A kis HIL-ELECTRA

Kivonat Kutatásunkban különböző ELECTRA small modellekkel kísérleteztünk. A BERT modellek tanítása sok időt és erőforrást igényel. Ezért a Google kifejlesztett egy olyan modellt, amely kis erőforással kevés idő alatt betanítható és hasonló eredményeket lehet vele elérni, mint a hagyományos BERT modellek. A kutatásunkban az ELECTRA small modelleket céloztuk meg. Betanítottunk három magyar nyelvű ELECTRA small modellt, majd finomhangoltuk három különböző feladatra: névelemfelismerés, főnévi csoportfelismerés és extraktív összefoglaló generálás. Az eredmények azt mutatják, hogy az ELECTRA small modellek nem tudják ugyan felülmúlni a legjobban teljesítő hagyományos BERT modelleket, de közel olyan magas eredményt érnek el, miközben betanításuk kevesebb erőforrást igényel. Továbbá, a felhasználó szempontjából is előnyös az ELECTRA small modellek, hiszen hasonló teljesítmény mellett sokkal kisebb méretű modellekkel dolgozik, ami igen fontos szempont manapság.
Kulcsszavak: ELECTRA, BERT, névelem, chunking, extraktív összefoglaló

1. Bevezetés

Az elmúlt években jelentős eredmények születtek az általános célú nyelvi modellek építésében. Az első áttörést a szóbeágyazás (Mikolov és mtsai, 2013b,a) módszere hozta meg, amely a szavakhoz sokdimenziós, folytonos vektort rendel. A szavak vektor-reprezentációi egy szamantikai teret képeznek, amelyben a hasonló jelentésű szavak közel állnak egymáshoz. A szóvektorok a szemantikai tarlamuk mellett szintaktikai tulajdonságokat is megtanulnak. A módszer egyik hátránya, hogy egy adott szóalak különböző jelentéseit is ugyanaz a vektor reprezentálja. Erre adnak megoldást a kontexuális 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 neurális hálós nyelvi modellek építésének egy komoly korlátja az, hogy például a BERT modell tanítása rendkívül nagy erőforrást igényel adatokban és számítási kapacitásban is: az angol nyelvű BERT base tanítása 4 Cloud TPU-n 4 napon keresztül folyt.

Ezek korlátok enyhítésére a Google az idén kifejlesztette az ELECTRA modellt (Clark és mtsai, 2020), amely kis erőforással (1 GPU), kevés idő alatt betanítható és hasonló eredményeket lehet vele elérni, mint a hagyományos BERT modellek. Végül, de nem utolsó sorban, a betanított modellek mérete is sokkal kisebb, ami igen fontos a mai „mobilos” világban.

2. Kapcsolódó irodalom

A BERT (Bidirectional Encoder Representations from Transformer) egy többszintű, kétirányú Transformer enkóder (Vaswani és mtsai, 2017). A BERT modellt két nyelvmodellezési feladaton tanítják elő: 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.

A BERT egyik előnye, hogy nem csak angol nyelvre tanítottak be modelleket. A Google betanított két többnyelvű modellt[1]: kisbetűsített és nem kisbetűsített. A modellek tanításához kiválasztották az első 104 nyelvet, amely a legnagyobb Wikipédiával rendelkezik. A egyes nyelvek Wikipédia mérete igen különbözik, az adat közel 20%-át teszi ki az angol Wikipédia, ezért normalizálással kontrollálták a mintavételezést, hogy kiküszöbölják ezt a problémát. Ezután minden nyelvet, hasonlóan az angolhoz, tokenizálásnak vetették alá, amelynek négy lépése volt: kisbetűsítés, ékezetek eltávolítása, írásjelek leválasztása, whitespacek kezelése. A nem kisbetűsített modell tanítása is ezeken a lépéseken esett át, a WordPiece szótár segítségével kezelik a nem kisbetűs és ékezetes szavakat. Természetesen a magyar nyelv is egyike a modell által lefedett 104 nyelvnek.

Magyar nyelvű önálló BERT modellt elsőnek Dávid Márk Nemeskey publikált (Nemeskey, 2020b) huBERT[2] néven. Három huBERT modell készült:

- huBERT: Magyar Webkorpusz 2.0-n[3] tanított BERT base
- huBERt Wikipedia cased: Magyar Wikipédián tanított nem kisbetűsített BERT base
- huBERt Wikipedia lowercased: Magyar Wikipédián tanított kisbetűsített BERT base

A Magyar Webkorpuszon előtanított huBERT state-of-the-art eredményeket ért el névelemfelismerés és főnévi csoportok azonosítása feladatokban(Nemeskey, 2020a).

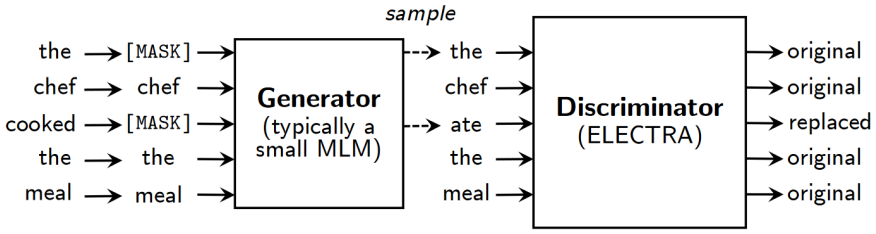
3. A ELECTRA

Az ELECTRA (Clark és mtsai, 2020) a GAN (Generative adversarial network) (Goodfellow és mtsai, 2014) módszeren alapszik. A módszer alapja (lásd 1. ábra), hogy

[1] https://github.com/google-research/bert/blob/master/multilingual.md
[2] https://hlt.bme.hu/en/resources/hubert
[3] https://hlt.bme.hu/en/resources/webcorpus2

két hálózatot tanítanak be, egy generátort és egy diszkriminátort. A tanítás során a generátor véletlenszerűen generál vektor reprezentációkat, amelyekből kimenetet generál. Majd mutatnak neki igazi kimenetet, ami alapján javít a véletlenszerű vektor generáláson. Ily módon, a tanítás végére a generátor egyre "okosabb" lesz és olyan kimenetet tud generálni, amely tényleg hasonlít az igazi kimenetre. Eközben a diszkriminátort arra tanítják, hogy egy adott adatsorozatra megmondja, hogy az igazi adatsor, vagy hamis adatsor. Ezt úgy érik el, hogy adnak igazi adatsorokat a tanítókorpuszból és olyan adatsorokat is, amelyeket a generátor állított elő. A BERT szómaszkolási feladatától eltérően, itt nem azt kell a rendszernek kitalálnia, hogy mi volt a lefedett eredeti szó, hanem azt kell eldöntenie egy adott szóról, hogy eredeti vagy egy felcserélt szó-e? Ezt azonban minden szóról meg kell állapítania, nem csak véletlenszerűen kiválasztot 15%-nyi szóról. A két hálózat össze van kötve, így a tanítás során kölcsönösen erősítik egymást.

Az ELECTRA a GAN módszert alakítja át nyelvmodell tanítására (lásd 1). A különbség a BERT modellhez képest, hogy itt a hálózat nem a maszkolt szavakat próbálja megjósolni, hanem a generátor a maszkolt szavakra véletlenszerűen generál szavakat, majd a diszkriminátor csak azt tanulja meg, hogy a generátor által adott szavak eredeti szavak, vagy véletlenszerúen generált szavak. Így a generátor lassan megtanulja, hogy a maszkolt szavak helyére milyen szavak illenek, míg a diszkriminátor azt tanulja meg, hogy egy adott szövegbemenet szavai tényleg jók-e. A tanítás után a generátort elvetik és csak a diszkriminátort tartják meg a finomhangoláshoz.



1. ábra: ELECTRA model

A Google 3 különböző ELECTRA modellt fejlesztett ki:

- ELECTRA small: 12 réteg; rejtett réteg mérete: 256; 14M paraméter;
- ELECTRA base: 12 réteg; rejtett réteg mérete: 768; 110M paraméter;
- ELECTRA large: 24 réteg; rejtett réteg mérete: 1024; 335M paraméter;

Az ELECTRA small igényli a legkevesebb erőforrást, ezért kutatásunkban csak az ELECTRA small modellekkel kísérleteztünk.

4. Korpuszok

Az ELECTRA modellek tanításához három különböző korpusszal kísérleteztünk.

- Magyar Wikipedia (wiki): 13.098.808 szegmens; 163.772.783 token;
- NYTI-BERT korpusz v1 (nyti): 283.099.534 szegmens; 3.993.873.992 token;
- NYTI-BERT korpusz + Magyar Wikipedia (merge)

A modell tanításhoz használt szótárméret: 64.000.

Az összevethetőség végett a finomhangolásokhoz ugyanazokat a korpuszokat használtuk, mint az emBERT (Nemeskey, 2020a) és a magyar összefoglaló generálás (Yang és mtsai, 2020) kísérleti.

A névelemfelismerés (NER) finomhangoláshoz a Szeged NER korpuszt (Szarvas és mtsai, 2006), a főnévi csoportok felismeréséhez (NP) a Szeged Treebank 2.0 (Csendes és mtsai, 2005) korpuszt, míg az extraktív összefoglaló generáláshoz (SUM) az online HVG (2012-2020) cikkek és hozzájuk tartozó leadeket használtuk.

A korpuszok méretei:

- NER (mondat): Tanító: 8.484; Validáció: 514; Teszt: 932
- NP (mondat): Tanító: 65.679; Validáció: 8.209; Teszt: 8.209
- SUM (cikk+lead): Tanító: 472.660; Validáció: 5.000; Teszt: 3.000;

5. Kísérletek

Az ELECTRA (Clark és mtsai, 2020) modellek tanításához a Google által implementált kódot használtuk[4]. Az alábbi három modellt tanítottuk:

- ELECTRA small wiki: Magyar wikipédián tanított, körülbelül 5 nap alatt futott le a tanítás.
- ELECTRA small nyti: NyTI v1 korpuszon tanított, körülbelül 7 nap alatt futott le a tanítás.
- ELECTRA small merge: NyTI v1 koprusz + Magyar Wikipedia korpuszon tanított, körülbelül 7 nap alatt futott le a tanítás.

Mindegyik modellt 1 darab GeForce RTX 2080 Ti típusú videókártyán tanítottuk. A futási időt befolyásolja a szótárméret is, kisebb szótárral gyorsítható.

A tanításhoz az alapértelmezett paramétereket (learning rate = 5e-4; weight decay rate = 0.01; embedding size: 128; 1 millió tanítási lépés;) használtuk, egyedül a batch méretet vettük le 80-ra, mivel nagyobb értéken túllépte a CUDA memória méretét.

A modellek teszteléséhez három különböző kísérletet végeztünk:

- Névelemfelismerés (NER)
- Maximális főnévi csoport felismerés (NP)
- Extraktív összefoglaló generálás (SUM)

[4] https://github.com/google-research/electra

A NER és az NP esetében a finomhangolást a Google által fejlesztett kódot[5] használtuk, az alábbi paraméterekkel: 4 epoch; learning rate = 0,001; weight decay rate = 0,01.

Az extraktív modellek tanításához a PreSumm[6] eszközt használtuk a megadott alapértelmezett paraméterekkel.

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

Az extraktív modellek kiértékeléséhez a ROUGE (Lin, 2004) fedés metrikát használtuk. A ROUGE (Recall-Oriented Understudy for Gisting Evaluation) egy fedés alapú módszer, ami a gépi fordítás során használt BLEU metrikán alapszik. Maga a ROUGE több almetódust is tartalmaz, melyek közül a méréseinkhez a ROUGE-1, ROUGE-2 és a ROUGE-L módszereket használtuk. A ROUGE-1 egy unigram, míg a ROUGE-2 egy bigram fedést számoló algoritmus. A ROUGE-L a leghosszabb közös szósorozatot vizsgálja bekezdés és mondat szinten.

6. Eredmények

Az 1. táblázatban látható a NER és az NP kísérletek eredményei. Az ELECTRA, a finomhangolás során, a súlyok inicializáláshoz csonka normális eloszlást használ, ezért minden tanítás, bár kicsi különbséggel, de más-más teljesítményt ért el. Ezért az eredményekben látható egy átlag és a legjobb érték. Összesen 10 mérés átlaga látható. A finomhangolás eredményét továbbá befolyásolja a batch size is, az eredményekben látható értékek 12-es batch méreten értük el.

	NER (%)	NP (%)
multi-BERT	97,08	95,05
huBERT wiki	97,03	96,41
huBERT web	97.62	96,97
ELECTRA small wiki átlag	90,40	93,13
ELECTRA small wiki legjobb	91,32	94,14
ELECTRA small nyti átlag	90,13	94,18
ELECTRA small nyti legjobb	90,51	94,20
ELECTRA small merge átlag	91,56	93,64
ELECTRA small merge legjobb	96,86	93,75

1. táblázat. NER, NP kísérletek eredményei

Az 1. táblázatban látható, hogy az ELECTRA small modellek egyik esetben sem éri el a BERT modellek teljesítményét. Ez várható is volt, hiszen az ELECTRA small modellek kevesebb paraméterrel rendelkeznek. Azonban látható a NER esetében, hogy a legjobb esetben 96,86%-ot ért el, ami majdnem eléri a BERT modellek teljesítményét, de ez egyszeri eset, a kezdeti súlyok szerencsés

[5] https://github.com/google-research/electra

[6] https://github.com/nlpyang/PreSumm

inicializálásnak köszönhetően. Az NP mérések esetében látható, hogy legjobb esetben az ELECTRA small merge modell csupán 2-3%-al marad csak el a magyar egynyelvű BERT modellektől.

A 2. táblázatban látható az extraktív modellekkel való kísérletek eredményei. Ahogy vártuk, egyik ELECTRA small modell sem múlja felül a huBERT state-of-the-art teljesítményét. Ami érdekes viszont, hogy az ELECTRA small modelljeink mind jobban teljesítenek, mint a huBERT wiki és a multi-BERT.

	ROUGE-1	ROUGE-2	ROUGE-L
multi-BERT	48.58	20.12	39.42
huBERT wiki	48.86	20.45	39.60
huBERT	49.45	21.07	40.14
ELECTRA small wiki	49.02	20.52	39.74
ELECTRA small nyti	49.04	20.53	39.76
ELECTRA small merge	49.01	20.50	39.72

2. táblázat. Extraktív összefoglaló ROUGE fedés eredmények

Bár az esetek többségében nem múlja felül a hagyományos BERT modellek értékeit, azonban figyelembe kell vennünk azt, hogy az ELECTRA small modellek kevesebb paraméterrel ér el hasonlóan jó eredményt, és tanításához csupán 1 GPU elegendő és még nagy szótárral is körülbelül 7 nap alatt betanítható. Továbbá, a felhasználó szempontjából is előnyös az ELECTRA small modellek, hiszen hasonló teljesítmény mellett sokkal kisebb méretű modellekkel dolgozik, ami igen fontos szempont.

7. Összegzés

Kutatásunkban magyar nyelvű ELECTRA small modellekkel kísérletezünk. Az ELECTRA modell legnagyobb előnye, hogy kis erőforráson, kevés idő alatt betanítható és ugyanakkor majdnem olyan magas eredményt lehet vele elérni mint a hagyományos BERT modellekkel.

Kísérleteink során betanítottunk három különböző ELECTRA small modellt. A tanításokhoz a Magyar Wikipédia és a Nyelvtudományi Intézet korpuszát használtuk fel. A modelljeinket leteszteltük a magyar névelemfelimerés, főnévi csoport felismerés és extrakív összefogalaló generálás feladatain.

Az eredmények igazolták a szakirodalomra alapozott várakozásainkat: már az ELECTRA small modul is megközelíti a stat-of-the-art eredményeket, sőt extraktív összefoglaló generálás területén felülmúlja a Magyar Wikipédián tanított huBERT modellt. A legnagyobb eredmény mégis az, hogy mindezeket az eredményeket úgy éri el, hogy összesen 1 GPU-n tanítottuk az ELECTRA modelleket, és a leggyorsabb esetben mindössze 5 nap elegendő volt a betanításra (ez az idő a szótárméret csökkentésével tovább gyorsítható). Továbbá, a felhasználó szempontjából is előnyös az ELECTRA small modellek, hiszen hasonló teljesítmény mellett sokkal kisebb méretű modellekkel dolgozik, ami igen fontos szempont.

A továbblépési lehetőségeink között szerepel az ELECTRA base illetve large modellek való kísérletezés.

Hivatkozások

Clark, K., Luong, M.T., Le, Q.V., Manning, C.D.: ELECTRA: Pre-training text encoders as discriminators rather than generators. In: ICLR (2020)

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)

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., Weinberger, K.Q. (szerk.) Advances in Neural Information Processing Systems. vol. 27, pp. 2672–2680. Curran Associates, Inc. (2014), https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

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., 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)

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

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)

Yang, Z.G., Perlaki, A., Laki, L.J.: Automatikus összefoglaló generálás magyar nyelvre bert modellel. In: XVI. Magyar Számítógépes Nyelvészeti Konferencia. pp. 343–353. Szegedi Tudományegyetem, Szeged (2020)