Proceedings of the of the 17th Conference on Computer Science and Intelligence Systems pp. 73–77
DOI: 10.15439/2022F197
ISSN 2300-5963 ACSIS, Vol. 30

Using Transformer models for gender attribution in Polish

Karol Kaczmarek
Adam Mickiewicz University,
Faculty of Mathematics and Computer Science,

Applica.ai Sp. z o.o.
Email: karol.kaczmarek@amu.edu.pl

Jakub Pokrywka, Filip Graliński
Adam Mickiewicz University,
Faculty of Mathematics and Computer Science,
Uniwersytetu Poznańskiego 4,
61-614 Poznań, Poland
Email: {firstname.lastname}@amu.edu.pl

***Abstract*—Gender identification is the task of predicting the gender of an author of a given text. Some languages, including Polish, exhibit gender-revealing syntactic expression. In this paper, we investigate machine learning methods for gender identification in Polish. For the evaluation, we use large (780M words) corpus "He Said She Said", created by grepping (for author's gender identification) gender-revealing syntactic expressions and normalizing all these expressions to masculine form (for preventing classifiers from using syntactic features). In this work, we evaluate TF-IDF based, fastText, LSTM and RoBERTa models, differentiating self-contained and non-self-contained approaches. We also provide a human baseline. We report large improvements using pre-trained RoBERTa models and discuss the possible contamination of test data for the best pre-trained model.**

I. Introduction

The task of *gender identification or attribution* consists in predicting the gender of an author of a given text. As such, it is an example of text classification, is usually tackled using supervised machine learning, and is relatively popular in the NLP community. Some recent example of experiments in automatic gender identification for various languages are: [17], [27], [2], [14]. For a critical analysis of gender detection systems and their limitations, see [18].

Collections of gender-labeled texts are required if a system based on supervised machine learning is to be trained. The usual approach is to use metadata such as information on authors (of books, papers, social media posts, etc.). Interestingly, some languages exhibit gender-revealing first-person expressions (cf. *soy polaco* vs *soy polaca* in Spanish), and such expressions can be used to automatically label texts as written by a male or female in order to create a data set. This approach (*distant supervised learning*, [21]) is similar to using emoticons for sentiment analysis tasks [23], [9].

Some languages (e.g. Slavic languages) are more amenable to this distant supervised approach than others (e.g. English or Chinese). The approach was applied to Polish to create a large collection of texts, the "He Said She Said" (HSSS) corpus [10]. In this paper, we (1) re-state the original challenge as a classification task with a probability-based evaluation metric, (2) report on large improvements on the gender detection task using pre-trained RoBERTa models, and (3) discuss the possible contamination of test data with the data on which RoBERTa models were trained.

TABLE I
The "He Said She Said" challenge in numbers.

		characters	words	items
train	total	1,240,131,217	177,428,897	3,601,424
train	male	628,793,876	89,795,752	1,800,712
train	female	611,337,341	87,633,145	1,800,712
dev-0	total	51,080,450	7,158,683	137,314
dev-0	male	26,066,897	3,641,716	68,657
dev-0	female	25,013,553	3,516,967	68,657
dev-1	total	51,009,045	7,275,691	156,606
dev-1	male	25,579,703	3,641,568	78,303
dev-1	female	25,429,342	3,634,123	78,303
test-A	total	43,597,629	6,234,069	134,618
test-A	male	22,253,841	3,175,881	67,309
test-A	female	21,343,788	3,058,188	67,309

In Section II, we discuss the HSSS challenge along with the modifications in the data set done for the purposes of this paper. In the main Section III, we discuss the methods we applied to tackle the challenge of gender identification. Section IV summarizes the results. Finally, we discuss the issues of training/testing data contamination in Section V.

II. He said she said task

Polish is one of the languages with a high frequency of gender-specific first-person expressions. (Only the few languages with gender distinction in the first person, e.g. Ngala [24], might have a higher frequency of such expressions.) This fact was leveraged to create a large gender-labeled corpus for Polish: the "He Said She Said" corpus [10]. Simply CommonCrawl dataset was grepped, using morphological dictionaries and handcrafted rules, for gender-specific first-person expressions. Obviously, there were some issues that needed to be addressed, e.g. quotes, titles, SEO spam.

Later, the corpus was turned into a classification challenge hosted at the Gonito.net platform [11]. All feminine gender-specific first-person expressions were changed to masculine forms in order to prevent classifiers from using the simple gender-revealing syntactic features. Obviously, without this normalization step, the challenge would be trivial. The corpus was randomly split into 4 sets: train set, two development (validation) sets (dev-0 and dev-1) and test set (test-A). The split was based on the websites from which the texts originated, i.e. texts from the same website would belong to the same set. Also, the sets were balanced so that 50%/50% distribution would be obtained, not just for the whole data set, but also for *each* website. For instance, let's consider a message board about pregnancy, in general, there are many more texts written by women there (at least judging by gender-marked first-person expressions), but for the challenge, the same number of male and female texts would be sampled from such a website. This, along with the fact that texts are short, makes the challenge rather difficult.

The challenge was presented [11] to showcase the Gonito.net platform and was discussed there only briefly. For more detailed information about the challenge, see Table I.

For this paper, two changes have been made to the original challenge:

1) *Likelihood* metric was chosen as the main metric (instead of simple accuracy), Likelihood is defined as the geometric mean of probabilities assigned to the gold-standard classes – the motivation was that accuracy is not enough to distinguish solutions of varying quality and confidence;
2) some unwanted blank characters were removed.

Some initial experiments with learning classifiers based on the HSSS data set were presented in [12].

III. Methods

We introduce the structure of our experiments as follows. Subsection III-A describes human baselines. Subsection III-B describes TF-IDF (term frequency-inverse document frequency) based methods. Subsection III-C describes some neural methods. Both III-B and III-C are self-contained. This means not including any data apart from training data available in HSSS task. Subsection III-D describes pre-trained transformer models. Table III presents all classifiers results.

- self-contained – we use only data available from the HSSS task: train on the training set, validate on the dev-0 (validation) set and report results on the test-A (test) set. We will use 256 sequence length which covers most (over 90%) of the HSSS data to speed up the training process.
- non-self-contained – we use publicly available models, which were pre-trained on large amounts of data (may be contaminated by examples from the test or validation set). We will use the sequence length that was saved for these models, which is usually 512.

TABLE II
Results on the test set sample of size 800 created for human evaluation.

method	test accuracy
TF-IDF + logistic regression	0.68500
Polish RoBERTa base	**0.77125**
LSTM (constrained)	0.73375
human 1	0.65250
human 2	0.67375
human 3	0.66250
human 4	0.65625
human ensemble	0.68125

A. Human Baseline

Four people (two females and two males) made predictions for random sample sets of size 200 for development set and 800 for the test set. They were explained how the dataset was created and asked not to look for the answer on the internet. We rejected human 1 result based on the development dataset result and created a human ensemble with the remaining 3 people predictions using majority voting. The results are presented with the best TF-IDF based, self-contained and overall methods in the Table II.

B. TF-IDF based methods

Term frequency-inverse document frequency (TF-IDF) is a common vector representation of a document in natural language processing. We use the TfidfVectorizer library from Scikit-learn with standard parameters. This includes word-level, lowercasing, $l2$ normalization. We did not restrict the vocabulary size and we used word-level splitting. The following classifiers were trained using TF-IDF vectors: Logistic Regression, XGBoost Classifier, SVM.

1) Logistic Regression: We used LogisticRegression from Scikit-learn library with standard parameters, except for the maximum number of iteration. We trained until classifier convergence.

2) Support Vector Machine Classifier: Support-Vector Network [5] is a common algorithm, that circumvents non-linear separability of data as well as separate samples from different categories. Although, in this case, we chose LinearSVC from Scikit-Learn, which uses a linear kernel. The reason is memory and computation issues related to the high dimension of TF-IDF representation and the number of samples in the HSSS task. Again, we used standard parameters, except for no maximum number of iteration, which led to convergence. We do not report likelihood due to the fact that SVM does not yield probabilities.

3) XGBoost Classifier: Tree boosting is an effective and popular method for regression and classification. We used XGboost library [3] with the choice of the parameters suited for better classifier quality.[1] This includes gbtree booster,

[1] Some of the parameters were taken from https://www.kaggle.com/serigne /stacked-regressions-top-4-on-leaderboard

learning rate set to 0.05 and max depth set to 3.

C. Neural Methods (self-contained)

1) FastText: FastText [15] is a shallow neural network library created for fast text classification model training and evaluation. We used a supervised setting with hyperparameter tuning, the word embeddings were initialized randomly. The best result was obtained with wordNgrams set to 2, word dimension set to 156, and context size window set to 5.

2) LSTM: Long Short Term Memory Networks [13] were used to obtain a state-of-the-art results on most NLP tasks before the era of Transformer language models [7]. In our tasks, for bidirectional LSTM, SentencePiece [19] tokenization performs better than word-level lowercase tokenization. Vocab size 50k was used with randomly initialized embeddings of size 100. We tried embedding size 300, but resulted in slightly worse classifier quality. We used one layer of 256 units, trained with Adam [16] optimizer with learning rate 0.001. The batch size used for training was 400 and sequences were trimmed and padded to 256 tokens.

3) Transformer: In the last time Transformer [26] and its modification like BERT [7], RoBERTa [20] or XLM-R [4] achieve state-of-the-art in the benchmarks such as GLUE [29] or SuperGLUE [28] benchmark. Most often used bidirectional Transformers are pre-trained on huge amounts of monolingual data in the Masked Language Model (MLM) process, where the model learns a bidirectional representation of tokens. Next, pre-trained models are finetuned to the specific task. This process reduces the time to train a new model from scratch and can be easily adapted to other tasks. In our case, the downstream task is classification, where the model uses a special token ([CLS], classification token), which represents the whole sentence and helps achieve better results.

We train self-contained classifier based on the RoBERTa model in two ways: with pre-training and without pre-training (train classifier from the scratch) stage. We only used the data that was available in the HSSS challenge to avoid any data leaks in the other data sets. To compare our methods we created Transformer with 8 layers, 8 heads, 256 sequence length and embedding size 512 and 2048 respectively for internal model representation and feed forward layer (after attention layer). We use 50k size vocabulary with Sentencepiece tokenization and randomly initialized embeddings of size 512. First, the model was pre-trained for 10 epochs with Masked Language Model (MLM) criterion and finetuned 10 epochs for the classification tasks. Second, the model was trained on the classification task for 20 epochs (comparing to the previous one, where it was 10 + 10 epochs for pre-training and classification) only. We pre-train and finetune with Adam optimizer with learning rate 0.0001 and 50 sentences per batch. Scores presented in the Table III show that the pre-training stage is the important element to achieve a better model for classification tasks.

D. Pre-trained Transformers

In this section we describe fine-tuning of models publicly available for Polish language: Polish RoBERTa [6] and multilingual XLM-R [4] (which supports 100 languages including Polish). Both models are available in the two versions: base (with 12 layers) and large (with 24 layers). Monolingual models like RoBERTa are focused on achieving the best results in a given language. On the other hand, multilingual models support as many languages as possible with results similar to monolingual models. The disadvantage of multilingual models is the size of the vocabulary, which is several times larger than monolingual models like Polish RoBERTa. Bigger vocabulary needs more resources to fine-tune models, but may improve results by cross-language relationships.

1) Polish RoBERTa finetuning: We finetuned Polish RoBERTa [6] (base and large model) using fairseq library [22] for 5 and 3 epochs respectively for the base and large model. Further training resulted in lower development dataset accuracy. We used Adam optimizer with a learning rate 0.00001 and around 200k warmup steps. The maximum sequence we use is 512 as in original Polish RoBERTa.

2) Polish RoBERTa finetuning with Monte-Carlo model averaging: Common practice when using dropout is to scale weights during inference time. However, as described in [25] (section 7.5), further investigated in [8], this procedure is only an approximation of Monte-Carlo model averaging. We checked, whether the Monte-Carlo model averaging yields better results than standard weight scaling in our case. By setting Polish RoBERTa (both base and large) in the training mode (with active dropout), making predictions 12 times, and averaging likelihood, we obtained slightly better results in both cases.

3) XLM-R finetuning: We finetuned multilingual XLM-R [4] base and large for 1 epoch, further training does not improve results. Each of the models was trained with 512 tokens using Adam optimizer with a learning rate 0.00004. Batch size has been set to 10 and 25 for the base and large model. Results are available in the Table III.

4) Polish RoBERTa last layer averaged: For the evaluation of how much information about language Polish RoBERTa possesses, we conducted the following experiment. We extracted the last layer tokens and averaged them. Then, we trained logistic regression classifier with no Polish RoBERTa finetuning. This was done until classifier convergence.

5) XLM-R last layer averaged: We conducted the same experiment with XLM-R as in subsection III-D4.

6) Polish RoBERTa fill mask: In order to check the predicting power of only pre-trained Polish RoBERTa models, we conducted the following experiment. We masked all gender-revealing first-person expression and used the models in Masked Language Model setting. We choose one random expression and looked for the most probable word indicating gender in the first 10 model predictions. Only 6333 samples out of 137314 in the test set did not reveal first-person expression in the first 10 predictions. No training or development sets were used in this experiment. However, this method does not yield good results (though the trivial baseline was beaten).

IV. Results

The self-contained models (BiLSTM and RoBERTa MLM + classifier) achieved better results than TF-IDF and fastText. The BiLSTM model achieves a bit better results than the Transformer base model, which suggests that the Transformer model needs more resources. The classifier trained from scratch (without pre-training) produces inferior results, and this shows again that the pre-training step is an important element in classification tasks. Neural methods achieve better results than the human baseline, but human results are comparable to TF-IDF.

Pre-trained models trained on the much larger data set than the HSSS data set achieve the best results. Monolingual and multilingual models achieve similar results, but XLM-R large achieve lower results than other pre-trained models, indicating that the bigger models may not improve results on the classification tasks. Polish RoBERTa large achieved similar results to the base version, which might mean that RoBERTa large needs more pre-training steps to get better results.

V. Contamination study

Using a pre-trained language model (or any other solution not constrained to the train set provided with the challenge) raises the question of data contamination or train-test overlap, i.e. (1) was the test set represented in the training set of the language model?, (2) did it make the results better (e.g. due to memorization of test texts by the language model)? See [1] for the discussion of data contamination in the case of the GPT-3 model when used for popular English NLP test sets.

We carried out a contamination study on the solution based on the Polish RoBERTa model (the best solution so far). As the Polish RoBERTa was trained (among other sources) on CommonCrawl 2019/2020 [6], and the HSSS was prepared using CommonCrawl 2012-2015 (mostly 2012), the risk of contamination was real (a significant percentage of Web content from 2012-2015 could survive up to 2019).

We searched the contents of CommonCrawl 2019 (as provided to us by the authors of [6][2]) for the six-gram fragments of the HSSS test set, obviously taking into account the fact that feminine gender-specific forms were modified during the preparation of the HSSS test set.

The summary of the contamination study is given in Table IV, where the results obtained with Polish RoBERTa are compared against the best constrained solution (an LSTM trained on the HSSS training set). The following conclusions can be made:

- results on the contaminated subset *are* better (and the difference of the Accuracy/Likelihood metrics on the contamination and not contaminated metric is significant), and this might indicate that the problem is real;
- still, the percentage of data contaminated is low (3%), hence the impact on the total is limited; if we were to lower the results on the contaminated subset to be the same as on the uncontaminated subset, the accuracy would be lower only by a small margin;
- note that this is not a proof of contamination; the cause of better results on the contaminated subset might be different, for example it might have been caused by the fact that CommonCrawl 2019 for Polish RoBERTa was filtered by a language model, whereas for the HSSS data set — only using handcrafted heuristics, i.e. sentences might be longer and “proper” (e.g. say with fewer spam texts), hence easier for a classification task.

VI. Conclusions

We showed that a pre-trained Transformer model can obtain strong results for a challenging classification tasks on short texts. It turned out that predictions done by humans (even aggregated) were much worse. What is important is that influence of contamination of the training set was practically excluded.

References

[1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[2] B. Bsir and M. Zrigui. Bidirectional LSTM for author gender identification. In *International Conference on Computational Collective Intelligence*, pages 393–402. Springer, 2018.

[3] T. Chen and C. Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD ’16, pages 785–794, New York, NY, USA, 2016. ACM.

[4] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*, 2019.

[5] C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[6] S. Dadas, M. Perełkiewicz, and R. Poświata. Pre-training Polish Transformer-Based language models at scale. In L. Rutkowski, R. Scherer, M. Korytkowski, W. Pedrycz, R. Tadeusiewicz, and J. M. Zurada, editors, *Artificial Intelligence and Soft Computing*, pages 301–314, Cham, 2020. Springer International Publishing.

[7] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805, 2018.

[8] Y. Gal and Z. Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. *Proceedings of The 33rd International Conference on Machine Learning*, 06 2015.

[9] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision, 2009.

[10] F. Graliński, Ł. Borchmann, and P. Wierzchoń. “He Said She Said” — a male/female corpus of Polish. In N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, and S. Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation* (*LREC 2016*), Paris, France, 2016. European Language Resources Association (ELRA).

[11] F. Graliński, R. Jaworski, Ł. Borchmann, and P. Wierzchoń. Gonito.net – open platform for research competition, cooperation and reproducibility. In A. Branco, N. Calzolari, and K. Choukri, editors, *Proceedings of the 4REAL Workshop*, pages 13–20. 2016.

[12] F. Graliński, R. Jaworski, Ł. Borchmann, and P. Wierzchoń. Vive la petite différence! Exploiting small differences for gender attribution of short texts. *Lecture Notes in Artificial Intelligence*, 9924:54–61, 2016.

[13] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, Nov. 1997.

[2] Unfortunately, we were unable to check the other sources, though the probability of them contaminating the test set seems much lower

TABLE III
Results. *Human baseline was evaluated only on the random sample of size 800. †References to repositories at Gonito.net [11] are given in curly brackets. Such a repository may be also accessed by going to http://gonito.net/q and entering the code there.

method	test likelihood	test accuracy	gonito submission†
human baseline*	0.00000	0.68125	{87a138}
TF-IDF + logistic regression	0.55278	0.67175	{ecc1ee}
TF-IDF + linear SVM	0.00000	0.66477	{da348f}
TF-IDF + XGBClassifier	0.54269	0.65112	{5a17c9}
fastText	0.54541	0.67448	{4d18c0}
Bi-LSTM	0.57177	0.69786	{a0d38c}
RoBERTa MLM + classifier	0.57068	0.69153	{203325}
RoBERTa classifier (only)	0.55784	0.67951	{6756e6}
Polish RoBERTa (base) finetuned	0.60913	0.74185	{049966}
Polish RoBERTa (large) finetuned	0.60503	0.74388	{2b8541}
XLM-R (base) finetuned	0.60015	0.72356	{bdac6e}
XLM-R (large) finetuned	0.57141	0.69047	{bdac6e}
Polish RoBERTa (base) active dropout	**0.62110**	0.74332	{ea4b15}
Polish RoBERTa (large) active dropout	0.61949	**0.74406**	{2e89da}
Polish RoBERTa (large) last layer + logic regression	0.54113	0.65956	{582542}
XLM-R (large) last layer + logic regression	0.54067	0.65545	{115246}
Polish RoBERTa (large) fill mask	0.00000	0.55828	{11633b}

TABLE IV
Contaminated vs not contaminated subset of the test set. P-values are calculated with the Mann–Whitney U test.

		contaminated	not-contaminated	all	p-value
items	#	4,076	130,542	134,618	
	%	3.0%	97.0%	100.0%	
Polish RoBERTa base	Likelihood	0.64656	0.62032	0.62110	0.0000
	Accuracy	0.77159	0.74244	0.74332	0.0007
LSTM (constrained)	Likelihood	0.58305	0.57142	0.57177	0.0000
	Accuracy	0.70118	0.69776	0.69786	0.3549

[14] S. Hussein, M. Farouk, and E. Hemayed. Gender identification of Egyptian dialect in Twitter. *Egyptian Informatics Journal*, 20(2):109–116, 2019.

[15] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, 2016.

[16] D. Kingma and J. Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.

[17] D. Kodiyan, F. Hardegger, S. Neuhaus, and M. Cieliebak. Author Profiling with bidirectional RNNs using Attention with GRUs: Notebook for PAN at CLEF 2017. In *CLEF 2017 Evaluation Labs and Workshop–Working Notes Papers, Dublin, Ireland, 11-14 September 2017*, volume 1866. RWTH Aachen, 2017.

[18] S. Krüger and B. Hermann. Can an online service predict gender? On the state-of-the-art in gender identification from texts. In *2019 IEEE/ACM 2nd International Workshop on Gender Equality in Software Engineering (GE)*, pages 13–16. IEEE, 2019.

[19] T. Kudo and J. Richardson. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In E. Blanco and W. Lu, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018: System Demonstrations, Brussels, Belgium, October 31 - November 4, 2018*, pages 66–71. Association for Computational Linguistics, 2018.

[20] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.11692, 2019.

[21] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*, pages 1003–1011. Association for Computational Linguistics, 2009.

[22] M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.

[23] J. Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL student research workshop*, pages 43–48, 2005.

[24] A. Siewierska. Gender distinctions in independent personal pronouns. In M. S. Dryer and M. Haspelmath, editors, *The World Atlas of Language Structures Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig, 2013.

[25] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, Jan. 2014.

[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[27] R. Veenhoven, S. Snijders, D. van der Hall, and R. van Noord. Using translated data to improve deep learning author profiling models. In *Proceedings of the Ninth International Conference of the CLEF Association (CLEF 2018)*, volume 2125, 2018.

[28] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *arXiv preprint 1905.00537*, 2019.

[29] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. 2019. In the Proceedings of ICLR.