A kis HIL-RoBERTa

Kivonat Kutatásunkban betanítottunk egy RoBERTa modellt magyar nyelvre. 2018-ban, a BERT megjelenésével, új fejezett nyílt a természetes nyelvfeldolgozás (NLP) történetében. A BERT, a kizárólag a figyelmi mechanizmusra épülő transzformátor modell viharos gyorsasággal tarolt és lett első nagyon sok angol benchmark feladatban. Sikerét növelte a transfer learning eljárás, melynek segítségével egy jól előtanított általános nyelvmodell csekély további ráfordítással finomhangolva még jobb eredményt lehet elérni a legtöbb nyelvtechnológiai kutatás területen. A BERT jelentős katalizátor hatás gyakorolt, és sorra születtek olyan modellek, arra tesznek kísérletet, hogy a BERT modell előtanítása során különböző módosításokkal még tovább növelhető a BERT teljesítménye. Eredményképpen több feladatban is megdöntötték a BERT eredményeit. Kutatásunkban magyar nyelvre tanítottuk be az egy ilyen "trónkövetelő" modellt, a RoBERTa modellt, majd megmértük, hogy magyar nyelvre is képes-e megdönteni a BERT modellek eredményeit.
Kulcsszavak: RoBERTa, BERT, névelem, chunking

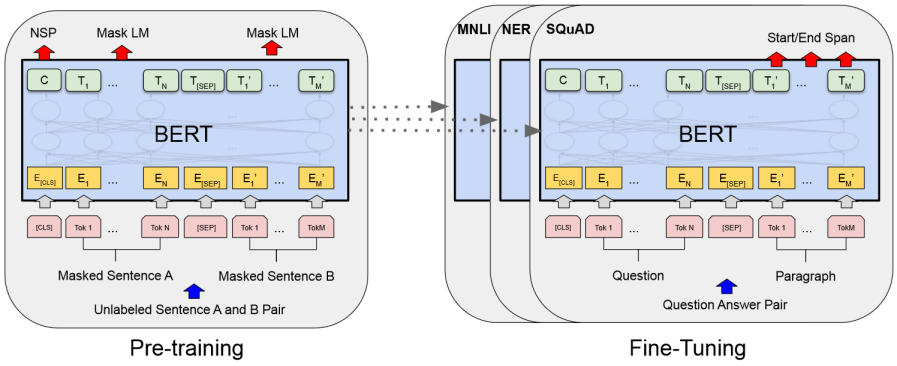
1. Bevezetés

Az elmúlt évtizedben a neurális hálón alapuló gépi tanulás jelentős eredményeket hozott a trmészetes nyelvfeldolgozás terén, ami az automatikus képfelismerés után a mesterséges intelligenciakutatás egyik sikertörténetévé vált. Ennek előfeltételez az volt, hogy a szavakat ne felszíni sztringekként kezeljük, hanem a szóbeágyazás (Mikolov és mtsai, 2013b,a) módszerével sokdimenziós, folytonos vektorokkal ábrázoljuk. A szóvektorok egy szemantikai teret képeznek, amelyben a hasonló jelentésű szavak közel állnak egymáshoz. A kezdetben javasolt szóvektorok egyik hátránya, hogy egy adott szóalak különböző jelentéseit is ugyanaz a vektor reprezentálja. Erre adnak megoldást a kontextuális beágyazáson alapuló modellek, mint az ELMo (Peters és mtsai, 2018) vagy a BERT (Devlin és mtsai, 2019) és annak származékai, pédául a RoBERTa (Liu és mtsai, 2019), ahol a szóvektorok tükrözik a szavak környezettől függően változó jelentését.

A BERT alapvetően megújította a neurális technológiával végzett NLP kutatásokat. A rendszer korántsem tökéletes, ezért többek között a RoBERTa szerzői arra tesznek kísérletet, hogy BERT előtanítási fázisában végezzenek olyan különböző módosításokat, amelyekkel a finomhangolás műveletét segítik és ezáltal jobb eredményt érjenek el a különböző nyelvtechnológiai feladatokban.

A BERT (Bidirectional Encoder Representations from Transformer) egy többszintű, kétirányú Transformer enkóder (Vaswani és mtsai, 2017). A BERT modellt két nyelvmodellezési feladaton tanítják elő (lásd 1. ábra): szómaszkolás és következő mondat predikálás. A maszkolás során betanító korpuszban szereplő

szavak 15%-át véletlenszerűen lemaszkolják, majd a feladat az, hogy rendszernek ki kell találnia a maszkolt szavakat. A mondat predikálás során a modell két mondatot kap, a feladat kitalálni, hogy a két mondat egymást követő, vagy két véletlenszerűen kiválaszotott mondat-e. A szótár méretének korlátozása érdekében a felszíni szóalakokat statisztikai alapon részekre bontják (wordpiece) tokenizáló (Schuster és Nakajima, 2012) segítségével. A BERT tanítása után az előtanított modellt finomhangolják egy-egy adott célfeladatnak megfelelően. A finomhangolás során egy előrecsatolt hálózattal a BERT modellt továbbtanítják a megadott feladatra.



1. ábra: BERT model

Magyar nyelvű önálló BERT modellt elsőnek Dávid Márk Nemeskey publikált (Nemeskey, 2020b) huBERT[1] néven. Három huBERT modell készült (időrendi sorrendben): huBERT wiki (cased és lowercase), a magyar Wikipédián tanított modell két változatban valamint a Magyar Webkorpusz 2.0-n[2] tanított huBERT base. A Magyar Webkorpuszon előtanított huBERT state-of-the-art eredményeket ért el névelemfelismerés és főnévi csoportok azonosítása feladatokban (Nemeskey, 2020a).

Tudomásunk szerint RoBERTa modell még nem készült magyar nyelvre.

2. A RoBERTa

A RoBERTa megismétli a BERT előtanítási eljárását, de közben az alábbi módosításokal próbálják fejleszteni a modell teljesítményét:

Nagyobb előtanító korpusz: A BERT előtanító korpusza az angol Wikipédia anyagából és a Google Book Corpusbó állt, és összesen 3,4 milliárd szót,

[1] https://hlt.bme.hu/en/resources/hubert
[2] https://hlt.bme.hu/en/resources/webcorpus2

16 Gb-nyi nyers szöveget tartalmazott. A RoBERTa betanításához a korpusz méretét tízszeresére növelték azaz 160 Gb-nyi, amelyet 5 különböző korpuszból állítottak össze, amely közel 30 milliárd szót tartalmaz.

Hosszabb modell tanítási lépés: Kísérleteztek 100 ezer, 300 ezer és 500 ezer lépéssel. Az eredményeik azt mutatták, hogy a lépésszám növelésével nő a rendszer teljesítménye.

Nagyobb batch méret: Kísérleteztek 256, 2000 és 8000 batch méreten. Az eredmény azt mutatta, hogy a batch méret növelésével javul az eredmény is. Legjobb eredményeiket 8000 batch méreten érték el. A lépésszám és a batch méret kölcsönösen összefügg egymással, a nagyobb batch méret mellett kevesebb lépésszám is elégséges ugyanahhoz a teljesítményhez. Ezáltal rövidül ugyan a modell előtanításának az ideje, egyben növekszik a szükséges memória igény. Természetesen a lépésszám és a batchméret egyidejű növelése hatványozottan növeli a rendszer teljesítményét, és a RoBERTa szerzői mindkét paramétert együttesen növelték.

A következő mondat prediktálás (NSP) feladat eltörlése: A szerzők kisérleti úton azt találták, hogy az NSP feladat nem járul lényegesen hozzá a rendszer tanulásához, ezért azt elhagyták.

Hosszabb bemeneti szövegek A RoBERTa maximálisan kihasználja az 512-es szekvenciahosszt. A mondatokat nem egyesével tölti be a szekvenciákba és az 512 karakterig fennmaradó részt üres karakterekkel <PAD> tölti ki, hanem egyszerre több mondatot olvas be, mindaddig, amíg meg nem telik az 512 karakter hosszú szekvencia. Még a dokumentum vége sem jelent új szekvencia nyitást, ekkor egy dokumentum szeparaátor karakter beszúrása után folytatódik az input szövegbeolvasás.

Dinamikus maszkolás: Fontos újítás az, hogy míg a BERT statikus maszkolást alkalmazott, azaz a szöveg előfeldolgozási lépéseként maszkolta a szavak 15%-át, amelyek aztán az előtanítás során mindig azonosak maradtak, a RoBERTa dinamikus szómaszkolást alkalmaz, vagyis a szómaszkolási mintát minden alkalommal újból elóállítja mielőtt a szekvenciát a rendszernek betölti.

BPE kódolás: A RoBERTa a szavak belső reprezentációját a Byte Per Encoding (BPE) (lec Radford és mtsai, 2019) módszerrel kódolja, amely a szó és a karakter reprezentáció egyfajta hibrid megoldása. Karakter ngram-ok iteratív egyesítéséből születnek a szóelemek, amelyeket innovatív módon nem az unikód karakterek, hanem a bájtok képzik alapját. Ez sokkal takarékosabb megoldás, és a BPE kódolás jóval kevesebb ismeretlen szót produkál, mint a BERT-ben használt WordPiece (Schuster és Nakajima, 2012) rendszer.

3. A modell előtanítása

3.1. A betanító korpusz

A modell előtanításához a magyar Wikipédia szövegét használtuk (13.098.808 szegmens; 163.772.783 token), amelyet a Magyar Webkorpusz 2.0-ből[3] töltöttünk

[3] https://hlt.bme.hu/en/resources/webcorpus2

le. A betanító szkript a Huggingface által ajánlott minta alapján készült. A szkriptet egy ponton kellett lényegesen módosítanunk: az adatok bináris alakra konvertálása memóriahiány miatt nem volt lehetséges az eredeti kódban javasolt LineByLineTextDataset eljárást alkalmazni.

3.2. A hardver és a betanítás menete

A betanítást 4 db. Nvidia GTX 1080Ti GPU kártyát tartalmazó rendszeren végeztük a 4 GPU-n párhuzamosan. Az egyes GPU kártyáknak 11 GB memóriája volt, összesen tehát 44 GB állt rendelkezésre az előtanításhoz. A kártyánkénti 8 azaz összesen 32-es batchméret mellett 214 órába telt a modell előtanítása. A 2. ábra mutatja a előtanítás menetét jellemző loss görbe alakulását. Az 1 250 000 lépésből álló tanítás végén a loss görbe a kezdeti 8.7-es értékről csökkenve 2.5 körül stabilizálódott.

Az előtanításhoz[4] az alábbi hyper-paramétereket használtuk:

- learning rate: 1e-4
- train epochs: 5
- save total limit 2
- save steps: 2000
- train batch méret: 8
- evaluate during training
- seed 42

4. A modell finomhangolása

Az elkészített modell finomhangolása egy gyakran használt transzfer tanítás módszer. Ezzel a felügyelt tanítási módszerrel specifikus feladatokra lehet tovább tanítani a modellt, mint az entitásfelismerés vagy a kontextusalapú kérdésválasz. Működését tekintve az előtanított modell „legvégére" egy klasszifikációs réteg kerül (Devlin és mtsai, 2019; Liu és mtsai, 2019), ami így a tovább tanítás során a bemenetet és annotációit tanulja meg. Különbség a BERT modellekhez képest a tanítás során, hogy a mondatok között plusz szeparátor tokent használ a RoBERTa, „<s></s>" formájában.

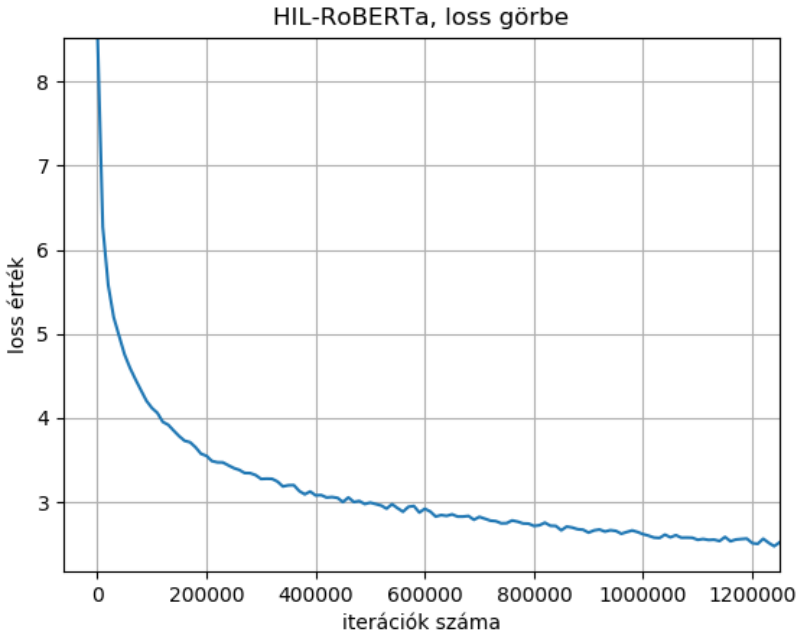
Modellünk teljesítményének méréséhez két feladatra finomhangoltuk: névelemfelismerés (NER) és főnévi csoport felismerés (NP).

A NER feladat tanításhoz a szakmában és itthon gyakran használt szegedi Corpus of Business Newswire Texts (Szeged NER) korpuszt használtuk, ami Hungarian Named Entity Corpora része (Szarvas és mtsai, 2006). Az NP feladathoz a Szeged Treebank 2.0 (Csendes és mtsai, 2005) korpuszt.

A finomhangoláshoz használt korpuszok méretei:

- NER (mondat): Tanító: 8.484; Validáció: 514; Teszt: 932

[4] https://huggingface.co/blog/how-to-train



2. ábra: HIL-RoBERTa model

– NP (mondat): Tanító: 65.679; Validáció: 8.209; Teszt: 8.209

A NER és az NP esetében a finomhangolást a transformers oldalán[5] található kódot használtuk.

A NER és NP modellek kiértékeléséhez az IOB alapú seqeval (Nakayama, 2018) metódust használtuk.

A finomhangolás hyper-paraméterei a legjobb eredményt elért modelleknél:

– learning rate: 1e-4
– training, eval és predict batch méret (per GPU): 2 (4 db GPU-n)
– seed: 42
– max sequence length: 512

A finomhangolást 4 darab GeForce RTX 2080 Ti típusú videókártyán tanítottuk. A finomhangolás NER esetében körülbelül 20 percet vett igénybe (4 epoch), az NP esetében körülbelül 2 óra (4 epoch).

[5] https://github.com/huggingface/transformers/tree/master/examples/token-classification

5. Eredmények

Az 1. és a 2 táblázatban láthatóak a NER és a NP kísérletek eredményei. Kétféle mérést végeztünk: 4 epoch és 30 epoch. Az emBERT (Nemeskey, 2020a) kísérletekben a legjobb NER eredményeket 30 epoch mellett érték el, ezért az összehasonlíthatóság végett mi is lefuttattuk 30 epoch-al a finomhangolásainkat. A multi-BERT, huBERT wiki és huBERT modellek esetén a 4 epoch mérés eredményeit magunk reprodukáltuk az emBERT kísérlet kódja[6] alapján.

	NER 4 epoch	NER 30 epoch
multi-BERT	96,30%	97,08%
huBERT wiki	96,63%	97,03%
huBERT	97,51%	97.62%
HIL-ELECTRA small wiki	91,32%	91,73%
HIL-RoBERTa	90,98%	92,78%

1. táblázat. NER kísérletek eredményei

	NP
multi-BERT	95,05%
huBERT wiki	96,41%
huBERT	96,97%
ELECTRA small wiki	94,14%
HIL-RoBERTa	89,48%

2. táblázat. NP kísérletek eredményei

A NER és NP kísérletek eredményei azt mutatják, hogy a RoBERTa modellünk nem éri el a BERT modellek teljesítményét. Ezt nem is vártuk tőle, hiszen ehhez sokkal nagyobb tanító korpuszra lenne szükség. Azonban a sokkal kisebb korpuszon tanítva is sikerült vele a BERT modellekhez hasonló magas eredményt elérni. Sőt NER esetében jobban teljesít az ugyancsak magyar Wikipédián tanított ELECTRA modellnél.

6. Összegzés

Kutatásunkban betanítottunk egy RoBERTa modellt magyar nyelvre és megmértük teljesítményét névelemfelismerés és főnévi csoportok felismerés feladatok révén. A modell tanításhoz a magyar Wikipédiát használtuk. Ez nyilvánvalóan messze elmarad a RoBERTa modell tanításához megkívánt mennyiségű adattól.

[6] https://github.com/DavidNemeskey/emBERT

Mégis úgy gondoltuk, hogy a RoBERTa modell egyéb innovatív vonásai, leginkább a dinamikus szómaszkolás érvényesülni tudnak kisebb tanító adat esetében is. Az eredményeink igazolják a várakozásainkat. Bár ez a kis RoBERTa modell, nem tudja megdönteni a BERT modellek eredményeit, azonban így is, hogy sokkal kisebb méretű korpuszon tanítottuk, közel olyan magas eredményt ért el, mint a BERT modellek. Névelemfelismerésben a HIL-RoBERTa teljesítménye jobb is, mint az ugyancsak magyar Wikipédián tanított ELECTRA modellé.

Tudomásunk szerint, ez az első magyar nyelvű RoBERTa modell.

Hivatkozások

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)

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized BERT pretraining approach (2019)

Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. CoRR abs/1301.3781 (2013a)

Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States. pp. 3111–3119 (2013b)

Nakayama, H.: seqeval: A python framework for sequence labeling evaluation (2018), https://github.com/chakki-works/seqeval, software available from https://github.com/chakki-works/seqeval

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)

Peters, M., 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)

lec Radford, Wu, J., Child, R., Luan, D., Amodeia, D., Sutskever, I.: Languagemodels are unsupervised multitask learners. Tech. rep., OpenAI (2019)

Schuster, M., Nakajima, K.: Japanese and korean voice search. In: ICASSP. pp. 5149–5152. IEEE (2012)

Szarvas, G., Farkas, R., Felföldi, L., Kocsor, A., Csirik, J.: A highly accurate named entity corpus for Hungarian. In: Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06). Genoa, Italy (May 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)