Az így létrejött korpuszok tulajdonságai az 1. táblázatban láthatóak.

	szegmens	token	type	bekezdés mondatszám (medián)	bekezdés tokenek száma (medián)
Webcopus 2.0	100.255.504	9.095.424.717	57.562.212	3	60
Angol WikiText-103	707.391	96.534.563	596.820	5	125
Magyar Wikipédia	1.098.156	90.349.849	3.137.980	4	69

1. táblázat. Előtanításhoz használt korpuszok jellemzői.

4.2. Modellek tulajdonságai és tanítása

A 2. táblázatban láthatóak a főbb különbségek a tanított modellek hiperparaméterei között. Főbb különbségek az enkóderek és dekóderek rétegeinek számában (Rétegek #), rejtett rétegeinek méretében (Rejtett), a figyelmi fejeinek számában (Fejek #), az előrecsatolt köztes rétegeinek méretében (FFN dim), valamint a bemeneti szöveg hosszában (Bemenet) és a szótár méretében (Szótár) mutatkoznak meg. A szótárak esetében a BART-base-enhu és a BART-large-enhu angol-magyar kétnyelvű szótárral, a többi modell magyar egynyelvű szótárral rendelkeznek.

A 3. táblázatban láthatóak a tanítás szempontjából fontosabb tulajdonságok. A tanulási ráta mindegyik modell esetén 2e-8 volt. Az egyik kiemelendő információ a mentési pont. A modellek tanításai során egyszer sem konvergált a modell, de ez még nem jelenti azt, hogy nem tanult meg semmit. Ezért különböző mentési pontoknál kivettem egy-egy modellt és finomhangolással (osztályozás

	Réteg #	Rejtett	Fejek #	FFN dim	Bemenet	Szótár
BART-base-512	6	768	12	3072	512	30.000
BART-base-1024	6	768	12	3072	1024	30.000
BART-large	12	1024	16	4096	1024	30.000
BART-base-enhu	6	768	12	3072	512	40.000
BART-large-enhu	12	1024	16	4096	1024	40.000

2. táblázat. Modellek tulajdonságai.

és szövegkivonatolás) teszteltem, hogy a modelljeim finomhangolhatóak-e. A 3. táblázatban láthatóak azok a mentési pontok, valamint a hozzájuk tartozó veszteségi értékek (Loss), amelyek végül kiválasztásra kerültek, és alapot képeznek a jelen kutatás további részeihez. A mentési pontok variabilitása azzal magyarázható, hogy különbözőek a batch méretek, a hardver hátterek és az a tény, hogy nem mindig volt elegendő erőforrás a továbbtanításra.

	Gép (4 db)	Batch (per GPU)	Mentési pont (lépés)	Loss
BART-base-512	Tesla V100S - 32GB	50	150.000	1,14
BART-base-1024	Tesla V100S - 32GB	8	290.000	2,29
BART-large	Tesla V100S - 32GB	8	220.000	2,22
BART-base-enhu	GeForce GTX 1080 - 12GB	12	170.000	1,44
BART-large-enhu	Tesla V100S - 32GB	7	500.000	2,75

3. táblázat. Előtanítás tulajdonságai.

5. Finomhangolás

Az előtanított modelleket 4 különböző feladatra finomhangoltam: Mondatszintű szentiment analízis szövegosztályozás (SZENT), Absztraktív összefoglalás, szövegkivonatolás (SZUM), Gépi fordítás (GF) és Szöveggenerálás: Petőfi versgenerálás (VERS).

5.1. Felhasznált korpuszok

A finomhangoláshoz a különböző feladatokra az alábbi korpuszokat használtam fel, melynek tulajdonságai A 4. táblázatban láthatóak:

- **HI**: HVG korpusz + index.hu korpusz, amelyből a HVG korpusz online cikkeket tartalmaz 2012–2020 időszakból, az index.hu korpusz online cikkeket tartalmaz 1999–2020 közötti időszakból.
- **NOL**: Népszabadság online korpusz; a nol.hu online cikkeit tartalmazza a 1999–2016 közötti időszakból.

– **MARCELL** (Váradi és mtsai, 2020): Jogi szövegek (dok) és a hozzájuk tartozó egy soros leírások 1991–2019 közötti időszakból.
– **MTS**: Magyar Twitter Szentiment Korpusz[5], a Precognox Kft.[6] jóvoltából. A korpusz 5 osztályos (MTS5), ahol 1 a legnegatívabb és 5 a legpozitívabb. Készítettem belőle egy 3 osztályos változatot (MTS3), ahol a 1-es és 2-es értékeket negatívként jelöltem, a 3-as értéket semlegesnek, valamint a 4-es és 5-ös értékeket pozitívként. Végül készítettem egy bináris változatot is (MTS2), ahol a 3-as értékű szegmenseket kihagytam, mert nem lehet eldönteni róluk egyértelműen, hogy pozitív vagy negatív.
– **SST**: Stanford Sentiment Treebank (Socher és mtsai, 2013), angol nyelvű szentiment analízis korpusz. Két változata van, a bináris osztályú SST-2 és az 1-5 likert skálájú SST-5.
– **OPUS**: OPUS (Tiedemann, 2012) korpuszból vett angol-magyar párhuzamos alkorpuszok gépi fordításhoz. Felhasznált alkorpuszok: ParaCrawl, OpenSubtitles, Tatoeba, DGT, WikiMatrix, EUbookshop, PHP manual, TED2020, KDEdoc, KDE4.
– **PETŐFI**: Petőfi Sándor összes költeményei mű letöltve a Magyar Elektronikus Könyvtár oldaláról[7].

	Feladat	Szegmens	Token #	Type #	Átlag token #
HI	SZUM	559.162	147.099.485 (cikk) 16.699.600 (lead)	2.949.173 (cikk) 749.586 (lead)	263,07 (cikk) 29,87 (lead)
NOL	SZUM	397.343	153.003.164 (cikk) 15.786.166 (lead)	2.482.398 (cikk) 623.445 (lead)	384,52 (cikk) 39,71 (lead)
MARCELL	SZUM	24.747	27.834.358 (dok) 277.732 (leírás)	444.352 (dok) 29.189 (leírás)	1124,82 (dok) 11,59 (leíras)
MTS2	SZENT	2.737	42.797	13.713	15,62
MTS3, MTS5	SZENT	4.000	59.997	18.423	14,99
OPUS	GF	56.837.602	613.206.646 (en) 507.702.362 (hu)	2.691.229 (en) 6.886.205 (hu)	10,79 (en) 8,93 (hu)
PETŐFI	VERS	854 (költemény)	151.486	50.029	-

4. táblázat. Finomhangoláshoz használt korpuszok tulajdonságai.

5.2. Finomhangolás kísérletek

A mondatszintű szentiment analízis szövegosztályozás feladatához, a magyar és az angol-magyar modellek tanításához, a Magyar Twitter Szentiment Korpuszt használtam, míg az angol-magyar kísérletekhez az SST korpuszokat.

[5] `http://opendata.hu/dataset/hungarian-twitter-sentiment-corpus`
[6] `https://www.precognox.com`
[7] `https://mek.oszk.hu/01000/01006/`

Az angol-magyar modell esetében mindamellett, hogy betanítottam a modelljeimet az eredeti SST korpuszokra és kiértékeltem, *zeroshot* kísérletet is végeztem, vagyis magyar szövegen való tanítás nélkül végeztem osztályozást az MTS tesztanyagon. Továbbá végeztem *transzfer* kísérletet (tf) is, ami az esetemben azt jelentette, hogy az angol SST finomhangolás után, tovább finomhangoltam az angol-magyar modellemet a magyar MTS korpuszon. Mindegyik tanítást maximum 128 bemeneti szöveghosszal, 2e-5 tanulási rátán (learning rate), 4-es batch/GPU (4 db GeForce GTX 1080 - 12GB) méreten és 15 epoch számon tanítottam. A finomhangoláshoz a Huggingface Transformers githubján található példakódot[8] használtam fel.

Az absztraktív összefoglalás, szövegkivonatolás feladatához a HI, a NOL és a MARCELL korpuszokat használtam. Összehasonlíthatóság végett, a NOL korpuszon betanítottam egy BERT alapú (huBERT modellel) absztraktív modellt a PreSumm eszközzel (Liu és Lapata, 2019), ugyanazokkal a beállításokkal, mint amit Yang és mtsai (2021) a kutatásaikban használtak. A BART base modelleket 512 / 1024 maximum bemeneti és 256 maximum kimeneti szöveghossz, 8-as batch/GPU (4 db GeForce GTX 1080 - 12GB) méret, 80 epoch, 2e-5 tanulási ráta, 15 ezer warmup lépés és fp16 beállítási hiperparaméterekkel tanítottam. Az egyetlen magyar large modellemet sikerült 4 db Tesla V100S - 32GB GPU-n finomhangolni. Azonban korlátolt erőforrás miatt, csak 20 epochon.

A sima absztraktív modellek tanítása mellett végeztem **transzfer** kísérleteket is, ami az esetemben azt jelentette, hogy az angol-magyar base modellemet betanítottam az angol CNN/Daily Mail korpuszon (40 epoch), majd a betanított modellt továbbtanítottam a HI és a NOL korpuszon. Erőforrás hiányában nem tudtam az angol-magyar large modellemet finomhangolni. A finomhangoláshoz a Huggingface Transformers githubján található példakódot[9] használtam fel.

A gépi fordítás feladatához a BART-base-enhu modellemmel tanítottam angol-magyar (enhu) és magyar-angol (huen) gépi fordító modelleket. A tanításhoz az OPUS korpuszban található angol-magyar alkorpuszokat használtam. Angol-magyar nyelvre egy 512 maximum bemeneti és 512 maximum kimeneti szöveghosszú modellt és egy 128 maximum bemeneti és 128 maximum kimeneti szöveghosszú modellt tanítottam. Erőforrás és idő hiányában, magyar-angol nyelvre csak 128 maximum bemeneti és 128 maximum kimeneti szöveghosszú modellt tanítottam. Az 512 szöveghosszú modellt 4-es batch/GPU (4 db GeForce GTX 1080 - 12GB) mérettel és 1 epoch számmal, míg a 128 szöveghosszú modelleket 26-os batch/GPU (4 db GeForce GTX 1080 - 12GB) mérettel és 2 epoch számmal tanítottam. További fontosabb hiperparaméterek: 15 ezer warmup lépés, fp16, 5e-5 tanulási ráta. A finomhangoláshoz a Huggingface Transformers githubján található példakódot[10] használtam fel.

[8] `https://github.com/huggingface/transformers/tree/master/examples/pytorch/text-classification`

[9] `https://github.com/huggingface/transformers/tree/master/examples/pytorch/summarization`

[10] `https://github.com/huggingface/transformers/tree/master/examples/pytorch/translation`

A szöveggenerálás feladatához a Petőfi Sándor összes költeménye című kötetet használtam fel. A tanítóanyag létrehozásához kitöröltem a címeket, keltezéseket (hely és dátum) és a tartalomjegyzéket. Két vers közé beraktam egy <s> címkét, ami jelzi a versek végét. Majd 3-as (sor) ablakot végigcsúsztatva a verseken generáltam forrás- és célnyelvi szövegeket. A forrásnyelvi szöveg lehetett 1 sor, 2 sor (egymást követő) és 3 sor (egymást követő), a kimeneti szöveg az 1, 2 vagy 3 sornak a következő sora. A BART modell jellege miatt, a forrásnyelvi szövegből véletlenszerűen kimaszkoltam 0-25% szót. Ily módon 99.453 sor tanítóanyag és 3000 sor validálási anyag keletkezett. A feladatot absztraktív generálás feladatként értelmeztem, ami egy szövegből-szöveg (seq2seq) generálási feladat. A forrásnyelvi szöveg egy hasonló méretű vagy hosszabb szöveg, mint a célnyelvi szöveg, ami a folytatása a forrásnyelvi szövegnek. Ezért a tanításhoz az absztraktív generáláshoz használt kódot használtam (ugyanazokkal a hiperparaméterekkel). Mivel a vers sorai nagyon rövidek, ezért maximum 128 bemeneti és maximum 128 kimeneti szöveghosszt használtam. A tanításhoz a BART-base-1024 modellt használtam fel. A vers generálásakor kézzel kell megadni az első sort, ez alapján generál a modell egy sor folytatást, majd az általunk megadott és az általa adott folytatásra generál egy újabb sort, a így keletkezett 3 sorra generál egy 4-dik sort. Ezután 3-as (sor) ablakkal tovább csúsztatva generálja a következő sorokat. A végső demóban egy minimális rímkényszert is próbáltam alkalmazni. A modell öt lehetséges folytatást generál, amelyeket sorrendbe raktam az alapján, hogy mennyire rímel a kettővel előtte lévő sorra. A rímet három magánhangzó mélységig vizsgáltam.

6. Eredmények

Az 5. táblázatban láthatóak a mondatszintű szentiment analízis osztályozás kísérlet eredményei. Egyértelműen látszik, hogy a BART modelljeim szignifikánsan alulmaradnak a huBERT-hez képest. Ez nem meglepő, hiszen az autoregresszív modelleknek nem erőssége az osztályozás, de még így is értékelhető minőségben lehet betanítani osztályozásos feladatokra. A legjobb eredményt a BART-base-512 adta, ez annak tudható be, hogy ezt a modellt sikerült a legnagyobb batch méret mellett tanítani. A kiértékeléshez a pontosság (accuracy) metrikát használtam. Mindegyik mérésnél 15 epochig tanítottam, azonban az eredmények táblázatba csak a legjobb eredmények kerültek be. Jellemzően a 3-5 epoch szám között érték el a legmagasabb pontosságot. Angol-magyar modellek esetén az SST korpuszból nem készítettem 1-3 likert skálájú alkorpuszt, ezért üresek ezek a mezők a zeroshot és a transzfer eredményeknél. A BART-base-enhu modell esetében az angol-magyar modellt finomhangoltam az MTS korpuszon A zeroshot esetében az angol-magyar modellt az SST korpuszon finomhangoltam, majd egyből kiértékeltem az MTS korpuszon, végül a transzfer esetében az SST korpuszon finomhangolt modellt továbbfinomhangoltam az MTS korpuszon. Az eredményekből az látszik, hogy a többnyelvű modellek gyengébben teljesítenek, mint az egynyelvű modellek, ami szintén várható volt, hiszen ezek csak Wikipédia anyagon tanultak és egyszerre kellett angolul és magyarul is tanulniuk.

A zeroshot eredmények meglehetősen gyengék, a transzfer kísérletek csak egy kicsivel tudták javítani a modell minőségét. Érdekes megfigyelés, hogy a large modellek nem teljesítenek jobban a base modelleknél. Továbbá az eredmények közé beillesztettem még az angol-magyar modelleknek az SST korpuszokon mért teljesítményét is.

	MTS2	MTS3	MTS5
huBERT	85.92	72.18	68.50
BART-base-512	**79,25**	**61,40**	**58,75**
BART-base-1024	76,66	56,89	57,75
BART-large-1024	76,29	54.88	58.75
BART-base-enhu	74,44	60,15	56,75
BART-base-enhu (zeroshot)	42,96	-	28,75
BART-base-enhu (transzfer)	74,81	-	57,25
BART-large-enhu	74,07	59,14	56,00
BART-large-enhu (zeroshot)	44,81	-	23,50
BART-large-enhu (transzfer)	72,59	-	56,74
	SST2	-	SST5
BART-base-enhu	79,01	-	36,72
BART-large-enhu	80,27	-	36,36

5. táblázat. Mondatszintű osztályozás eredménye.

A 6. táblázatban láthatóak az absztraktív összefoglalás kísérlet eredményei. A modell legnagyobb erőssége ebben a feladatban mutatkozik meg. Szignifikánsan jobb eredményt értem el a BART alapú modellekkel, mint a BERT alapú megoldással. Yang és mtsai (2021) munkájukban a fedés eredményeket publikálták. Azonban csak a PreSumm eszközre jellemző, hogy több, hosszabb szöveget generál kimenetnek. Összehasonlítva a HI korpuszon:

– Eredeti lead méretek: Átlag: 26,42, Medián: 24.
– PreSumm összefoglalók méretei: Átlag: 104,61, Medián: 105.
– BART-base-512 összefoglalók méretei: Átlag: 28, Medián: 24.

Az összehasonlításból észrevehető, hogy a PreSumm rendkívül hosszú összefoglalókat generál, ezért nem meglepőek a magas fedés mértékek, azonban így a pontosság mértékek drasztikusan csökkennek (látszik a 6. táblázat PreSumm F-mértékeiből). Az összehasonlításból látszik, hogy a BART törekszik a hossz megtanulására is, és közel olyan hosszúságú összefoglalókat generál mint az eredeti leadek (annak ellenére, hogy maximum 128 kimeneti hosszra van beállítva).

Azonban a MARCELL korpuszon már az esetek nagy részében alulteljesítenek a BART modellek a PreSummhoz képest. Ez annak tulajdonítható, hogy a rövidségre való törekvése most a hátrányára fordult. Egyedül a transzfer tanítással készült modell tudott magasabb eredményt elérni. A méretek összehasonlítva a MARCELL korpuszon:

– Eredeti lead méretek: Átlag: 11.59, Medián: 9.

– PreSumm összefoglalók méretei: Átlag: 11,466, Medián: 9.
– BART-base-512 összefoglalók mérekei: Átlag: 9,97, Medián: 8.

Továbbá azt is figyelembe kell venni, hogy a PreSumm kísérletben a generált szövegből csak az első mondatot vették figyelembe, azonban a rendszer alapból több mondatot is generált.

Érdekes eredmény, hogy annak ellenére, hogy a MARCELL korpuszban rendkívül hosszú a bemeneti szöveg, a hosszú bemeneti szövegű BART-base-1024 teljesített a leggyengébben. Ebből arra tudok következtetni, hogy az egysoros leíráshoz a releváns információk inkább a bemeneti szöveg elején találhatóak, ezért a rendkívül hosszú szöveg csak megzavarja a generálást. Azonban az, hogy a transzfer tanítás ilyen mértékben tudta javítani a teljesítményt, azt jelentheti, hogy az angol tudásból olyan információt tudott kinyerni, ami segített neki a finomhangolásban.

A kiértékeléshez a ROUGE (Lin, 2004) metrikát használtam. A 6. táblázatban az F-mértékek láthatóak a következő formátumban: ROUGE-1/ROUGE-2/ROUGE-L.

Összehasonlíthatóság végett, a 6. táblázat végére beillesztettem az eredeti BART modell eredményét (a CNN/Daily Mail korpuszon)[11]. A kutatásom célja nem az angol eredmények felülmúlása volt, ezért csak 40 epoch számon tanítottam. Figyelembe véve, hogy kevesebb epoch szám mellett és csak Wikipédia szövegeken tanult elő a modell, mindössze 4% körüli értékkel marad csak le az eredeti BART modell eredményétől. A magyar modelleket nézve, magasabb epoch számon még jobb eredményt tudtam volna elérni. A tapasztalat az epoch számot illetően az, hogy a nagyobb epoch szám az összefoglaló generálás esetében nem eredményezett túltanulást.

A large modellekkel való kísérletek esetében, csak 40 epochig tanultak, kicsi batch méreten, így ők teljesítettek a leggyengébben. Továbbá erőforrás hiányában a NOL korpuszon nem sikerült finomhangolni.

	HI	NOL	MARCELL
PreSumm (huBERT)	22,42/10,24/18,72	26,34/10,90/22,01	75,85/68,35/74,61
BART-base-512	30,18/13,86/22,92	46,48/32,40/39,45	71,25/62,79/69,75
BART-base-1024	**31,86/14,59/23,79**	**47,01/32,91/39,97**	71,01/62,58/69,42
BART-large	30,12/13,07/22,72	-	70,24/60,69/68,53
BART-base-enhu	31,36/14,34/23,48	42,71/27,59/35,38	71,47/63,04/69,93
BART-base-enhu-tf	31,76/14,47/23,47	45,05/30,46/37,64	77,06/70,64/75,96
	CNN/Daily Mail		
BART-base-enhu	40,07/17,61/27,35		
BART eredeti	44,16/21,28/40,90		

6. táblázat. Absztraktív összefoglaló generálás F-mérték eredmények.

[11] `https://paperswithcode.com/sota/abstractive-text-summarization-on-cnn-daily`

A 9. táblázatban látható egy példa arra, hogy a különböző modellek milyen összefoglalókat generáltak. A példa önmagában is összetett, kétféle készülékről is ír, ezért nehezen állapítható meg, hogy melyik is a fontosabb információ. Igyekeztem olyan példát mutatni, ami inkább a modellek határait, hátrányait mutatja. A példában egyértelműen látszik a ROUGE metrika egyik hátránya, miszerint az eredti lead szövege meglehetősen szűkszavú, figyelemfelkeltő, de semmi hasznos információt nem szolgáltat, remélve, hogy megmozgatva az olvasó kíváncsiságát, bevonzza őt. Ez azonban torzít a ROUGE értékeken, hiszen a metrikával azt mérjük, hogy mennyire hasonlít a gép által generált szöveg a leadhez. Továbbá szembetűnő a PreSumm által generált szöveg hosszúsága. Ha a tartalmat nézzük a nagy része hű az eredeti cikkhez. Megfigyelhető még a BART-base-512 helyesírási hibája, ez más példákban is megjelenik, valamint a BART-large modell erősebb „hallucinációja", amelyek más példákban is megmutatkoznak. Ezek a számokban is észlelhetőek, hiszen ezek teljesítettek a leggyengébben. A generált mondatok nyelvtanilag helyesek, tartalmilag viszont csak részlegesen felelnek meg az eredeti cikkeknek.

Az egynyelvű BART base modelljeim (HI[12,13] és NOL[14,15]) elérhetőek a Hugging Face oldalon.

A 7. táblázatban láthatóak a gépi fordítás kísérlet eredményei. Referenciaként a Google fordítót[16] választottam, két okból. Első ok, hogy érdekelt a Google fordító mai állapota, és hogy vajon jobbak-e nála az általam tanított modellek. A másik ok, hogy egy szabadon elérhető neurális alapokon működő gépi fordítót kerestem, mivel nem állt szándékomban külön másik neurális gépi fordító rendszert tanítani. A meglévő szabadon elérhető rendszerek közül az egyik legnépszerűbb rendszer mellett döntöttem. Az eredmények azt mutatják, hogy sikerült mindegyik esetben szignifikánsan felülmúlni a Google fordítót. Kiértékeléshez a BLEU (Papineni és mtsai, 2002) és a 3-gram chrF (Popović, 2015) metrikákat használtam. Érdekes, hogy az 512 bemeneti hosszal rendelkező modell 1 epoch alatt hasonló vagy jobb eredményt ért el, mint a 128 bementi hosszal rendelkező modell 2 epoch alatt.

Az 512 bemeneti hosszal rendelkező BART base fordító modelljeim[17,18] elérhetőek a Hugging Face oldalon.

Végül, de nem utolsó sorban a 8. táblázatban látható a szöveggenerálás kísérlet eredménye. A kvantitatív kiértékeléshez az absztraktív összefoglalónál használt ROUGE metrikát használtam. Azonban egy ilyen jellegű feladatnál az automatikus kiértékelési metrikák kevésbé relevánsak, sőt az emberi kiértékelés sem egyértelmű. Ezért inkább egy generált verset tettem be, ahol az első sort magam

[12] https://huggingface.co/NYTK/summarization-hi-bart-hungarian
[13] https://huggingface.co/NYTK/summarization-hi-bart-base-1024-hungarian
[14] https://huggingface.co/NYTK/summarization-nol-bart-hungarian
[15] https://huggingface.co/NYTK/summarization-nol-bart-base-1024-hungarian
[16] https://translate.google.hu
[17] https://huggingface.co/NYTK/translation-bart-en-hu
[18] https://huggingface.co/NYTK/translation-bart-hu-en

	BLEU	chrF-3
Google en-hu	25,30	54,08
BART-base-enhu (512, 1 epoch)	34,38	58,88
BART-base-enhu (128, 2 epoch)	33,59	58,23
Google hu-en	34,48	59,59
BART-base-huen (512, 1 epoch)	38,03	61,37
BART-base-huen (128, 2 epoch)	38,63	61,58

7. táblázat. Gépi fordítás eredményei.

adtam meg manuálisan. Mivel Petőfi Sándor költeményein tanult a modell, a generált szöveg „erősen Petőfi Sándor stílusú”.

Szegeden, január végén,
Lopott, koldult és magamért,
Lelkem reája...
Szeretlek téged, kedvesem,
Hol a boldogság mostanában?
Barátságos meleg szobába.
Sötétség volt, mint a hold,
S mint a hold, a csillag az éjben,
16.51/12.93/16.53

8. táblázat. Versgenerálás eredménye.

7. Összegzés

Kutatásomban különböző BART modelleket tanítottam magyar nyelvre, majd különböző nyelvtechnológiai feladatokra továbbtanítottam őket. A kísérletem során magyar nyelvű és magyar-angol nyelvű BART base és large modelleket tanítottam elő. Majd ezeket a előtanított modelleket szövegosztályozás, absztraktív generálás, gépi fordítás és szöveggenerálás feladataira finomhangoltam. Az eredmények azt mutatták, hogy a BART, mint autoregresszív modell elsősorban szöveggenerálás feladataira teljesít jól, azon belül is absztraktív szövegösszefoglaló feladatában. Ezen a területen magyar nyelvre „state of the art” eredményeket értem el.

Hivatkozások

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., Stoyanov, V.: Unsupervised cross-lingual representation learning at scale. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 8440–8451. Association for Computational Linguistics, Online (Jul 2020)

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)

Fan, A., Bhosale, S., Schwenk, H., Ma, Z., El-Kishky, A., Goyal, S., Baines, M., Çelebi, O., Wenzek, G., Chaudhary, V., Goyal, N., Birch, T., Liptchinsky, V., Edunov, S., Grave, E., Auli, M., Joulin, A.: Beyond english-centric multilingual machine translation. ArXiv abs/2010.11125 (2020)

Feldmann, Á., Hajdu, R., Indig, B., Sass, B., Makrai, M., Mittelholcz, I., Halász, D., Yang, Z.G., 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)

Joshi, M., Chen, D., Liu, Y., Weld, D.S., Zettlemoyer, L., Levy, O.: SpanBERT: Improving Pre-training by Representing and Predicting Spans. Transactions of the Association for Computational Linguistics 8, 64–77 (01 2020)

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., Zettlemoyer, L.: BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 7871–7880. Association for Computational Linguistics, Online (Jul 2020)

Lin, C.Y.: ROUGE: A package for automatic evaluation of summaries. In: Text Summarization Branches Out. pp. 74–81. Association for Computational Linguistics, Barcelona, Spain (Jul 2004)

Liu, Y., Lapata, M.: Text summarization with pretrained encoders. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing. pp. 3730–3740. Association for Computational Linguistics, Hong Kong, China (2019)

Merity, S., Xiong, C., Bradbury, J., Socher, R.: Pointer sentinel mixture models. In: 5th International Conference on Learning Representations. Palais des Congrès Neptune, Toulon, France (2017)

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

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

Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. pp. 311–318. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA (Jul 2002), https://aclanthology.org/P02-1040

Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). pp. 2227–2237. Association for Computational Linguistics, New Orleans, Louisiana (Jun 2018)

Popović, M.: chrF: character n-gram F-score for automatic MT evaluation. In: Proceedings of the Tenth Workshop on Statistical Machine Translation. pp. 392–395. Association for Computational Linguistics, Lisbon, Portugal (Sep 2015), https://aclanthology.org/W15-3049

Radford, A., Narasimhan, K.: Improving language understanding by generative pre-training (2018)

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), http://jmlr.org/papers/v21/20-074.html

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://aclanthology.org/D13-1170

Song, K., Tan, X., Qin, T., Lu, J., Liu, T.Y.: Mass: Masked sequence to sequence pre-training for language generation. In: International Conference on Machine Learning. pp. 5926–5936 (2019)

Tiedemann, J.: Parallel data, tools and interfaces in opus. In: Chair), N.C.C., Choukri, K., Declerck, T., Dogan, M.U., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S. (szerk.) Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12). European Language Resources Association (ELRA), Istanbul, Turkey (may 2012)

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)

Váradi, T., Koeva, S., Yamalov, M., Tadić, M., Sass, B., Nitoń, B., Ogrodniczuk, M., Pęzik, P., Barbu Mititelu, V., Ion, R., Irimia, E., Mitrofan, M., Păiș, V., Tufiș, D., Garabík, R., Krek, S., Repar, A., Rihtar, M., Brank, J.: The MARCELL legislative corpus. In: Proceedings of the 12th Language Resources and Evaluation Conference. pp. 3761–3768. European Language Resources Association, Marseille, France (May 2020)

Yang, Z.G., Agócs, Á., Kusper, G., Váradi, T.: Abstractive text summarization for hungarian. Annales Mathematicae et Informaticae 53, 299–316 (2021)

Zhang, J., Zhao, Y., Saleh, M., Liu, P.: Pegasus: Pre-training with extracted gap-sentences forabstractive summarization. In: Thirty-seventh International Conference on Machine Learning (2020)

A. függelék: Példa az absztraktív modellek összegzéseiről

Eredeti cikk szövege:
Egyre többet hallani arról , hogy az okostelefon-gyártók olyan ujjlenyomat-olvasókkal kísérleteznek , amelyek be lennének építve a kijelzőbe . Korábban sokan úgy gondolták , hogy a Galaxy S8-aknál vagy legalábbis a Galaxy Note 8-nál jelentheti be az újdonságot a Samsung , de úgy tűnik , egyelőre még nem tudta ezt megoldani . A KGI neves elemzője , Ming-Chi Kuo szerint ez a funkció legkorábban a Galaxy Note 9-ben fog megjelenni , azaz még legalább egy évet várnunk kell rá . A szakember egyébként jól ismert pontos jövendöléseiről , bár elsősorban az Apple és nem a Samsung termékeire összpontosít . A Samsung már a Galaxy S8-nál üveg alá tette az ujjlenyomat-olvasót (aminek az elhelyezése amúgy nem váltott ki egyöntetű lelkesedést) , de a kijelző alá tenni azt , egyelőre még senkinek sem sikerült . Pedig ha ez megvalósulna , az sok lehetőséget nyithatna meg a gyártók előtt . Kuo szerint a Samsung már eldöntötte , hogy nem változtat az ujjlenyomat-olvasón a Galaxy Note 8-nál , viszont a Note 9-nél már valószínűleg az új megoldást mutatja be . Kuo azt is megszellőztette , hogy a dél-koreai óriás lecseréli biometrikus szkenner partnerét , a Synapticsot egy Egis nevű cégre .
Eredeti lead szövege:
Egy rég várt jellemző debütálását valószínűsítik a jövő évben megjelenő Galaxy Note 9-ben .
PreSumm (huBERT) által generált szöveg:
Ming-Chi Kuo , a KGI neves elemzője szerint a Samsung már a Galaxy Note 8-nál is bevezetheti az ujjlenyomat-olvasót , de egyelőre még nem tudni , mikor . A Samsung már eldöntötte , hogy nem változtat az ujjlenyomat-olvasón , viszont lecseréli a Synapticsot egy Egis nevű cégre . A szakember szerint ez sok lehetőséget nyithatna meg a gyártók előtt , ha megvalósulna az új funkció , akkor sok új lehetőséget nyitna meg a gyártó előtt . Ha ez megvalósulna , a Samsung lecseréli a Synaptitcsot egy Egisre .
BART-base-512 által generált szöveg:
A Ming-Chi Kuo szerint legkorábban egy év múlva jelenikhet meg a Samsung Galaxy Note 8 .
BART-base-1024 által generált szöveg:
A Samsung egyik neves elemzője , Ming-Chi Kuo szerint legalább egy évet kell várni rá , hogy a Galaxy S8-nál is megjelenhessen ujjlenyomat-olvasó .
BART-large által generált szöveg:
Ha igazak a híresztelések , jövőre egy új Samsung-csúcstelefon kerülhet a boltok polcaira .
BART-enhu által generált szöveg:
Az ismert Ming-Chi Kuo szerint a Samsung már eldöntötte , hogy nem változtat az ujjlenyomat-olvasón .
BART-enhu-transzfer által generált szöveg:
Ha hihetünk a pletykáknak , legkorábban a Galaxy Note 9-ben jelenik majd meg a Samsung új , ujjlenyomat-olvasós funkciója .

9. táblázat. Egy példa a modellek összegzéseiről.