An Empirical Analysis Towards Replacing Vocabulary-Rigid Embeddings by a Vocabulary-Free Mechanism

Alejandro Rodriguez Perez [*1] Korn Sooksatra [*1] Pablo Rivas [1 2] Ernesto Quevedo Caballero [1] Javier S. Turek [3] Gisela Bichler [4] Tomas Cerny [1] Laurie Giddens [5] Stacie Petter [6]

Abstract

This paper addresses the limitations of subword-based models in NLP by aligning the word-embedding layer of a vocabulary-rigid transformer model to a vocabulary-free one. In order to do so, a CNN is trained to mimic the word embeddings layer of a BERT model, using a sequence of byte tokens as input. The study compares cosine-based and Euclidean-based loss functions for training the student network and finds better results with cosine-based metrics. The research contributes techniques for re-training transformer embedding layers and provides insights into loss function selection. The findings have implications for developing flexible and robust NLP models.

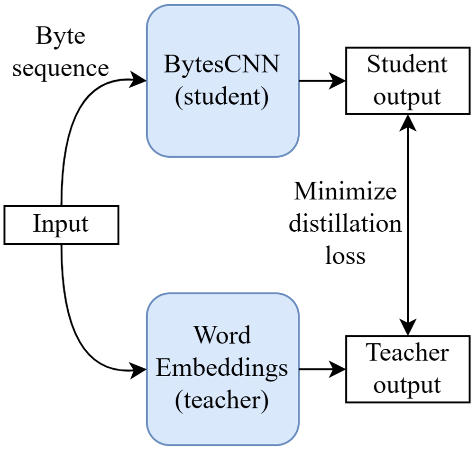

Figure 1: Embedding alignment framework.

1. Introduction

Transformer-based models have become the standard for natural language processing (NLP) tasks due to their ability to capture complex linguistic patterns and dependencies (Chitty-Venkata et al., 2022; Bondarenko et al., 2021). Subword-level tokenization is commonly used in these models to handle out-of-vocabulary inputs and provide more flexibility (Devlin et al., 2019; Radford et al., 2019; Lewis et al., 2020).

However, recent research has highlighted the limitations of subword-level tokenization, including poor generalization for out-of-vocabulary words and domains due to their reliance on a fixed vocabulary (Bostrom & Durrett, 2020; Klein & Tsarfaty, 2020; Hofmann et al., 2021; Dong et al., 2020; Xu et al., 2021). This limitation is particularly problematic for forensic NLP models used to detect covert criminal communications (CCC) that employ unusual characters and subwords for obfuscation (Bromberg et al., 2020; Pei & Cheng, 2022; Tong et al., 2017; Wagner et al., 2020; Wang et al., 2019; Zhu et al., 2019).

To address these limitations, various solutions have been proposed, including character-based models that can handle a wider range of linguistic inputs (Cao & Rimell, 2021; Wang et al., 2021; Hofmann et al., 2022; Mielke et al., 2021; El Boukkouri et al., 2020; Clark et al., 2022; Ma et al., 2020; Pinter et al., 2021).

In this paper, we explore the idea of aligning the word-embedding layer of a vocabulary-rigid transformer model to a vocabulary-free one, inspired by the work of Mersha & Stephen (2022), who introduced DistillEmb, a method for distilling learned word embeddings into a convolutional neural network using contrastive learning with a triplet loss (Schroff et al., 2015). Our goal is to investigate transfer learning from a vocabulary-rigid transformer to a vocabulary-free one, as depicted in Figure 1, through

[*]Equal contribution [1]Department of Computer Science, School of Engineering & Computer Science, Baylor University, Texas, USA [2]Center for Standards and Ethics in Artificial Intelligence, Texas, USA [3]Intel Labs, Portland, Oregon, USA [4]School of Criminology & Criminal Justice, California State University – San Bernardino, California, USA [5]Information Technology and Decisions Sciences Department, G. Brint Ryan College of Business, University of North Texas, USA [6]School of Business, Wake Forest University, North Carolina, USA. Correspondence to: Pablo Rivas <pablo_rivas@baylor.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

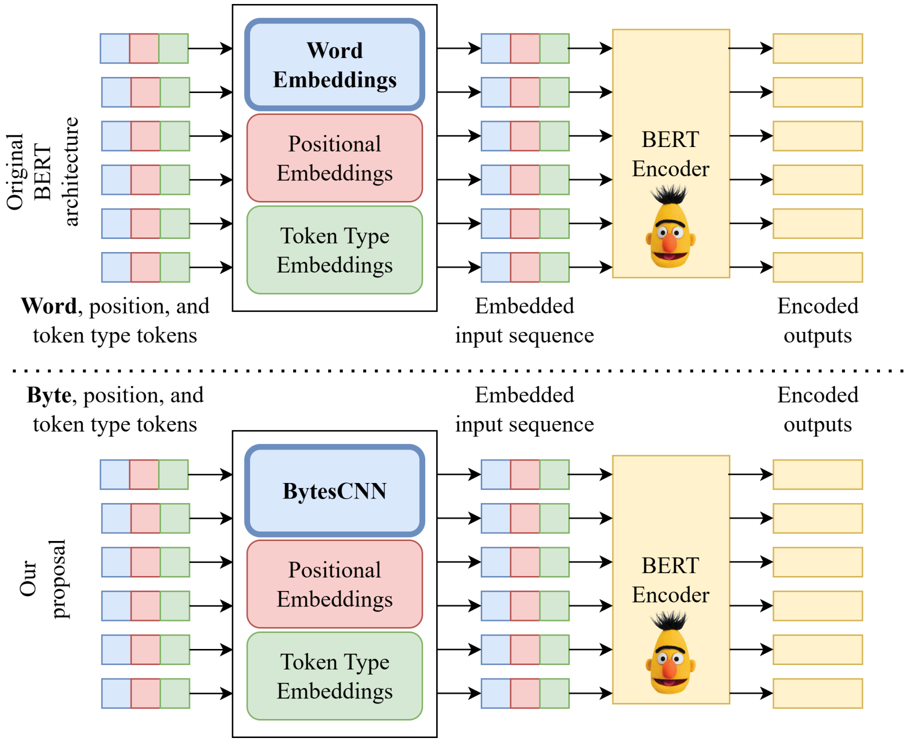

Figure 2: Illustration of the replacement of the transformer word embedding layer with the aligned BytesCNN.

a distillation process (Hinton et al., 2015). This approach allows us to derive a byte-based transformer model without the need for time-consuming pre-training.

We hypothesize that this methodology improves the model's generalization performance and enables it to handle a broader range of linguistic inputs, including out-of-vocabulary words and domain-specific language. By addressing the limitations of subword-based models and exploring transfer learning, we aim to develop more flexible and robust models for NLP tasks.

2. Methodology

Research has demonstrated the efficacy of employing byte-derived word representations to effectively pre-train a large language model (LLM) (Tay et al., 2022; Xue et al., 2022; Clark et al., 2022). This advancement allows for a departure from word representations limited by fixed vocabularies towards vocabulary-flexible alternatives. However, the process of training an LLM from scratch is resource-intensive. As an alternative, we propose a methodology for leveraging transfer learning from a word embedding matrix to a byte-based embedding network. The primary objective is to attain a vocabulary-flexible representation while maximizing the transfer of knowledge inherent in a pre-trained LLM.

To accomplish this, we introduce a neural network that operates on arbitrary byte sequences and generates an embedding vector of appropriate size. Subsequently, we train this byte-based network, utilizing a knowledge distillation framework, to emulate the behavior of the word embedding matrix for words (including subwords) present in the vocabulary of the existing pre-trained LLM, as illustrated in Figure 1. We term this process **alignment**.

The byte-based embedding is then integrated into the LLM in lieu of the word embedding matrix, facilitating the effective transfer of knowledge from a pre-trained LLM with a fixed vocabulary to a vocabulary-flexible counterpart. Figure 2 depicts this modification applied to a transformer model.

2.1. Bytes CNN Architecture

The substitution of the embedding layer in our study was based on a model proposed by El Boukkouri et al. (2020). In the remainder of the paper, we refer to it as BytesCNN to highlight the utilization of byte-based tokens from the vocabulary, enabling the model to process any given sequence of bytes.

Notably, the BytesCNN model employs four parallel convolutions with distinct kernel sizes and number of channels applied to the input. Subsequently, the resulting outputs are concatenated after undergoing max pooling and ReLU activation. To further process this concatenated output, a Highway layer (Srivastava et al., 2015) is applied. Finally, the vector is projected into the embedding space.

2.2. Loss Function

In our study, we employed a diverse set of loss functions to facilitate our analysis. In addition to the conventional mean square error (MSE) loss, we explored the utilization of cosine error and two composite loss functions that combine MSE and cosine error, as elaborated upon in a subsequent subsection.

Through examination, we discovered that the embedding vectors within the BERT model exhibit a tendency to reside proximately on the surface of a ball with an approximate radius of 1.41 units. Moreover, there exists a slight variance in the norms of these vectors, specifically measured at 0.19. This empirical observation reinforced our rationale for adopting cosine-based loss functions in our research.

Our first loss function uses a plain cosine error between two vectors. Given two vectors, x and y, the cosine error is defined as

$$L(x, y) = 1 - \cos(x, y), \tag{1}$$

where $cos(x, y)$ is the cosine of the angle between the vectors x and y.

The utilization of this particular function during prediction gives rise to an issue: the embedding network does not undergo optimization to align with the original vectors' lengths. To address this issue, when employing an embedding network trained with this loss, we apply normalization to the embedding vectors, adjusting them to the mean length of the original embedding matrix.

As an alternative approach to normalization, we utilize loss functions that encompass both direction and magnitude considerations, with an emphasis on directionality. These loss functions integrate both Euclidean distance and cosine distance within a unified loss formulation, with one employing addition and the other utilizing multiplication as the combining operations.

The additive Euclidean-cosine error function is defined as:

$$L(x, y) = |x - y| + \alpha(1 - \cos(x, y)) \tag{2}$$

Whereas the multiplicative Euclidean-cosine error is:

$$L(x, y) = \alpha(|x - y|)(1 - cos(x, y)) + |x - y| \tag{3}$$

It is worth noting that the multiplicative Euclidean-cosine error should exhibit faster convergence towards vectors aligning closely with the target direction since the loss function grows more quickly in directions that deviate further from the target.

2.3. Contrastive Learning

We also employ the training of aligned embedding layers using a contrastive learning objective, similar to the approach proposed byMersha & Stephen (2022). Specifically, we adopt a triplet loss function. The objective of the triplet loss is to minimize the Euclidean distance between similar vectors while maximizing the distance between dissimilar vectors. Mathematically, the triplet loss is defined as:

$$L(x, y_p, y_n) = \max(|x - y_p|^2 - |x - y_n|^2 + \alpha, 0), \tag{4}$$

where x, y_p, and y_n denote the anchor vector, positive vector, and negative vector, respectively. In our case, the anchor vector corresponds to the output of the BytesCNN, the positive vector represents the ground truth embedding from the BERT word embedding layer, and the negative vector is selected following the same procedure as described by Mersha & Stephen (2022).

Moreover, we incorporate a variant of the triplet loss that utilizes the cosine error instead of the Euclidean distance. This variant, known as the Angular Triplet Loss, has been explored in the context of person re-identification (Ye et al., 2020; Li et al., 2021). It is defined as:

$$L(x, y_p, y_n) = \max(\cos(x, y_n) - \cos(x, y_p) + \alpha, 0) \tag{5}$$

Note the positive and negative operations appear inverted due to the definition of the cosine error between vectors x and y as $1 - \cos(x, y)$.

Throughout our experiments, we set the value of α to 1 for all evaluations of the triplet loss.

3. Experimental Setup and Results

We employed the BERT model (Devlin et al., 2019) as the basis for evaluating our proposed methodologies. It is important to note that, in principle, our method can be applied to any transformer architecture or network utilizing a word embedding layer. For our experiments, we utilized the `bert-base-uncased` model, which is accessible through the Huggingface models hub [1].

During the training process, we conducted 10^4 epochs with a batch size of 100. The Adam optimizer was employed with default parameter values: $\alpha = 10^{-3}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. Additionally, we implemented a learning rate reduction strategy, reducing the learning rate by a factor of 0.9 when the training progress plateaued.

The vocabulary size of the BERT model amounts to 30522 words. However, we excluded any tokens that were not utilized during the training procedure, specifically those

[1] `https://huggingface.co/bert-base-uncased`

reserved for future additions. Consequently, the effective vocabulary size used in our experiments was reduced to 29528 words. Subword tokens, e.g. *##able*, are included, and all the characters are used, including the # symbols.

We explored two variants of the BytesCNN architecture, namely BytesCNN-small and BytesCNN-big, which differ in terms of their sizes. The small variant adheres to the configuration defined by El Boukkouri et al. (2020), consisting of seven 1D convolutional layers with the following filter specifications: $(1, 32)$, $(2, 32)$, $(3, 64)$, $(4, 128)$, $(5, 256)$, $(6, 512)$, and $(7, 1024)$. In each filter specification denoted as (K, O), K represents the kernel size, and O corresponds to the number of output channels. On the other hand, the BytesCNN-big variant duplicates each filter. Table 1 presents a comparison of the parameter counts between these variants and the BERT word embedding layer, disregarding the parameters associated with the unused tokens.

Model	Num. parameters
BERT word embedding	22,677,504
BytesCNN-small	18,562,416
BytesCNN-big	70,674,288

Table 1: No. of parameters of different embedding layers.

Our experiments were carried out on the two pre-training tasks utilized by BERT: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). We attribute significant importance to evaluating the performance in these tasks, as they constitute the foundation on which the baseline model is pre-trained.

For evaluation, we used the `wikitext-2-raw -v1` subset of Wikipedia (Wikitext dataset). We also ran the experiments on the IMDB dataset. In both cases, we used the train split. The datasets are available online .[2] [3]

Table 2 shows the accuracy of each scenario as evaluated on MLM and NSP in the Wikitext and IMDB datasets. Only averages of randomly-seeded experiments are shown. Standard deviations, albeit not shown, were in the order of 10^{-4} for most cases. For each model size and each task, the top-3 performing models are in italics, and the best-performing is highlighted in bold.

As observed, none of the alternative models achieve exactly the same performance as the baseline. Since those are only trained to mimic the baseline, we do not expect a better performance from the student models. However, some of them obtain very close results to the baseline. To validate this, we conducted two-sample paired t-tests to determine whether there is a statistically significant difference between the baseline and each one of the models. Table 3 depicts the resulting p-values.

Several conclusions can be drawn. First, the difference between small models and the respective big models is notable. All the big model variants' results (except for the triplet Euclidean) do not provide significant evidence to reject the null hypothesis under a significance level of 0.9, meaning they are not significantly different from the baseline as far as these experiments are concerned. Additionally, note that models that were trained using a form of cosine-based distance perform better than those that solely use Euclidean-based losses. It is difficult, however, to draw an absolute conclusion on which cosine-based loss is better.

We developed some experiments to test if using a BytesCNN as the word embedding layer makes the transformer more robust to a noisy input without any extra fine-tuning. Further investigation is required, but preliminary results show that the transformer with the BytesCNN embedding layer is not more robust against the tested model of noise than the baseline model. Therefore, we are inclined to think that to achieve such improved performance, the model needs fine-tuning. A similar principle should apply to transferability to other languages. However, note that fine-tuning this model is significantly faster than what would be the solution otherwise: train from scratch with a new vocabulary. Hence one significant advantage to this method.

4. Conclusion

Our research aimed to investigate the feasibility of transferring knowledge from a fixed-vocabulary embedding layer to a CNN-based neural network that generates word representations based on byte sequences. We utilized a teacher-student methodology and analyzed various loss functions to fit the student's network representation to the teacher's representation. Our results indicate that it is possible to align byte-based embedding with the baseline word embedding matrix, effectively converting a vocabulary-restricted model into a vocabulary-free model while retaining its knowledge significantly. Furthermore, we found that cosine-based metrics are more effective than Euclidean-based loss functions for training the student network.

Our approach offers several contributions to the NLP field, including a method for re-training transformer-based model embedding layers and an evaluation of different loss functions for alignment. Our findings have important implications for developing more adaptable and robust NLP models that can handle various inputs, including those in forensic applications. Future research could explore applying our approach to other NLP tasks and investigating the potential

[2] https://huggingface.co/datasets/wikitext/viewer/wikitext-2-v1/train

[3] https://huggingface.co/datasets/imdb

	BERT Baseline		**BytesCNN-small**		**BytesCNN-big**	
Dataset	**Wikipedia**	**IMDB**	**Wikipedia**	**IMDB**	**Wikipedia**	**IMDB**
Task: **Masked Language Modeling**						
	0.6226	0.5748				
MSE			0.2666	0.2059	0.5876	0.5319
Cosine			***0.5600***	***0.5022***	***0.5919***	0.5356
AEC			*0.5271*	*0.4692*	0.5883	*0.5374*
MEC			0.3071	0.2504	*0.5914*	***0.5376***
TE			0.0939	0.0414	0.3005	0.1836
TA			*0.5578*	*0.5002*	*0.5912*	*0.5350*
Task: **Next Sentence Prediction**						
	0.9421	0.6570				
MSE			0.6215	0.5267	*0.9407*	0.6474
Cosine			*0.9364*	*0.6296*	0.9397	*0.6560*
AEC			*0.9369*	*0.6336*	***0.9416***	***0.6568***
MEC			0.7102	0.5292	*0.9412*	0.6539
TE			0.5603	0.5129	0.6119	0.5390
TA			***0.9372***	***0.6358***	0.9397	*0.6544*

Table 2: Accuracy of BERT with a BytesCNN aligned embedding layer in two datasets on the MLM and NSP tasks. Baseline performance included for reference. AEC, MEC, TE, and TA stand for additive Euclidean-cosine, multiplicative Euclidean-cosine, triplet Euclidean, and triplet angular.

BytesCNN-small					
MSE	Cosine	AEC	MEC	TE	TA
0.013	0.073	**0.108**	0.012	0.022	0.091
BytesCNN-big					
MSE	Cosine	AEC	MEC	TE	TA
0.111	**0.157**	**0.176**	**0.149**	0.016	**0.145**

Table 3: List of p-values of several two-sample paired t-tests. Each cell corresponds to the p-value of a t-test where one sample is the baseline model and the other is the model labeled in the corresponding column. The results in which the null hypothesis cannot be rejected with a significance level of at least 0.9 are highlighted in bold.

of byte-based representations for enhancing the forensic capabilities of NLP models.

Acknowledgements

This article is based on work supported by the National Science Foundation under Grant No. 2039678, 2136961, and 2210091. Any opinions, findings, conclusions, or recommendations expressed in this article are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

Bondarenko, Y., Nagel, M., and Blankevoort, T. Understanding and overcoming the challenges of efficient transformer quantization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7947–7969, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.627. URL https://aclanthology.org/2021.emnlp-main.627.

Bostrom, K. and Durrett, G. Byte pair encoding is suboptimal for language model pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 4617–4624, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.414. URL https://aclanthology.org/2020.findings-emnlp.414.

Bromberg, M., Welmans, L., and Lee, C. Reading between the text (s)-interpreting emoji and emoticons in the australian criminal law context. *New Criminal Law Review*, 23(4):655–686, 2020.

Cao, K. and Rimell, L. You should evaluate your language model on marginal likelihood over tokenisations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 2104–2114, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.161. URL https://aclanthology.org/2021.emnlp-main.161.

Chitty-Venkata, K. T., Emani, M., Vishwanath, V., and Somani, A. K. Neural architecture search for transformers: A survey. *IEEE Access*, 10:108374–108412, 2022.

Clark, J. H., Garrette, D., Turc, I., and Wieting, J. Canine: Pre-training an efficient tokenization-free encoder for language representation. *Transactions of the Association for Computational Linguistics*, 10:73–91, 2022. doi: 10.1162/tacl_a_00448. URL https://aclanthology.org/2022.tacl-1.5.

Devlin, J., Chang, M.-W., Lee, K., and 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, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Dong, Y., Wang, S., Gan, Z., Cheng, Y., Cheung, J. C. K., and Liu, J. Multi-fact correction in abstractive text summarization. *arXiv preprint arXiv:2010.02443*, 2020.

El Boukkouri, H., Ferret, O., Lavergne, T., Noji, H., Zweigenbaum, P., and Tsujii, J. CharacterBERT: Reconciling ELMo and BERT for word-level open-vocabulary representations from characters. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6903–6915, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.609. URL https://aclanthology.org/2020.coling-main.609.

Hinton, G. E., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *ArXiv*, abs/1503.02531, 2015.

Hofmann, V., Pierrehumbert, J., and Schütze, H. Superbizarre is not superb: Derivational morphology improves BERT's interpretation of complex words. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 3594–3608, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.279. URL https://aclanthology.org/2021.acl-long.279.

Hofmann, V., Schuetze, H., and Pierrehumbert, J. An embarrassingly simple method to mitigate undesirable properties of pretrained language model tokenizers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 385–393, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.43. URL https://aclanthology.org/2022.acl-short.43.

Klein, S. and Tsarfaty, R. Getting the ##life out of living: How adequate are word-pieces for modelling complex morphology? In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pp. 204–209, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.sigmorphon-1.24. URL https://aclanthology.org/2020.sigmorphon-1.24.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and 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, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL https://aclanthology.org/2020.acl-main.703.

Li, Y., Xue, R., Zhu, M., Xu, J., and Xu, Z. Angular triplet loss-based camera network for reid. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7. IEEE, 2021.

Ma, W., Cui, Y., Si, C., Liu, T., Wang, S., and Hu, G. CharBERT: Character-aware pre-trained language model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 39–50, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.4. URL https://aclanthology.org/2020.coling-main.4.

Mersha, A. and Stephen, W. Distilling word embeddings via contrastive learning. *Transfer Learning for NLP Workshop 2022 – WiNLP*, 2022.

Mielke, S. J., Alyafeai, Z., Salesky, E., Raffel, C., Dey, M., Gallé, M., Raja, A., Si, C., Lee, W. Y., Sagot, B., and Tan, S. Between words and characters: A brief history of open-vocabulary modeling and tokenization in nlp. *ArXiv*, abs/2112.10508, 2021.

Pei, J. and Cheng, L. Deciphering emoji variation in courts: a social semiotic perspective. *Humanities and Social Sciences Communications*, 9(1):1–8, 2022.

Pinter, Y., Stent, A., Dredze, M., and Eisenstein, J. Learning to look inside: Augmenting token-based encoders with character-level information. *ArXiv*, abs/2108.00391, 2021.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Schroff, F., Kalenichenko, D., and Philbin, J. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 815–823, 2015.

Srivastava, R. K., Greff, K., and Schmidhuber, J. Highway networks. *ArXiv*, abs/1505.00387, 2015.

Tay, Y., Tran, V. Q., Ruder, S., Gupta, J., Chung, H. W., Bahri, D., Qin, Z., Baumgartner, S., Yu, C., and Metzler, D. Charformer: Fast character transformers via gradient-based subword tokenization. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=JtBRnrlOEFN.

Tong, E., Zadeh, A., Jones, C., and Morency, L.-P. Combating human trafficking with multimodal deep models. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1547–1556, 2017.

Wagner, A., Marusek, S., and Yu, W. Emojis and law: contextualized flexibility of meaning in cyber communication. *Social Semiotics*, 30(3):396–414, 2020.

Wang, L., Laber, E., Saanchi, Y., and Caltagirone, S. Sex trafficking detection with ordinal regression neural networks. *ArXiv*, abs/1908.05434, 2019.

Wang, X., Ruder, S., and Neubig, G. Multi-view subword regularization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 473–482, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.40. URL https://aclanthology.org/2021.naacl-main.40.

Xu, K., Wu, H., Song, L., Zhang, H., Song, L., and Yu, D. Conversational semantic role labeling. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29: 2465–2475, 2021.

Xue, L., Barua, A., Constant, N., Al-Rfou, R., Narang, S., Kale, M., Roberts, A., and Raffel, C. Byt5: Towards a token-free future with pre-trained byte-to-byte models. *Transactions of the Association for Computational Linguistics*, 10:291–306, 2022.

Ye, H., Liu, H., Meng, F., and Li, X. Bi-directional exponential angular triplet loss for rgb-infrared person reidentification. *IEEE Transactions on Image Processing*, 30:1583–1595, 2020.

Zhu, J., Li, L., and Jones, C. Identification and detection of human trafficking using language models. In *2019 European Intelligence and Security Informatics Conference (EISIC)*, pp. 24–31. IEEE, 2019.

A. Limitations

This investigation has several limitations that need to be acknowledged. Firstly, although our method can theoretically be applied to any transformer-based model, we only used a BERT model as a baseline in our experiments. Additionally, our evaluation only covers BERT's pre-training tasks, and we did not conduct a thorough hyperparameter study.

Secondly, our hypothesis regarding the distribution of embedding vectors' size in BERT's vocabulary may not hold for other language models. Therefore, the success of the cosine-based model may only be applicable to BERT and may not generalize to other models.

Furthermore, our approach appears to negatively impact the generalization ability of transformer-based models in the datasets and tests we conducted with statistical significance. However, further research is necessary to draw definitive conclusions in this regard.

Finally, our objective was to distill the embedding layer by reducing the number of parameters. However, we found that the best results were obtained using a larger model. While our investigation shows promising results, it is important to address these limitations and conduct further research to extend the applicability of our method to other transformer-based models.

B. Ethics Statement

The research conducted did not involve the use of human subjects, and instead relied solely on pre-existing datasets and BERT models. The model used in the research was thoroughly tested on benchmark datasets, and no ethical concerns were identified. However, it is worth noting that the model may inherit biases from the original BERT embeddings, which were trained on a large corpus of text. These biases may have ethical implications, and further research is necessary to address these concerns. The researchers attempted to be transparent in their methodology and demonstrate how they created and tested their model.