This journal provides immediate open access to its content under the Creative Commons BY 4.0 license. Authors who publish with this journal retain all copyrights and agree to the terms of the above-mentioned CC BY 4.0 license

DOI: 10.2478/seeur-2023-0035

PRE-TRAINING MLM USING BERT FOR THE ALBANIAN LANGUAGE

Labehat Kryeziu
Ph.D. Candidate, Faculty of Contemporary Sciences and Technologies
South East European University, North Macedonia
lk29054@seeu.edu.mk

Visar Shehu
Full Professor, Faculty of Contemporary Sciences and Technologies
Southeast European University, North Macedonia
v.shehu@seeu.edu.mk

ABSTRACT

Knowing that language is often used as a classifier of human intelligence and the development of systems that understand human language remains a challenge all the time (Kryeziu & Shehu, 2022). Natural Language Processing is a very active field of study, where transformers have a key role. Transformers function based on neural networks and they are increasingly showing promising results. One of the first major contributions to transfer learning in Natural Language Processing was the use of pre-trained word embeddings in 2010 (Joseph, Lev, & Yoshua, 2010). Pre-trained models like ELMo (Matthew, et al., 2018) and BERT (Delvin, et al., 2019) are trained on large corpora of unlabeled text and as a result learning from text representations has achieved good performance on many of the underlying tasks on datasets from different domains. Pre-training in the language model has proven that there has been an improvement in some aspects of natural language processing, based on the paper (Dai & Le, 2015). In present paper, we will pre-train BERT on the task of Masked Language Modeling (MLM) with the Albanian language dataset (alb_dataset) that we have created for this purpose (Kryeziu et al., 2022). We will compare two approaches: training of BERT using the available OSCAR dataset and using our alb_dataset that we have collected. The paper

shows some discrepancies during training, especially while evaluating the performance of the model.

Key words: NLP, transformers, pre-training, BERT

INTRODUCTION

Language is undoubtedly the primary tool that characterize human beings, and it is crucial to achieve an appropriate level of human-like intelligence representation through Natural Language Processing. However, previous attempts in this field have been largely ineffective, and text understanding systems have been limited to specific domains for which they were designed. These systems have often failed to handle new or unknown inputs, and their rigidity made it difficult to capture the complexities of natural language. As a result, the field of natural language processing is continually evolving, associated with frequent and significant changes in neural network architecture. It is quite challenging to predict which combination of these model structures would be optimal, considering various factors such as model size, data quality, and training speed. Nevertheless, the use of transformers, such as BERT (Bidirectional Encoder Representations from Transformers), is a widely accepted approach to learn text representation. These transformer-based models have shown remarkable success in solving sequence-to-sequence problems. NLP has demonstrated exceptional results in several languages, including English, French, German, and Chinese, thanks to the availability of vast digital textual corpora.

Over the past five years, deep neural networks have emerged as a prominent category of machine learning models, widely used in scenarios where learning from data is necessary. These models automatically learn from a hierarchical structure of layers, minimizing the need for manual feature engineering. Consequently, human effort is mainly dedicated to selecting the appropriate architecture for a given task and training environment. Notably, pre-trained word embeddings were among the initial breakthroughs in transfer learning for natural language processing, as reported in 2010 by Joseph et al. (2010). Pre-trained models such as ELMo (Matthew et al., 2018) and BERT (Delvin et al., 2019) are trained on extensive collections of unlabeled text data, leading to superior performance across a range of tasks and datasets from various domains.

In present paper we aim to pre-train BERT for the Albanian language. For the dataset, we use our own generated and curated dataset (Kryeziu et. al., 2022). We call this dataset

alb_dataset, and we have made it available in the Huggingface platform. In addition to this dataset, there are other available corpora, such as the OSCAR dataset, CC_Aligned etc. However, as we discussed in (Kryeziu et. al, 2022), there are many issues with these datasets. While OSCAR shows similar statistical properties to the alb_dataset, there are still many issues with it, which has encouraged us to use our approach.

LITERATURE REVIEW

Pre-training of neural language models has been demonstrated to be highly effective on large-scale collections of unlabeled text, resulting in significant improvements in learning through transfer to natural language processing (NLP). In practice, most pre-trained language models are incorporated as a fundamental component in NLP systems to enhance and streamline a variety of downstream tasks (Devlin et al., 2019; Radford et al., 2019; Yang et al., 2019). As presented in the paper by Devlin et al. (2019), linguistic models based on transformers, such as BERT (bidirectional encoder representations from transformers), have shown positive outcomes in extracting NLP information from unlabeled text and improving performance across various domains. Several studies in the medical field, including SciBERT, ClinicalBERT, and BioBERT, have demonstrated that pre-training BERT models on large biomedical and scientific text corpora yields favorable results in diverse text extraction tasks (Alsentzer et al., 2019; Lee et al., 2020). The success achieved by transformer models on multiple NLP tasks has motivated researchers to explore their application in a broad range of downstream applications, as reported in Devlin et al. (2019) and Radford et al. (2019). The authors were encouraged by the effectiveness of unsupervised fine-tuning approaches in achieving these outcomes.

A variety of transformer models have been successful in various language tasks. Initially proposed in (Vaswani, et al., 2017), transformers were first used for machine translation, significantly improving the performance of the existing models. Among these models is Bidirectional Encoder Representations from Transformers (BERT), which has achieved state-of-the-art results on several natural language processing tasks, including natural language inference and question answering (Delvin, et al., 2019). BERT is a pre-trained model that utilizes large text corpora and typically involves two unsupervised tasks. The first task, called "Masked Language Model" (MLM), involves restricting the model to predict the missing word in a specific position in the text. The second task is "Next Sentence Prediction" (NSP). These tasks help the model learn internal representations of language that can be utilized for

different language tasks. This architecture has been shown to be effective, representing a general language model that can be fine-tuned with relatively less data for a variety of tasks and outperforming previous architectures (Delvin, et al., 2019).

Bidirectional Encoder Representations from Transformers (BERT) has proven to be a highly effective model for various natural language processing (NLP) tasks, such as text classification and question answering (Yuwen & Zhaozhuo, 2018). In a recent study by Abdelali, Hassan, and Mubarak (2021), existing BERT models were compared, including the multilingual BERT (mBERT) which includes Arabic. However, the pre-trained AraBERT model has demonstrated better performance on various tasks and has surpassed mBERT. AraBERT was trained on a large dataset of Modern Standard Arabic (MSA), while ArabicBERT was trained on an even larger corpus of Arabic text from the web (OSCAR Corpus). While multilingual BERT was trained with texts from Wikipedia in 104 languages, monolingual BERT models have been shown to outperform it in many cases, such as for Finnish (Virtanen et al., 2019), Dutch (Delobelle et al., 2020), Spanish (Canete et al., 2019), Turkish (Schweter, 2020), Chinese (Tian et al., 2020), and more.

After the introduction of the first BERT model, several other models inspired by it were developed, such as BERT-base, BERT-large, and multilingual BERT. Researchers later proposed the first BERT model trained in Chinese, and a new model called BERTje was created by applying the BERT architecture to pre-train Dutch texts. However, it is not yet known how well multilingual BERT performs in many languages, as it only covers 104 out of approximately 4000 languages in the LRL group. RoBERTa was developed to improve upon BERT's performance on tasks such as reading comprehension, question answering, and language understanding. Unlike BERT, RoBERTa only uses the masked language model (MLM) task, omitting the next sentence prediction (NSP) task. Although RoBERTa does not have a multilingual model, it has demonstrated good performance for Dutch. Morphologically rich languages, such as Turkish, have achieved good results when using the BERT-multilingual neural model in NLP tasks (Özçift, et al., 2021). With the aim of training BERT specifically for Albanian, we have used transformers to perform the masked language modeling task on the alb_dataset, a dataset we created.

METHODOLOGY

Creating the dataset has been a challenge in itself; for the creation of our dataset (alb_dataset), numerous texts written by different authors, news from different portals, Wikipedia, minutes of the Parliament of Albania, various reports of the ministries of Kosovo and Albania, North Macedonia, etc., have been used. Texts selected for the creation of the dataset are written in the standardized Albanian language. A web crawler was implemented which automatically downloaded text from different pages: media, government sites, Wikipedia, etc. We were careful to crawl only documents that had a low probability of having spelling and grammar errors and also text that was written in the standardized Albanian language (not using dialects).

After collecting the texts, a second data cleaning step was implemented as well. During this phase we cleaned the text from html, removed special characters from text and removed any misspelled words by comparing them to a dictionary of the Albanian language. The size of this is substantial 950 MB, with 149,473,629 words in total (Kryeziu et al., 2022).

TRAINING THE MODEL

To train the model we used the Huggingface library (HuggingFace, 2022) and Google Collab using TPU powered virtual machines. We wanted to evaluate two different approaches: pre-train BERT using the OSCAR dataset and pre-train BERT using our own dataset. The steps used for training were the same for both approaches:

- Split the dataset to manageable files
- Tokenize the files using BPE tokenizers
- Train the model using PyTorch with CUDA enabled
- Build a transformer model

Both datasets that we used for training have a large size and we had to optimize our approaches when training the model. For this purpose, we split the dataset into more manageable 15 MB chunk files.

For training the model we had to choose which tokenizer would be the most suitable for the Albanian language. The impact of tokenization has been well documented in past research. A good study in this context has been published in (Kamps et. al., 2020). In a similar fashion, (Gerz et. al., 2016) discuss the merits of tokenization for morphologically rich languages.In general, there are four type of tokenizers: byte-level, character-level, word-level,

and subword-level tokenizers. Depending on the language used, different tokenizers would lead to different results. E.g., it would not make sense to use word-level tokenizers for languages such as Chinese of Japanese since spaces between words do not exist.

The Albanian language uses the Latin alphabet and uses spaces to separate words. Therefore, our first approach was to use word-level tokenizer. We also considered to use subword-level tokenizers and byte-level tokenizers. We discuss the results from these three approaches in the evaluation of our methods.

The different types of tokenizers will output the tokens differently. For example, a word level tokenizer given the input:

Ky është një shembull i tokenizuesit për gjuhën shqipe.

will output the following:

['Ky', 'është', 'një', 'shembull', 'i', 'tokenizuesit', 'për', 'gjuhën', 'shqipe']

On the other hand, a byte level tokenizer, would output the following:

['Ky', 'ĠÃ', '«', 'sht', 'Ã', '«', 'Ġnj', 'Ã', '«', 'Ġshembull', 'Ġi', 'Ġtoken', 'iz', 'uesit', 'Ġp', 'Ã', '«', 'r', 'Ġgju', 'h', 'Ã', '«', 'n', 'Ġshqipe', '.']

Note that the byte-level tokenizer will create tokens that would correspond to byte sequences and will not use a special treatment for the punctuation marks.

Finally, we have the subword level tokenizer. This tokenizer will break the string into subwords as follows:

['Ky', 'Ġës', 'ht', 'ë', 'Ġnjë', 'Ġsh', 'emb', 'ull', 'Ġi', 'Ġtok', 'en', 'iz', 'ue', 'sit', 'Ġpër', 'Ġgjuh', 'ën', 'Ġshq', 'ipe']

After we have a tokenizer, we want to build a MLM model for the Albanian language. Note that there are not many good MLM models for the Albanian language. The models that we built, are available for use in Hugging Face. We trained three models (word-level tokenizer, byte-level tokenizer and subword-level tokenizer) and evaluated them using the loss function for each of them. We used the same parameters for each model:

- **Maximum sequence length** was set to 256 since we were training a MLM model for the Albanian language. In a previous paper (Kryeziu et. al., 2022) we saw that sentences are between 75 and 100 characters. Therefore, we considered 256 to be enough.

- **Train batch size** was set to 756 as that is the common recommendation.
- We used two **epochs** since we did not see much improvement if set to more. Note that the recommended number of epochs is six.
- **Learning rate** was set to lr=5e-5

EVALUATION

For our first approach and to establish a baseline, we used the OSCAR dataset. Unfortunately, the results of that training were not satisfactory. We can see that by the loss function (in two epochs) in the picture below. The loss function shows that the model is learning but not to a satisfactory level. Further evaluations show that MLM does not predict the correct words. There can be many reasons why this happens, and this would be a topic of research in our future studies.

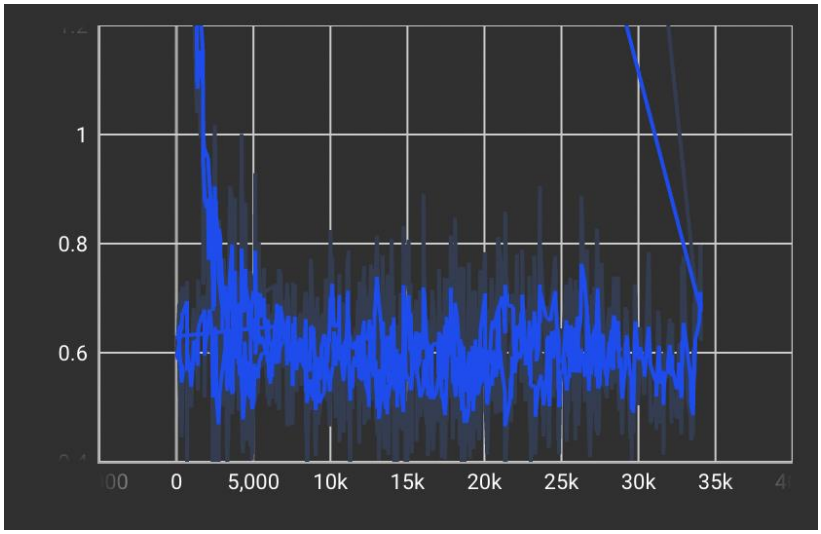


Figure 1 - Loss function when training model using OSCAR dataset

In the next few approaches, we used our own dataset with different tokenization strategies. Some of the sentences we used for testing are below:

1. `Sot do të [MASK] në shkollë.`
2. `Studentët e UEJL këtë javë kanë [MASK] të provimeve.`
3. `Dje [MASK] ishte me shi.`
4. `Gjuha programuese Python është e preferuar në mësimin e [MASK].`

The outputs are as follows for each tokenizer:

Table 1- Outputs for a Word-level tokenizer

Input	Tokenizer	Output
Sot do të [MASK] në shkollë.	Word level	['shkoj', 'marr', 'punoj', 'ndjek', 'studioj']

Studentët e UEJL këtë javë kanë [MASK] të provimeve.	Word level	['përballje', 'nënkuptojnë', 'përgatitjen', 'mbajtjen', 'shkallën']
Dje [MASK] ishte me shi.	Word level	['moti', 'era', 'qielli', 'ajri', 'ambienti']
Gjuha programuese Python është e preferuar në mësimin e [MASK].	Word level	['informatikën', 'kompjuterik', 'programimit', 'statistikën', 'algjebrën']

Table 2 - Outputs for a sub-word level tokenizer

Input	Tokenizer	Output
Sot do të [MASK] në shkollë.	Sub-Word level	[' marr', ' shkoj', ' ndjek', ' punoj', ' qëndroj']
Studentët e UEJL këtë javë kanë [MASK] të provimeve.	Sub-Word level	[' prov', ' sem', ' detyr', ' test', ' mbajt']
Dje [MASK] ishte me shi.	Sub-Word level	[' mot', ' qiell', ' ver', ' ditë', ' ngrohtë']
Gjuha programuese Python është e preferuar në mësimin e [MASK].	Sub-Word level	[' programimit', ' kërkimit', ' teknologjisë', ' sintaksës', ' programeve']

The results we got for byte-level tokenizers were not meaningful. Obviously, the byte-level tokenizer would output bytes in the UTF-8 encoding. These results have the following shape:

```
['\xc3', '\x8b', '\xc3', '\xab', '\xc3', '\xa7', '\xc3', '\xab']
```

The above prediction is for the first string. However, they did not result to a meaningful output. The reason for this was that the input data we used for training was a combination of UTF and ASCII text. Furthermore, UTF data was encoded using different encoding schemes for different files. To resolve this issue, we will need to pre-process our input data, so we have a more consistent input format.

The other two tokenizers, word-level and sub-word level, usually perform good with similar and acceptable results. There is an issue with the sub-word level tokenizer which fails to correctly split Albanian words into syllables. This is because of the complex rulesets of the Albanian language and the failure of the tokenizers to recognize digraphs as single letters.

CONCLUSIONS AND FURTHER DISCUSSION

In this paper we researched different means of training a BERT MLM for the Albanian language. We used a dataset that we had collected and curated for this purpose.

One major issue we encountered was the hardware requirements needed to train our models. It would be very difficult and expensive to run the models in personal computers. Our data shows that in order to train a model with our dataset we needed at least 37GB of RAM memory and a GPU with at least 12 GB RAM. For this purpose, we chose to build our models using different cloud providers. Using Google Colab, we had satisfactory results for smaller datasets, however, Google Colab offers only 25.5 GB of RAM memory to its paid users (only 12 GB for free users). Another cloud provider that satisfied our requirements was Paperspace (https://www.paperspace.com/) which allowed us to use virtual machines with 16 GB VRAM and 48 GB RAM.

The input data needs also to be pre-processed so we can get more trustworthy results. This is seen when we use a byte-level tokenizer.

A major contribution to the field would be to fine-tune the training process specifically for the Albanian language. The current approach we have taken are very generic as per the approach discussed in the relevant papers. However, they are usually specifically designed for the English language. A model for the Albanian should specifically consider the language structure and the grammar of the language. This will affect the choice of the tokenizer. Two other considerations are the vocabulary size of the Albanian language and handling diacritical marks (e.g., ë and ç). Finally, the Albanian language uses combination of letters treated as a single character (e.g., dh, gj, sh, nj …) which many tokenizers (word-level and byte-level) fail to recognize. All the above considerations should be taken into account if one needs to build a good model for the language.

REFERENCES

1. Abdelali, A., Hassan, S., & Mubarak, H. (2021). Pre-Training BERT on Arabic Tweets: Practical Considerations. *Qatar Computing Research Institute.* Doha 5825, Qatar: arXiv.
2. Alsentzer, E., Murphy, J., Boag, W., Weng, W.-H., Jindi, D., Naumann, T., & McDermott, M. (2019). Publicly Available Clinical BERT Embeddings. (pp. 72-78). Proceedings of the 2nd *Clinical Natural Language Processing Workshop*.
3. Canete, J., Chaperon, G., & Fuentes, R. (2019). Spanish pre-trained bert model and evaluation data. *PML4DC* at ICLR.
4. Cui, Y., Che, W., Liu, T., Qin, B., Yang, Z., Wang, S., & Hu, G. (2019). Pre-Training with Whole Word Masking for Chinese BERT.
5. Dai, A., & Le, Q. (2015). Semi-supervised sequence learning", *In Advances in neural information processing systems*., (pp. 3079–3087).
6. De Vries, W., Van Cranenburgh, A., Bisazza, A., Caselli, T., Van Noord, G., & Nissim, M. (2019). BERTje: A Dutch BERT Model. arXiv preprint arXiv.
7. Delobelle, P., Winters, T., & Berendt, B. (2020). RobBERT: a Dutch RoBERTa-based Language Model.
8. Delvin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL*-HLT.
9. Hugging Face (2022). Transformers (4.4.2) [Computer software]. Retrieved from https://github.com/huggingface/transformers
10. Gerz, D., Vulić, I., Ponti, E., Naradowsky, J., Reichart, R., Korhonen, A. (2018) Language Modeling for Morphologically Rich Languages: Character-Aware Modeling for Word-Level Prediction. Transactions of the Association for Computational Linguistics; 6 451–465. doi: https://doi.org/10.1162/tacl_a_00032
11. Joseph, T., Lev, R., & Yoshua, B. (2010). Word representations: A simple and general method for semi-supervised learning. *48th Annual Meeting of the Association for Computational Linguistics.*
12. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. (pp. 1097-1105). Advances in neural information processing systems.
13. Kryeziu, L., & Shehu, V. (2022). A Survey of Using Unsupervised Learning Techniques in Building Masked Language Models for Low Resource Languages. *2022 11th Mediterranean Conference on Embedded Computing (MECO)* (pp. 1-6). Budva: MECO.
14. Kryeziu, L., Shehu, V., & Caushi, A. (2022). Evaluation and Verification of NLP Datasets for the Albanian Language. *International Conference on Artificial Intelligence of Things.* Istanbul, Turkey.
15. Kamps, J., Kondylidis, N., & Rau, D. (2020). Impact of Tokenization, Pretraining Task, and Transformer Depth on Text Ranking. In *TREC*.
16. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., Ho So, C., & Kang, J. (2020). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, (pp. 1234–1240).
17. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach.
18. Matthew, E., Mark, N., Mohit, J., Matt, G., Christopher, C., Kenton, L., & Luke, Z. (2018). Deep contextualized word representations.
19. Özçift, A., Akarsu, K., Yumuk, F., & Söylemez, C. (2021). Advancing natural language processing (NLP) applications of morphologically rich languages with

bidirectional encoder representations from transformers (BERT): an empirical case study for Turkish.
20. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI*.
21. Schweter, S. (2020). Berturk - bert models for turkish.
22. Tian, H., Yang, K., Liu, D., & Lv, J. (2020). Anchibert: A pre-trained model for ancient chineselanguage understanding and generation. arXiv.
23. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., . . . Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, (pp. 6000–6010).
24. Virtanen, A., Kanerva, J., Ilo, R., Luoma, J., Luotolahti, J., Salakoski, T., . . . Pyysalo, S. (2019). Multilingual is not enough: Bert for finnish.
25. Winograd, T. (1972). Understanding natural language. (pp. 1-191). Cognitive psychology.
26. Wu, S., & Dredze, M. (2019). The surprising cross-lingual effectiveness of BERT. *Conference on Empirical Methods in Natural Language Processing* (pp. 833–844). Hong Kong, China: Association for Computational Linguistics.
27. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., & Le, Q. V. (2019). Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in Neural Information Processing Systems (NeurIPS)*, (pp. 5754–5764).
28. Yuwen, Z., & Zhaozhuo, X. (2018). Bert for question answering on squad 2.0.