An evaluation of BERT and Doc2Vec model on the IPTC Subject Codes prediction dataset

Marko Pranjić
marko.pranjic@styria.ai
Jožef Stefan International Postgraduate School
Jamova cesta 39
Ljubljana, Slovenia
Trikoder d.o.o.
Zagreb, Croatia

Marko Robnik-Šikonja
marko.robnik@fri.uni-lj.si
University of Ljubljana, Faculty of Computer and Information Science
Ljubljana, Slovenia

Senja Pollak
senja.pollak@ijs.si
Jožef Stefan Institute
Jamova cesta 39
Ljubljana, Slovenia

ABSTRACT

Large pretrained language models like BERT have shown excellent generalization properties and have advanced the state of the art on various NLP tasks. In this paper we evaluate Finnish BERT (FinBERT) model on the IPTC Subject Codes prediction task. We compare it to a simpler Doc2Vec model used as a baseline. Due to hierarchical nature of IPTC Subject Codes, we also evaluate the effect of encoding the hierarchy in the network layer topology. Contrary to our expectations, a simpler baseline Doc2Vec model clearly outperforms the more complex FinBERT model and our attempts to encode hierarchy in a prediction network do not yield systematic improvement.

KEYWORDS

news categorization, text representation, BERT, Doc2Vec, IPTC Subject Codes

1 INTRODUCTION

The field of Natural Language Processing (NLP) has greatly benefited from the advances in deep learning. New techniques and architectures are developed at a fast pace. The Transformer architecture [12] is the foundation for most new NLP models and it is especially successful with models for text representation, such as BERT model [1] which dominates the text classification. The gains in performance promised by the large BERT models comes at the price of significant data resources and computational capabilities required in the model pretraining phase. The practitioners take one of the models pretrained in the language of the data and finetune it for the specific classification problem. Multilingual BERT-like models have also shown remarkable potential for cross-lingual transfer ([7], [8], [6]). A majority of the research with BERT-like models is focused on English, while less-resourced languages tend to be neglected.

The IPTC Subject Codes originate in the journalistic setting. The news articles are tagged with the IPTC topics to enable search and classification of the news content, as well as to facilitate content storage and digital asset management of news content at media houses. It provides a consistent and language agnostic coding of topics across different news providers and across time. Solving the automatic classification of the news content to the standardized set of topics would enable faster news production and higher quality of the metadata for news content.

In this paper, we use recently published STT News[10] dataset in Finnish to evaluate the performance of the monolingual FinBERT model [13] on the IPTC Subject Codes prediction task, together with the Doc2Vec[3] model as a baseline. We attempt to encode the hierarchical nature of the prediction task in the prediction network topology by mimicking the structure of the labels. Finally, impact of using a different tokenizers with the same model is evaluated.

The paper is structured as follows. In Section 2, we describe the dataset and the labels relevant for the prediction task. Section 3 describes the methods used to model the prediction task and all variations of experiments. In Section 4, we provide results of our experiments and, finally, in Section 5 we conclude this paper and suggest ideas for further work.

2 DATASET

The STT corpus [10] contains 2.8 million news articles from the Finnish News Agency (STT) published between 1992 and 2018. The articles come with a rich metadata information including the news article topics encoded as IPTC Subject Codes[1]. The IPTC Subject Codes are a deprecated version of IPTC taxonomy of news topics focused on text. The IPTC Subject Codes standard describes around 1400 topics structured in three hierarchical levels. The first level consists of the most general topics. Topics on the second level are subtopics of the ones at the first level and, likewise, topics on the third level are subtopics of the ones on second level. All topics on the third level are leaf topics - there are no more subdivisions, but there are also some topics on the second level that are leaf topics and do not extend to the third level. A set of IPTC topics at STT is an extended version of IPTC Subject Codes as some codes used at STT are not part of the IPTC standard.

Not all articles in the STT corpus contain the IPTC Subject Codes, as can be seen in Figure 1, showing the ratio of articles containing this information through time. IPTC Subject Codes were introduced in STT in May 2011 and around 10-15% of articles do not contain this information.

If an article contains a specific sub-topic, it also contains its upper-level topics. For example, if an article contains the third level topic "poetry", it also contains the second level topic "literature" that generalizes the "poetry", as well as the first level topic "arts, culture and entertainment". In this way, article metadata contains full path through the topic hierarchy.

Information Society 2021, 4–8 October 2021, Ljubljana, Slovenia

[1] https://iptc.org/standards/subject-codes/

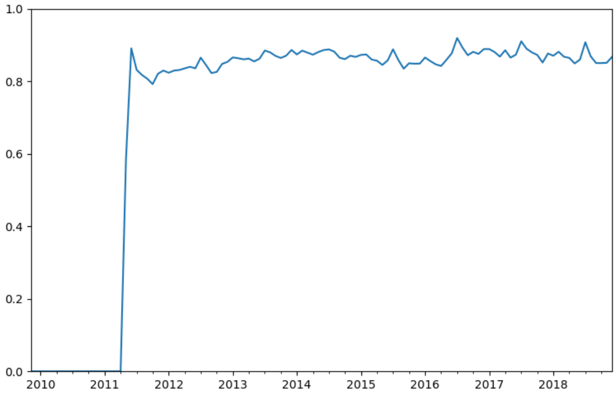

Figure 1: The Ratio of news articles in STT corpus containing IPTC Subject Codes.

Most articles are assigned only a small number of leaf-level topics (and its higher-level topics), but they can contain up to 7, 19 and 30 topics from the first, second and third level, respectively.

We split the dataset to train, validation and test set such that all articles published after 31-12-2017 belong to the test set and discard articles without IPTC Subject Codes from it. The rest of the articles were randomly split such that 5% of articles containing IPTC Subject Codes represent the validation set and all other articles belong to the train set.

After this step, there are around 30 thousand articles in the validation set, around 100 thousand in test set and 2.7 million in training set - of which some 560 thousand contain IPTC Subject Codes annotation.

The train set contains 17 different topics on the first level, 400 on the second level, and 972 on the third (the most specific) level. In our experiments, we evaluate models only on topics found in the training set.

3 METHODOLOGY

For our experiments, we used a network design consisting of two stacked neural networks (extractor and predictor). The extractor processes the text and produces the text representation in the format of a numeric vector. The predictor (the second part) is a multi-label prediction network that maps the extracted text representation vector to IPTC Subject Codes. For the extractor part, we evaluate the Doc2Vec and BERT model and for the predictor our models use one or three layer neural network.

3.1 Doc2Vec

Before the contextual token embeddings became popular, this model was regularly used to represent a text paragraph with a fixed vector. It was introduced in [3] with two variants of the algorithm - PV-DM (Paragraph Vector-Distributed Memory) and PV-CBOW (Paragraph Vector-Continuous Bag-of-Words). In the PV-DM variant of the algorithm, a training context is defined as a sliding window over the text. The model is a shallow neural network trained to predict the central word of this context window given the embeddings of the rest of the context words together with the embedding of the whole document. During training, the network learns both the word embeddings and the embedding for the document. The simpler PV-CBOW variant does not employ a context window, the neural network is trained to predict a randomly sampled word from the document. Our experiments use the PV-DM variant of the algorithm available in the Gensim[2] library with most of the hyperparameters set to their default values. We set the context window width to 5 and train the network for 10 epochs on the news content from the training data. The model produces a 256 dimensional output vector. Once the model is trained, we do not finetune it further during training of the prediction task.

Tokenization of the data was done using the SentencePiece[2] tokenizer. It was trained to produce a vocabulary of 40,000 tokens by using randomly selected 1 million sentences sampled from the articles in the training set. Additionally, we ran experiments using the same WordPiece[14] tokenizer that is used with the FinBERT model.

3.2 BERT

BERT is an deep neural-network architecture of bidirectional text encoders introduced in [1]. The base model consists of 12 Transformer [12] layers. It is trained using the masked language modeling (MLM) and next sentence prediction (NSP) objectives on a large text corpora. Maximum length of the input sequence for the model is 512 tokens and each token is represented with 768 dimensions. Model inference produces a context dependent representations of the input tokens. The whole input sequence can be represented with a single vector by using the context dependent representation of the *[CLS]* token. In [1], this representation is used as an aggregate sequence representation for classification tasks. Another way to represent the whole sequence, as used in [9], is to take the average representation of all output tokens (AVG). In this paper, we use FinBERT, a BERT model introduced in [13] that was pretrained on Finnish corpora.[3] We should note that this model contains the STT corpus as part of its training data.

Input to the model is restricted to 512 tokens[4] and longer news articles are trimmed such that only the first 512 tokens are used. In the dataset, there are less than 5% and 7% of documents in the training and test data that are longer than 512 tokens. We experiment with the CLS and the AVG representations and in both cases the article representation is a 768 dimensional vector. The FinBERT model is finetuned during training of the IPTC Subject Codes prediction task.

3.3 Prediction network

For the predictor part, we experiment with two different architectures. The first is a single layer of the neural network that maps the input vector to the predictions and can be seen in the Figure 2. The IPTC Subject Codes on all levels are concatenated together, thus producing a 1389 outputs in the final layer.

The second architecture utilizes the tree hierarchy of the IPTC Subject Codes. We assumed that a flat output (the previous approach) requires the network to predict each label independently, irrespective of the level of the target label. By introducing separate layers for each target level, we expect that the model will implicitly learn the hierarchy among labels. We designed this network in three layers and the architecture is shown in Figure 3. The first layer of the network predicts labels from the third IPTC hierarchical level (the most fine-grained topics), the second layer

[2] https://radimrehurek.com/gensim/

[3] We also test the FinEst BERT[11] but since the better performance was achieved with the FinBERT[13], we do not include FinEst BERT it in the results.

[4] The tokenizer used with the model is a predefined WordPiece tokenizer that came with the FinBERT model.

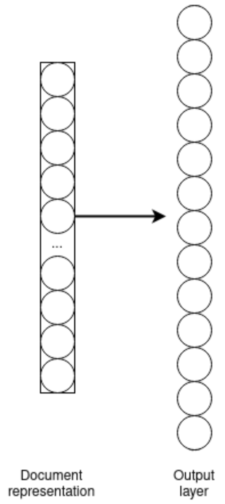

Figure 2: Predictor network architecture, flat variant. The image does not show a normalization layer before the output layer.

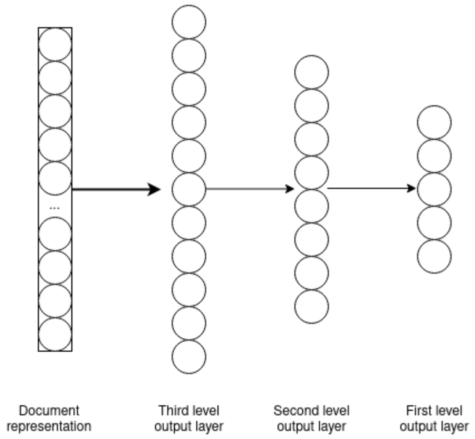

Figure 3: Predictor network architecture, tree variant. The image does not show a normalization layer before each output layer.

predicts topics from the second level and the third layer predicts only the toplevel IPTC Subject Codes.

3.4 Training

Each model was trained using the batch size of 128 articles and AdamW[4] optimizer with the learning rate of 1e-3. We compute the metrics on the validation set every 100 iterations. Once the loss on the validation data starts increasing, we stop the training and evaluate the best performing checkpoint on the test data. The loss function used in all experiments is the sum of binary cross-entropy losses calculated at each topic level. The news articles that do not have an annotation for certain topic level do not contribute to the loss of that level.

4 EXPERIMENTS AND RESULTS

All experiments were repeated three times and we report the median of those three runs in Table 1. The extraction network was evaluated with four configurations. The FinBERT model is using a WordPiece (WP) tokenizer and either the CLS token or the average (AVG) of all output tokens as a text representation. The Doc2Vec model is using either the WordPiece (WP) tokenizer or the SentencePiece (SP) tokenizer.

4.1 Evaluation metrics

We approach the article categorization problem through the information retrieval paradigm. Namely, we try to return the set of the most probable IPTC Subject Codes assigned to each article in the STT corpus. We use two performance metrics, the mean average precision (mAP) and recall at 10 (R@10). The mean average precision returns the expectation of the area under the precision-recall curve for a random query. The recall at 10 computes the ratio of correct topics found in the 10 tags with the highest predicted probability. To measure the generalization of our prediction models, we compute these metrics separately for each level of the IPTC Subject Codes.

4.2 Results and discussion

In all experiments, the Doc2Vec model performed significantly better than the FinBERT model, regardless of the specific extractor or predictor setup. This is surprising in the light of other successful applications of BERT models. Nevertheless, as there are less than 5% of articles in the training set and less than 7% of articles in the test set that have more than 512 tokens (the limitation of BERT but not Doc2Vec) we cannot assign the poor performance of BERT to this limitation.

Some other relevant findings are as follows. While for some tasks[9] the BERT average token representation performs better than the representation based on the CLS token, in our experiments the CLS and the AVG representations perform comparably. The three-layer network mimicking the shape of the tree-like IPTC Subject Codes hierarchy did not yield any systematic improvement over the single, flat layer of the neural network. Difference in tokenizers for Doc2Vec experiments shows small, but consistent improvement when using the SentencePiece tokenizer.

5 CONCLUSIONS AND FURTHER WORK

In this work, we have compared a monolingual FinBERT and Doc2Vec model on the IPTC Subject Codes prediction task in Finnish language. We evaluated several variations of experiments and achieved consistently better results with a Doc2Vec model. In contrast to the Doc2Vec, the BERT model has a limitation in the form of maximum number of input tokens. We believe the results cannot be explained by this as the data used does not contain a significant amount of documents exceeding this limit. We plan to explore this topic further in hope of understanding and addressing this problem. Recent work in BERT finetuning strategies[5] identifies a problem of vanishing gradients due to excessive learning rates and implementation details of the optimizer.

Our attempt at encoding the hierarchical nature of the prediction task did not yield systematic improvement and we believe it is worthwhile to explore other strategies and improve on this area, like encoding the hierarchy of the predictions in the loss function itself.

For Doc2Vec experiments, consistently better results were achieved using the SentencePiece[2] tokenizer over the WordPiece[14] tokenizer used in FinBERT model. Both of those tokenizers retain the whole information of the input as there are no destructive operations on the text. We plan further experiments

Table 1: Results for different experimental configurations.

Extractor	Predictor	mAP (lvl 1)	mAP (lvl 2)	mAP (lvl 3)	R@10 (lvl 1)	R@10 (lvl 2)	R@10 (lvl 3)
FinBERT (CLS)	Flat	0.5432	0.2047	0.1031	0.9058	0.3687	0.2242
FinBERT (CLS)	Tree	0.5434	0.1949	0.1043	0.9058	0.3602	0.2417
FinBERT (AVG)	Flat	0.5401	0.2026	0.1006	0.9045	0.3692	0.2391
FinBERT (AVG)	Tree	0.5410	0.2088	0.1089	0.9078	0.3724	0.2367
Doc2Vec (WP)	Flat	0.8091	0.5204	0.2990	0.9721	0.7008	0.4750
Doc2Vec (WP)	Tree	0.8127	0.5202	0.2972	0.9743	0.7099	0.4714
Doc2Vec (SP)	Flat	0.8298	0.5550	0.3149	0.9803	0.7277	**0.4951**
Doc2Vec (SP)	Tree	**0.8315**	**0.5643**	**0.3282**	**0.9832**	**0.7358**	0.4896

to confirm and quantify these findings and understand what enables such improvement of downstream prediction task at the tokenizer level.

ACKNOWLEDGMENTS

The work was partially supported by the Slovenian Research Agency (ARRS) core research programmes P6-0411 and P2-0103, as well as the research project J6-2581 (Computer-assisted multilingual news discourse analysis with contextual embeddings). This paper is supported by European Union's Horizon 2020 research and innovation programme under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media).

REFERENCES

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*. (June 2019), 4171–4186. doi: 10.18653/v1/N19-1423.

[2] Taku Kudo and John Richardson. 2018. Sentencepiece: a simple and language independent subword tokenizer and detokenizer for neural text processing. In (January 2018), 66–71. doi: 10.18653/v1/D18-2012.

[3] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31st International Conference on Machine Learning* (Proceedings of Machine Learning Research) number 2. Eric P. Xing and Tony Jebara, editors. Volume 32. PMLR, Bejing, China, (June 2014), 1188–1196. https://proceedings.mlr.press/v32/le14.html.

[4] Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*. https://openreview.net/forum?id=Bkg6RiCqY7.

[5] Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2021. On the stability of fine-tuning {bert}: misconceptions, explanations, and strong baselines. In *International Conference on Learning Representations*. https://openreview.net/forum?id=nzpLWnVAyah.

[6] Andraž Pelicon, Marko Pranjić, Dragana Miljković, Blaž Škrlj, and Senja Pollak. 2020. Zero-shot learning for cross-lingual news sentiment classification. *Applied Sciences*, 10, 17. issn: 2076-3417. doi: 10.3390/app10175993. https://www.mdpi.com/2076-3417/10/17/5993.

[7] Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, (July 2019), 4996–5001. doi: 10.18653/v1/P19-1493. https://aclanthology.org/P19-1493.

[8] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21, 140, 1–67. http://jmlr.org/papers/v21/20-074.html.

[9] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, (November 2019), 3982–3992. doi: 10.18653/v1/D19-1410. https://aclanthology.org/D19-1410.

[10] STT. 2019. Finnish news agency archive 1992-2018, source (http://urn.fi/urn:nbn:fi:lb-2019041501). (2019).

[11] Matej Ulčar and Marko Robnik-Šikonja. 2020. Finest bert and crosloengual bert. In *Text, Speech, and Dialogue*. Petr Sojka, Ivan Kopeček, Karel Pala, and Aleš Horák, editors. Springer International Publishing, Cham, 104–111.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (NIPS'17). Curran Associates Inc., Red Hook, NY, USA, 6000–6010. isbn: 9781510860964.

[13] Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: bert for finnish. (2019). arXiv: 1912.07076 [cs.CL].

[14] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: bridging the gap between human and machine translation. (2016). arXiv: 1609.08144 [cs.CL].