NLytics at CheckThat! 2021: Check-Worthiness Estimation as a Regression Problem on Transformers

Albert Pritzkau[1]

[1]*Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE, Fraunhoferstraße 20, 53343 Wachtberg, Germany*

Abstract

The following system description presents our approach to the estimation of check-worthiness of text chunks. The given task has been framed as a regression problem. In order predict a numerical value for a chunk we opted for RoBERTa (A Robustly Optimized BERT Pretraining Approach) as a neural network architecture for sequence classification. Starting off with a pre-trained model for language representation we fine-tuned this model on the given classification task with the provided annotated data in supervised training steps.

Keywords

Ranking, Regression, Deep Learning, Transformers, RoBERTa

1. Introduction

The proliferation of disinformation online, has given rise to a lot of research on automatic fake news detection. CLEF 2021 - CheckThat! Lab [1, 2] considers disinformation as a communication phenomenon. Instead of categorizing posts into categories such as "fake" or "non-fake", the task at hand is to evaluate and rank posts based on the credibility of the content. By detecting the use of various linguistic features in communication, it takes into account not only the content but also how a subject matter is communicated. The shared task [3] defines the following subtasks given in English:

Subtask A Given a tweet, predict whether it is worth fact-checking.

Subtask B Given a political debate/speech, produce a ranked list of its sentences, ordered by their check-worthiness.

The aim of any information retrieval (IR) system is to find relevant documents. A huge amount of research has been spent on relevance estimation as a central concept of IR. In addition, corresponding methods are widely used in particular for ranking search results. In this work, we covered our approach on both relevance estimation tasks by framing it as a regression problem, knowing that, this way, the overall problem is strongly approximated. To build our

CLEF 2021 – Conference and Labs of the Evaluation Forum, September 21–24, 2021, Bucharest, Romania

albert.pritzkau@fkie.fraunhofer.de (A. Pritzkau)
https://www.fkie.fraunhofer.de/ (A. Pritzkau)
0000-0001-7985-0822 (A. Pritzkau)

models, both subtasks assumes purely textual content as inputs. Below, we will describe the system built for both two subtasks. At the core of our systems is RoBERTa [4], a pre-trained model based on the Transformer architecture [5].

2. Related Work

The goal of the shared task is to investigate automatic techniques for identifying various rhetorical and psychological features of disinformation campaigns. A comprehensive survey on fake news has been presented by Zhou and Zafarani [6]. Based on the structure of data reflecting different aspects of communication, they identified four different perspectives on fake news: (1) the false knowledge it carries, (2) its writing style, (3) its propagation patterns, and (4) the credibility of its creators and spreaders.

The shared task emphasizes communicative styles that systematically co-occur with persuasive intentions of (political) media actors. Similar to de Vreese et al. [7], propaganda and persuasion is considered as an expression of political communication content and style. Hence, beyond the actual subject of communication, the way it is communicated is gaining importance.

We build our work on top of this foundation by first investigating content-based approaches for information discovery. Traditional information discovery methods are based on content: documents, terms, and the relationships between them. They can be considered as a general Information Extraction (IE) methods, automatically deriving structured information from unstructured and/or semi-structured machine-readable documents. Communities of researchers contributed various techniques from machine learning, information retrieval, and computational linguistics to the different aspects of the information extraction problem. From a computer science perspective, existing approaches can be roughly divided into the following categories: rule-based, supervised, and semi-supervised. In our case, we followed the supervised approach by reframing the complex language understanding task as a simple classification problem. Text classification also known as text tagging or text categorization is the process of categorizing text into organized groups. By using Natural Language Processing (NLP), text classifiers can automatically analyze human language texts and then assign a set of predefined tags or categories based on their content. Historically, the evolution of text classifiers can be divided into three stages: (1) simple lexicon- or keyword-based classifiers, (2) classifiers using distributed semantics, and (3) deep learning classifiers with advanced linguistic features.

2.1. Deep Learning for Information Extraction

Recent work on text classification uses neural networks, particularly Deep Learning (DL). Badjatiya et al. [8] demonstrated that these architectures, including variants of recurrent neural networks (RNN) [9, 10, 11], convolutional neural networks (CNN) Zhang et al. [12], or their combination (CharCNN, WordCNN, and HybridCNN), produce state-of-the-art results and outperform baseline methods (character n-grams, TF-IDF or bag-of-words representations).

2.2. Deep Learning architectures

Until recently, the dominant paradigm in approaching NLP tasks has been focused on the design of neural architectures, using only task-specific data and word embeddings such as those mentioned above. This led to the development of models, such as Long Short Term Memory (LSTM) networks or Convolution Neural Networks (CNN), that achieve significantly better results in a range of NLP tasks than less complex classifiers, such as Support Vector Machines, Logistic Regression or Decision Tree Models. Badjatiya et al. [8] demonstrated that these approaches outperform models based on char and word n-gram representations. In the same paradigm of pre-trained models, methods like BERT [13] and XLNet [14] have been shown to achieve the state of the art in a variety of tasks.

2.3. Pre-trained Deep Language Representation Model

Indeed, the usage of a pre-trained word embedding layer to map the text into vector space which is then passed through a neural network, marked a significant step forward in text classification. The potential of pre-trained language models, as e.g. Word2Vec [15], GloVe [16], fastText [17], or ELMo [18] to capture the local patterns of features to benefit text classification, has been described by Castelle [19]. Modern pre-trained language models use unsupervised learning techniques such as creating RNNs embeddings on large texts corpora to gain some primal "knowledge" of the language structures before a more specific supervised training steps in.

2.4. About BERT and RoBERTa

BERT stands for Bidirectional Encoder Representations from Transformers. It is based on the Transformer model architectures introduced by Vaswani et al. [5]. The general approach consists of two stages: first, BERT is pre-trained on vast amounts of text, with an unsupervised objective of masked language modeling and next-sentence prediction. Second, this pre-trained network is then fine-tuned on task specific, labeled data. The Transformer architecture is composed of two parts, an Encoder and a Decoder, for each of the two stages. The Encoder used in BERT is an attention-based architecture for NLP. It works by performing a small, constant number of steps. In each step, it applies an attention mechanism to understand relationships between all words in a sentence, regardless of their respective position. By pre-training language representations, the Encoder yields models that can either be used to extract high quality language features from text data, or fine-tune these models on specific NLP tasks (classification, entity recognition, question answering, etc.). We rely on RoBERTa [4], a pre-trained Encoder model which builds on BERT's language masking strategy. However, it modifies key hyperparameters in BERT such as removing BERT's next-sentence pre-training objective, and training with much larger mini-batches and learning rates. Furthermore, RoBERTa was also trained on an order of magnitude more data than BERT, for a longer amount of time. This allows RoBERTa representations to generalize even better to downstream tasks compared to BERT. In this study, RoBERTa is at the core of each solution of the given subtasks.

Table 1
Statistical summary of token counts on the training set.

Source	Tweets (A)			Debates/Speeches (B)		
	Label	0	1	Label	0	1
doc count	822	532	290	42033	41604	429
mean	31.68	28.67	37.20	12.55	12.48	19.87
std	14.23	15.06	10.54	10.65	10.59	13.27
min	3	3	11	1	1	1
25%	19	14	30	5	5	10
50%	34	28	40	9	9	17
75%	44	43	44	17	17	26
max	56	56	53	138	138	91

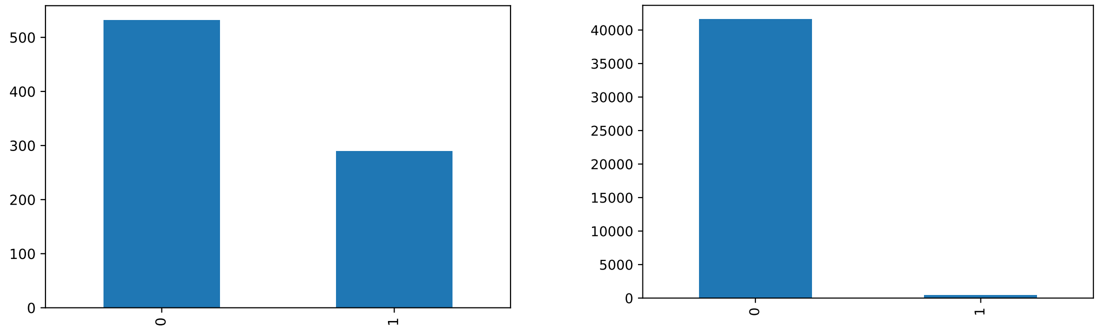
3. Dataset

The data for the task was developed during the CLEF-2021 CheckThat! campaign [1, 2, 3]. For subtask A, the organizers provided a training set of 822 tweets collected from a variety of COVID-19- related topics. For subtask A, the organizers provided a training set of 822 tweets collected from a variety of COVID-19- related topics. The provided validation set contains 140 tweets. For subtask B, the training set contain 42033 statements collected from a variety of political debates/speeches. The provided validation set for this subtask contains 3586 records. The individual statements from both sources are annotated and considered as check-worthy if it contains a verifiable factual claim. Check-worthy statements are labeled 1, all others are labeled 0.

3.1. Exploratory data analysis

As presented in Table 1, the given training sets for the individual tasks differ significantly in scope. The collection contain 822 and 42033 entries, respectively. The length of the individual statements is far below the limit of BERT-based model of 512 tokens to be processed. Both datasets are therefore not subject to the restriction of the input length of the model used. The mean token count of each statement, however, differs significantly between the two tasks and amounts to about half of the tokens in the statements of the debates (12.55) compared to those of tweets (31.68). This may have implications for the performance of attention-based models, which primarily rely on sequence information.

Unbalanced class distribution Imbalance in data can exert a major impact on the value and meaning of accuracy and on certain other well-known performance metrics of an analytical model. Figure 1 depicts a clear skew towards information classified as not check-worthy. This is be especially true in the case of the debate chunk labels.



(a) tweet labels

(b) debate chunk labels

Figure 1: Label distribution - training set

3.2. Evaluation measures

For both tasks the submitted ranked lists per claim have been evaluated using ranking evaluation measures MAP (Mean Average Precision), MRR (Mean Reciprocal Rank), RP (R-precision) and Precision@k for $k \in \{1, 3, 5, 10, 10, 20, 30\}$ (Precision for the top-k documents). The MAP score has been defined as the official evaluation measure to rank the submissions.

4. Our approach

In this section, we provide a general overview of our approach to both subtasks.

4.1. Experimental setup

Model Architecture Subtasks A and B are both evaluated as a ranking task. Our model for this subtask is based on RoBERTa. For the classification task, fine-tuning is performed using *RobertaForSequenceClassification*[20] – roberta-base – as the pre-trained model. *RobertaForSequenceClassification* optimizes for Binary Cross Entropy Loss using an AdamW optimizer with an initial learning rate set to 2e-5. Fine-tuning is done on NVIDIA TESLA P100 GPU using the Pytorch [21] framework with a vocabulary size of 50265 and an input size of 512. The model is trained to optimize the objective for 20 epochs. The submission for each subtask is based on the best performing model checkpoint on the validation set.

Input Embeddings The input embedding layer converts the inputs into sequences of features: word-level sentence embeddings. These embedding features will be further processed by the latter encoding layers.

Word-Level Sentence Embeddings A sentence is split into words $w_1, ..., w_n$ with length of n by the WordPiece tokenizer [22]. The word w_i and its index i (w_i's absolute position in the sentence) are projected to vectors by embedding sub-layers, and then added to the index-aware word embeddings:

$$\hat{w}_i = WordEmbed(w_i)$$

$$\hat{u}_i = IdxEmbed(i)$$

$$h_i = LayerNorm(\hat{w}_i + \hat{u}_i)$$

Attention Layers Attention layers [23, 24] aim to retrieve information from a set of context vectors y_j related to a query vector x. An attention layer first calculates the matching score a_j between the query vector x and each context vector y_j. Scores are then normalized by softmax:

$$a_j = score(x, y_j)$$

$$\alpha_j = exp(a_j)/\Sigma_k exp(a_k)$$

The output of an attention layer is the weighted sum of the context vectors w.r.t. the softmax normalized score: $Att_{X \to Y}(x, \{y_j\}) = \Sigma_j \alpha_j y_j$. An attention layer is called self-attention when the query vector x is in the set of context vectors y_j. Specifically, we use the multi-head attention following Transformer [5].

Target Encoding The goal of regression is to predict a single, continuous target value for each example in the dataset. A transformer-based regression model typically consists of a transformer model with a fully-connected layer on top of it. Setting the number of labels to one, the fully-connected layer will have a single output neuron which predicts the target value. To perform regression, thus, is just a matter of changing the loss function. The classifier is replaced with a regressor for the error to be propagated to the rest of the network.

4.2. Results and Discussion

We participated in both ranking tasks in English. Official evaluation results on the test set are presented in Table 2 and Table 3 for each subtask, respectively. Both tables contain a ngram-baseline submission from the competition organizers.

We focused on suitable combinations deep learning methods as well as their hyperparameter settings. Finetuning pre-trained language models like RoBERTa on downstream tasks has become ubiquitous in NLP research and applied NLP. The submission for each subtask is based on the best performing model checkpoint on the validation set. MSE (Mean Squared Error) was used as evaluation measure to estimate the performance with a lowest value of 0.24017 and 0.01887 for each subtask, respectively.

When improving on these baseline models, class imbalance appears to be a primary challenge. This is clearly reflected in Figures 1 and 2, in particular, for the second subtask on chunks of speeches and debates. In our case, it results in no check-worthy claim at all being found in the test data. Based on our model, in the case of tweets, far more posts are classified as check-worthy than was predetermined by the gold standard.

A commonly used tactic to deal with imbalanced datasets is to assign weights to each label. Alternative solutions for coping with unbalanced supervised dataset are undersampling or oversampling. Undersampling only considers a subset of an overpopulated class to end up with

Table 2
Results on the test set on subtask A (tweets).

Rank	Team	MAP	MRR	RP	P@1	P@3	P@5	P@10	P@20	P@30
1	NLP&IR@UNED	0.224	1.000	0.211	1.000	0.667	0.400	0.300	0.200	0.160
2	Fight for 4230	0.195	0.333	0.263	0.000	0.333	0.400	0.400	0.250	0.160
3	ibaris	0.149	1.000	0.105	1.000	0.333	0.200	0.200	0.100	0.120
4	bigIR	0.136	0.500	0.105	0.000	0.333	0.200	0.100	0.100	0.120
5	Team GPLSI	0.132	0.167	0.158	0.000	0.000	0.000	0.200	0.150	0.140
6	csum112	0.126	0.250	0.158	0.000	0.000	0.200	0.200	0.150	0.160
7	abaruah	0.121	0.200	0.158	0.000	0.000	0.200	0.200	0.200	0.140
8	**NLytics**	**0.111**	**0.071**	**0.053**	**0.000**	**0.000**	**0.000**	**0.000**	**0.050**	**0.120**
9	Accenture	0.101	0.143	0.158	0.000	0.000	0.000	0.200	0.200	0.100
10	TOBB ETU	0.081	0.077	0.053	0.000	0.000	0.000	0.000	0.050	0.080
	ngram-baseline	0.052	0.020	0.000	0.000	0.000	0.000	0.000	0.000	0.020

Table 3
Results on the test set on subtask B (speeches/debates).

Rank	Team	MAP	MRR	RP	P@1	P@3	P@5	P@10	P@20	P@30
1	Fight for 4230	0.402	0.917	0.403	0.875	0.833	0.750	0.600	0.475	0.350
	ngram-baseline	0.235	0.792	0.263	0.625	0.583	0.500	0.400	0.331	0.217
2	**NLytics**	**0.135**	**0.345**	**0.130**	**0.250**	**0.125**	**0.100**	**0.137**	**0.156**	**0.135**

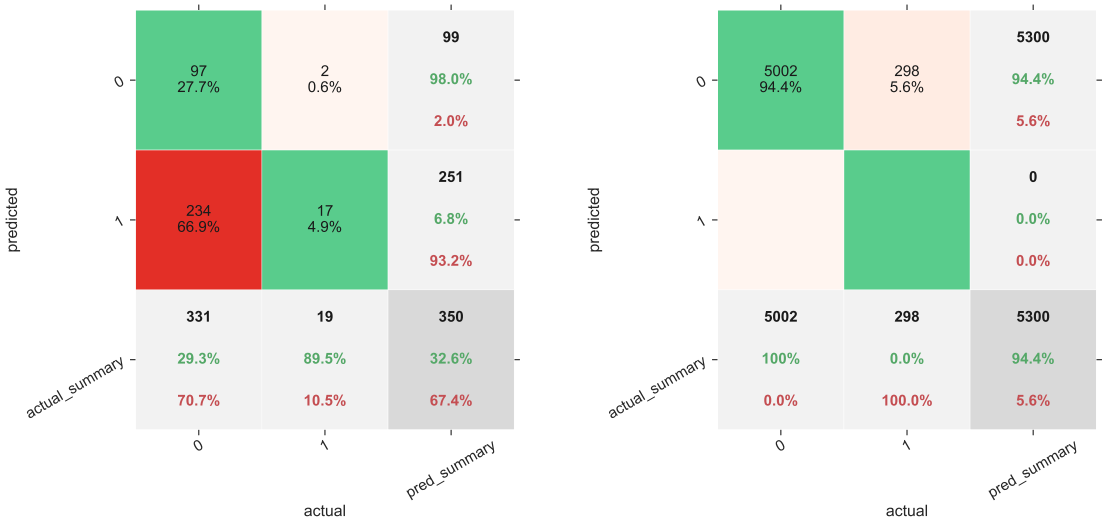
a balanced dataset. With the same goal oversampling creates copies of the unbalanced classes. However, it remains questionable whether the measures to tackle class imbalance will lead to success, since we expect additional confounding issues. The highly adversial and task-specific nature of the relevance criterion of check-worthiness, may have a counterproductive effect on the goal of generalizability of the language representations of transformer-based models. Thus, overfitting poses the most difficult challenge in this experiment, reducing its generalizability.

In addition to imbalance, document length is a potential confounding factor. The significant difference in the mean token count seems to be reflected in the classification result not even passing the given ngram-baseline on subtask B. Since the classification features are derived primarily from sequence information, we assume that this has a decisive influence on the result.

5. Conclusion and Future work

In future work, we plan to investigate more recent neural architectures for language representation such as T5 [25] and GPT-3 [26].

Furthermore, we expect great opportunities for transfer learning from the areas such as argumentation mining [27] and offensive language detection [28]. To deal with data scarcity as a general challenge in natural language processing, we examine the application of concepts such as active learning, semi-supervised learning [29] as well as weak supervision [30].



(a) tweets

(b) speeches/debates

Figure 2: Confusion matrix for each subtask on the test set compared to the gold standard.

References

[1] P. Nakov, G. Da San Martino, T. Elsayed, A. Barrón-Cedeño, R. Míguez, S. Shaar, F. Alam, F. Haouari, M. Hasanain, N. Babulkov, A. Nikolov, G. K. Shahi, J. M. Struß, T. Mandl, The CLEF-2021 CheckThat! Lab on Detecting Check-Worthy Claims, Previously Fact-Checked Claims, and Fake News, in: Proceedings of the 43rd European Conference on Information Retrieval, ECIR˜'21, Lucca, Italy, 2021, pp. 639–649. URL: https://link.springer.com/chapter/10.1007/978-3-030-72240-1{_}75.

[2] P. Nakov, G. Da San Martino, T. Elsayed, A. Barrón-Cedeño, R. Míguez, S. Shaar, F. Alam, F. Haouari, M. Hasanain, N. Babulkov, A. Nikolov, G. K. Shahi, J. M. Struß, T. Mandl, S. Modha, M. Kutlu, Y. S. Kartal, Overview of the CLEF-2021 CheckThat! Lab on Detecting Check-Worthy Claims, Previously Fact-Checked Claims, and Fake News, in: Proceedings of the 12th International Conference of the CLEF Association: Information Access Evaluation Meets Multiliguality, Multimodality, and Visualization, CLEF˜'2021, Bucharest, Romania (online), 2021.

[3] S. Shaar, M. Hasanain, B. Hamdan, Z. S. Ali, F. Haouari, M. K. Alex Nikolov, F. A. Yavuz Selim Kartal, G. Da San Martino, A. Barrón-Cedeño, R. Míguez, T. Elsayed, P. Nakov, Overview of the CLEF-2021 CheckThat! Lab Task 1 on Check-Worthiness Estimation in Tweets and Political Debates, in: Working Notes of CLEF 2021—Conference and Labs of the Evaluation Forum, CLEF˜'2021, Bucharest, Romania (online), 2021.

[4] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A robustly optimized BERT pretraining approach, 2019. arXiv:1907.11692.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polo-

sukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, volume 2017-Decem, 2017, pp. 5999–6009. arXiv:1706.03762.

[6] X. Zhou, R. Zafarani, Fake News: A Survey of Research, Detection Methods, and Opportunities, ACM Comput. Surv 1 (2018). arXiv:1812.00315.

[7] C. H. de Vreese, F. Esser, T. Aalberg, C. Reinemann, J. Stanyer, Populism as an Expression of Political Communication Content and Style: A New Perspective, International Journal of Press/Politics 23 (2018) 423–438. URL: http://journals.sagepub.com/doi/10.1177/1940161218790035. doi:10.1177/1940161218790035.

[8] P. Badjatiya, S. Gupta, M. Gupta, V. Varma, Deep learning for hate speech detection in tweets, in: 26th International World Wide Web Conference 2017, WWW 2017 Companion, International World Wide Web Conferences Steering Committee, 2017, pp. 759–760. doi:10.1145/3041021.3054223. arXiv:1706.00188.

[9] L. Gao, R. Huang, Detecting online hate speech using context aware models, in: International Conference Recent Advances in Natural Language Processing, RANLP, volume 2017-Septe, Association for Computational Linguistics (ACL), 2017, pp. 260–266. doi:10.26615/978-954-452-049-6-036. arXiv:1710.07395.

[10] J. Pavlopoulos, P. Malakasiotis, I. Androutsopoulos, Deeper attention to abusive user content moderation, in: EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings, Association for Computational Linguistics, Stroudsburg, PA, USA, 2017, pp. 1125–1135. URL: http://aclweb.org/anthology/D17-1117. doi:10.18653/v1/d17-1117.

[11] G. K. Pitsilis, H. Ramampiaro, H. Langseth, Effective hate-speech detection in Twitter data using recurrent neural networks, Applied Intelligence 48 (2018) 4730–4742. doi:10.1007/s10489-018-1242-y. arXiv:1801.04433.

[12] Z. Zhang, D. Robinson, J. Tepper, Detecting Hate Speech on Twitter Using a Convolution-GRU Based Deep Neural Network, in: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), volume 10843 LNCS, Springer Verlag, 2018, pp. 745–760. doi:10.1007/978-3-319-93417-4_48.

[13] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (2018). arXiv:1810.04805.

[14] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, Q. V. Le, XLNet: Generalized Autoregressive Pretraining for Language Understanding, Technical Report, 2019. arXiv:1906.08237.

[15] T. Mikolov, Q. V. Le, I. Sutskever, Exploiting Similarities among Languages for Machine Translation (2013). arXiv:1309.4168.

[16] J. Pennington, R. Socher, C. D. Manning, GloVe: Global vectors for word representation, in: EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 2014, pp. 1532–1543. doi:10.3115/v1/d14-1162.

[17] A. Joulin, E. Grave, P. Bojanowski, T. Mikolov, Bag of tricks for efficient text classification, in: 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of Conference, volume 2, 2017, pp. 427–431. URL: https://github.com/facebookresearch/fastText. doi:10.18653/v1/e17-2068. arXiv:1607.01759.

[18] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep Contextualized Word Representations, Association for Computational Linguistics (ACL), 2018, pp. 2227–2237. doi:10.18653/v1/n18-1202. arXiv:1802.05365.

[19] M. Castelle, The Linguistic Ideologies of Deep Abusive Language Classification, 2019, pp. 160–170. doi:10.18653/v1/w18-5120.

[20] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, A. Rush, Transformers: State-of-the-Art Natural Language Processing, in: arxiv.org, 2020, pp. 38–45. URL: https://github.com/huggingface/. doi:10.18653/v1/2020.emnlp-demos.6. arXiv:1910.03771v5.

[21] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems, volume 32, Neural information processing systems foundation, 2019. URL: http://arxiv.org/abs/1912.01703. arXiv:1912.01703.

[22] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, Ł. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, J. Dean, Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation (2016). arXiv:1609.08144.

[23] D. Bahdanau, K. H. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in: 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, International Conference on Learning Representations, ICLR, 2015. arXiv:1409.0473.

[24] K. Xu, J. L. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. S. Zemel, Y. Bengio, Show, attend and tell: Neural image caption generation with visual attention, in: 32nd International Conference on Machine Learning, ICML 2015, volume 3, International Machine Learning Society (IMLS), 2015, pp. 2048–2057. arXiv:1502.03044.

[25] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, arXiv 21 (2019) 1–67. arXiv:1910.10683.

[26] 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, D. Amodei, Language models are few-shot learners, 2020. arXiv:2005.14165.

[27] M. Stede, Automatic argumentation mining and the role of stance and sentiment, Journal of Argumentation in Context 9 (2020) 19–41. URL: https://www.jbe-platform.com/content/journals/10.1075/jaic.00006.ste. doi:10.1075/jaic.00006.ste.

[28] M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, R. Kumar, Predicting the type and target of offensive posts in social media, in: NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, volume 1, Association

for Computational Linguistics, Stroudsburg, PA, USA, 2019, pp. 1415–1420. URL: http://aclweb.org/anthology/N19-1144. doi:10.18653/v1/n19-1144. arXiv:1902.09666.

[29] S. Ruder, B. Plank, Strong Baselines for Neural Semi-supervised Learning under Domain Shift, ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers) 1 (2018) 1044–1054. arXiv:1804.09530.

[30] A. Ratner, S. H. Bach, H. Ehrenberg, J. Fries, S. Wu, C. Ré, Snorkel: rapid training data creation with weak supervision, in: VLDB Journal, volume 29, Springer, 2020, pp. 709–730. doi:10.1007/s00778-019-00552-1.