NLyticsFKIE at SemEval-2021 Task 6: Detection of Persuasion Techniques In Texts And Images

Albert Pritzkau
Fraunhofer FKIE / Wachtberg, Germany
albert.pritzkau@fkie.fraunhofer.de

Abstract

The following system description presents our approach to the detection of persuasion techniques in texts and images. The given task has been framed as a multi-label classification problem with the different techniques serving as class labels. The multi-label classification problem is one in which a list of target variables such as our class labels is associated with every input chunk and assumes that a document can simultaneously and independently be assigned to multiple labels or classes.

In order to assign class labels to the given memes, we opted for RoBERTa (A Robustly Optimized BERT Pretraining Approach) as a neural network architecture for token and sequence classification. Starting off with a pretrained model for language representation we fine-tuned this model on the given classification task with the provided annotated data in supervised training steps. To incorporate image features in the multi-modal setting, we rely on the pre-trained VGG-16 model architecture.

1 Introduction

Social networks provide opportunities to conduct disinformation campaigns for organizations as well as individual actors. The proliferation of disinformation online, has given rise to a lot of research on automatic fake news detection. SemEval-2021 Task 6 considers disinformation as a communication phenomenon. By detecting the use of various persuasion techniques in communication, it takes into account not only the content but also how a subject matter is communicated.

The goal of the shared task is to build models for identifying such techniques in the textual content of a meme only (two subtasks) and in a multimodal setting in which both the textual and the visual content are to be analysed together (one subtask).

The shared task defines the following subtasks:

Subtask 1

Given only the "textual content" of a meme, identify which of the 20 techniques are used in it. This is a multilabel classification problem.

Subtask 2

Given only the "textual content" of a meme, identify which of the 20 techniques are used in it together with the span(s) of text covered by each technique. This is a multilabel sequence tagging task. The task is the combination of the two subtasks of the SemEval 2020 task 11 on "detecting propaganda techniques in news articles". Note that subtask 1 is a simplified version of subtask 2 in which the spans covered by each technique is not supposed to be provided.

Subtask 3

Given a meme, identify which of the 22 techniques are used both in the textual and visual content of the meme (multimodal task). This is a multilabel classification problem.

In this work, we covered our approach on both technique classification (TC) tasks (Subtask 1 and Subtask 3) detecting the type of communication technique used in a given message. To build models, the first subtask assumes purely textual content as inputs, whereas the third is designed in multimodal setting in which both the textual and the visual content are to be analysed together. Below, we describe the systems built for these two subtasks. At the core of our systems is RoBERTa (Liu et al., 2019), a pre-trained model based on the Transformer architecture (Vaswani et al., 2017).

Although we did not manage to participate in

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 1037–1044
Bangkok, Thailand (online), August 5–6, 2021. ©2021 Association for Computational Linguistics

the second subtask, we will describe our solution below for the sake of completeness.

Finally, we will address some limitation of the general settings of this shared task.

2 Related Work

The goal of the shared task is to investigate automatic techniques for identifying various rhetorical and psychological techniques in online disinformation campaigns. A comprehensive survey on fake news has been presented by Zhou and Zafarani (2018). 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. (2018), 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 and then open up our focus to dissemination mechanisms. Traditional information discovery methods are based on content: documents, terms, and the relationships between them (Leskovec and Lang, 2008). 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 IE

Recent work on text classification uses neural networks, particularly Deep Learning (DL). Badjatiya et al. (2017) demonstrated that these architectures, including variants of recurrent neural networks (RNN) (Gao and Huang, 2017; Pavlopoulos et al., 2017; Pitsilis et al., 2018), convolutional neural networks (CNN) Zhang et al. (2018), 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. (2017) 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 (Devlin et al., 2018) and XLNet (Yang et al., 2019) 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 (Mikolov et al., 2013), GloVe (Pennington et al., 2014), fastText (Joulin et al., 2017), or ELMo (Peters et al., 2018) to capture the local patterns of features to benefit text classification, has been described by Castelle (2019). 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. (2017). 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 (Liu et al., 2019), 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.

2.5 Image Feature Extraction using Pre-trained Models

Convolutional neural network (CNN) visual features have demonstrated their powerful ability as a universal representation for various recognition tasks. In this study we rely on the extraction of visual features on state of the art convolutional neural network architectures. From the most popular architectures such as VGG (Simonyan and Zisserman, 2015), ResNet (He et al., 2016), AlexNet (Krizhevsky et al., 2017), GoogLeNet (Szegedy et al., 2015) we initially generated the image features using a pre-trained VGG-16 model architecture.

2.6 Multimodal Deep Learning

Multimodal deep learning involves multiple modalities used together to predict some output. The different modalities present in a particular piece of content are extracted and fused early in the classification process. In this study, we concatenated the features extracted from images and text sequences using a Convolutional Neural Network (CNN) and RoBERTa encodings (Liu et al., 2019), respectively. These features were used to try and predict persuasive techniques.

3 Dataset

The dataset to this task is provided by Dimitrov et al. (2021). Furthermore, there is a related shared task "SemEval 2020 task 11 on Detecting propaganda techniques in news articles" (Martino et al., 2020) since it serves as the basis for the second sub-task. In particular, the second subtask is the combination of the two subtasks of the previous task. Finally, there is a recent survey on computational propaganda detection by da San Martino et al. (2019).

4 Our approach

In this section, we provide a general overview of our approaches to the three subtasks. Subtasks 1 and 3 are both given as multilabel classification problems, whereas subtaks 2 is given as a multilabel sequence tagging task.

4.1 Experimental setup: Subtask 1

Model Architecture This subtask is a multiclass multi-label problem, as one or more labels have to be assigned to each sample. Our model for this subtask is based on RoBERTa.

Input Embeddings The input embedding layer converts the inputs (memes text) 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 (Wu et al., 2016). 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:

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

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

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

Target Encoding We encode the target labels using a multi-label binarizer as an analog of one-hot aka one-of-K scheme to multiple labels.

4.2 Experimental setup: Subtask 2

This subtask is given as a multilabel sequence tagging problem.

Tagging format We transformed the initial span markup into a IOB tagging format (Inside, Outside, Begin). As we have 20 possible entity classes, each token can be assigned one of the 41 tags given by an O-tag, and the I-tag and B-tag of the various techniques, respectively.

Model Architecture We fine-tuned a RoBERTa model to predict the above IOB tags for each token in the input sentence. One problem with the above setup is that each token is classified independently of the surrounding tokens: while these surrounding tokens are taken into account in the contextualized embeddings that RoBERTa produces, there is no modeling of the dependency between the predicted labels: for example, logically an I-tag should not follow O. Since RoBERTa does not model the dependencies between the predicted token, we further added a linear-chain Conditional Random Field (CRF) model (Lafferty et al., 2001) as an additional layer, in order to model the dependency between the predicted labels of individual tokens. Since the sequence of an O-tag following an I-tag does not appear in the training set, it assigns by observation a very low probability to the transition from an O-tag to an I-tag. We trained the resulting RoBERTa-CRF model as shown in Figure 1. The CRF receives the logits for each input token, and makes a prediction for the entire input sequence, taking into account the dependencies between the labels, similarly to (Lample et al., 2016). Note that RoBERTa works with byte pair encoding (BPE) units, while for the CRF it makes more sense to work with complete words. Thus, only head tokens were used as input to the CRF, and skipping any word continuation tokens.

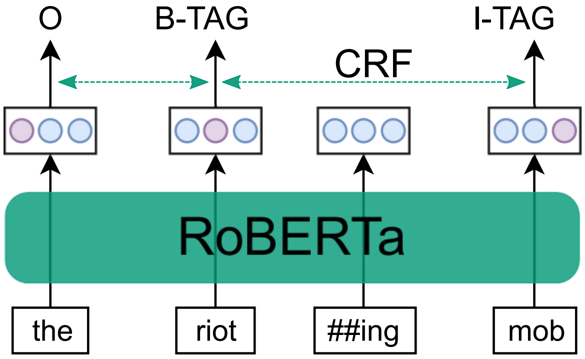

Figure 1: RoBERTa-CRF model with IOB-encoded target. The CRF model ignores non-starting word pieces such as the depicted ##ing token.

4.3 Experimental setup: Subtask 3

Model Architecture We build our cross-modality model with self-attention and cross-attention layers following the recent progress in designing natural language processing models (e.g., transformers (Vaswani et al., 2017)). Our model takes two inputs as part of a meme: an image and its related text. Each image is represented as a feature vector, and each sentence is represented as a sequence of words. As depicted in Figure 2, via design and combination of the self-attention and cross-attention layers, our model is able to generate language representations, image representations, and cross-modality representations from the inputs. Next, we describe the components of this model in detail.

Input Embeddings The input embedding layers convert the inputs (i.e., an image and a short text) into two sequences of features: word-level sentence embeddings and image 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 (Wu et al., 2016). 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:

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

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

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

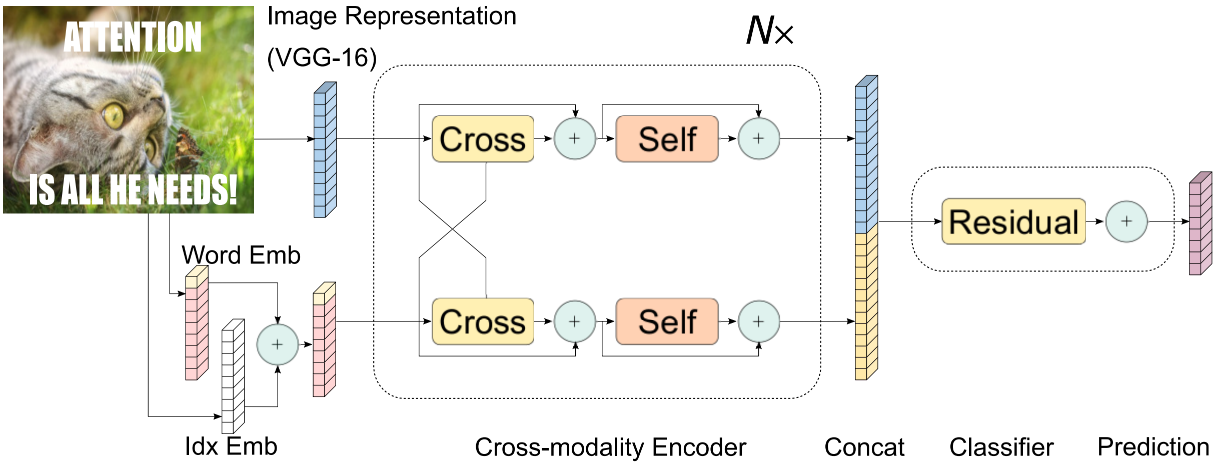

Figure 2: Multimodal model configuration for learning imaage-and-text cross-modality representations. 'Self' and 'Cross' are abbreviations for self-attention sublayers and cross-attention sublayers, respectively.

Visual features In this study we rely on the extraction of visual features on state of the art convolutional neural network architectures by generating the image features using a pre-trained VGG-16 model.

Encoders We build our encoders, i.e., the language encoder, and the cross-modality encoder, on the basis of two kinds of attention layers: self-attention layers and crossattention layers. We first review the definition and notations of attention layers and then discuss how they form our encoders.

Attention Layers Attention layers (Bahdanau et al., 2015; Xu et al., 2015) 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 (Vaswani et al., 2017).

Single-Modality Encoders After the embedding layers, we apply a temporal convolutional layer to each single modality. The result of this projection is a uniform feature space with defined dimensions as the input to the cross-modality encoder.

Cross-Modality Encoder Each cross-modality layer in the cross-modality encoder consists of two self-attention sub-layers, one bi-directional cross-attention sublayer, and a feed-forward sub-layer. We stack (i.e., using the output of k-th layer as the input of (k+1)-th layer) $N_\times$ these cross-modality layers in our encoder implementation. Inside the k-th layer, the bi-directional cross-attention sub-layer ('Cross') is first applied, which contains two unidirectional cross-attention sub-layers: one from text to image and one from image to text. The query and context vectors are the outputs of the (k-1)-th layer (i.e., text features $\{t_i^{k-1}\}$ and image features $\{i_j^{k-1}\}$):

$$\hat{t}_i^k = CrossAtt_{T\to I}(t_i^{k-1}, \{i_1^{k-1}, ..., i_m^{k-1}\})$$

$$\hat{i}_j^k = CrossAtt_{I\to T}(i_j^{k-1}, \{t_1^{k-1}, ..., t_n^{k-1}\}$$

The cross-attention sub-layer is used to exchange the information and align the entities between the two modalities in order to learn joint cross-modality representations. For further building internal connections, the self-attention sub-layers ('Self') are then applied to the output of the crossattention sub-layer:

$$\tilde{t}_i^k = SelfAtt_{T\to T}(\hat{t}_i^k, \{\hat{t}_1^k, ..., \hat{t}_m^k\})$$

$$\tilde{i}_j^k = SelfAtt_{I\to I}(\hat{i}_j^k, \{\hat{i}_1^k, ..., \hat{i}_n^k\}$$

We add a residual connection and layer normalization (annotated by the '+' sign in Fig. 1) after each sublayer as in Vaswani et al. (2017).

Classification At the core of the classifier we use a residual block as introduced by He et al. (2016). The input to the residual block is given by concatenation of the output of the cross-modality encoder. The input and output size of the residual block corresponds to the sum of the output size of the cross-modality encoder. Lastly, in order to obtain the desired one-hot encoding as the output of the classsifier, a linear transformation is applied.

Target Encoding We encode the target labels using a multi-label binarizer as an analog of one-hot aka one-of-K scheme to multiple labels.

4.4 Results and Discussion

We participated in both techniques classification tasks (subtask 1 and 3). The official evaluation ranked our system 9th and 13th out of 16 and 15 teams, respectively. In this study, 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. Even without extensive pre-processing of the training data, we already achieve competitive results and can serve as strong baseline models which, when fine-tuned, significantly outperform training models from scratch. When improving on these baseline models, data scarcity appears to be an immense challenge. This is especially evident in the ratio of the given training samples to the number of possible target classes. We expected better results with the multimodal solution. The causes of the problem will be investigated in more detail in the future.

5 Conclusion and Future work

We described our approach for the SemEval-2021 Task 6 on Detection of Persuasion Techniques in Text and Images. We employed RoBERTa-based neural architectures, additional CRF layers, and a cross-modality framework for learning the connections between image and text in a multi-modal transformer architecture.

In future work, we plan to investigate more recent neural architectures for language representation such as T5 (Raffel et al., 2019) and GPT-3 (Brown et al., 2020). In case of the multimodal setting, it might also be useful to evaluate alternative model architectures such as ResNet (He et al., 2016) to improve image representation.

Furthermore, we expect great opportunities for transfer learning from the areas such as argumentation mining (Stede, 2020) and offensive language detection (Zampieri et al., 2019). 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 (Ruder and Plank, 2018) as well as weak supervision (Ratner et al., 2020).

References

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. In *26th International World Wide Web Conference 2017, WWW 2017 Companion*, pages 759–760. International World Wide Web Conferences Steering Committee.

Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. 2015. 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.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Michael Castelle. 2019. The Linguistic Ideologies of Deep Abusive Language Classification. pages 160–170.

Giovanni da San Martino, Seunghak Yu, Alberto Barrón-Cedeño, Rostislav Petrov, and Preslav Nakov. 2019. Fine-grained analysis of propaganda in news articles.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

Dimiter Dimitrov, Bishr Bin Ali, Shaden Shaar, Firoj Alam, Fabrizio Silvestri, Hamed Firooz, Preslav Nakov, and Giovanni Da San Martino. 2021. Task 6 at SemEval-2021: Detection of Persuasion Techniques in Texts and Images. In *Proceedings of the 15th International Workshop on Semantic Evaluation*, SemEval˜'21, Bangkok, Thailand.

Lei Gao and Ruihong Huang. 2017. Detecting online hate speech using context aware models. In *International Conference Recent Advances in Natural Language Processing, RANLP*, volume 2017-Septe, pages 260–266. Association for Computational Linguistics (ACL).

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2016-Decem, pages 770–778. IEEE Computer Society.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. 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, pages 427–431.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2017. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90.

John Lafferty, Andrew Mccallum, and Fernando Pereira. 2001. Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data Abstract. 2001(June):282–289.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT 2016 - Proceedings of the Conference*, pages 260–270. Association for Computational Linguistics (ACL).

J Leskovec and KJ Lang. 2008. Statistical properties of community structure in large social and information networks. *Proceedings of the 17th international conference on World Wide Web. ACM*, pages 695–704.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach.

Giovanni Da San Martino, Alberto Barrón-Cedeño, Henning Wachsmuth, Rostislav Petrov, and Preslav Nakov. 2020. SemEval-2020 Task 11: Detection of propaganda techniques in news articles.

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. 2013. Exploiting Similarities among Languages for Machine Translation.

John Pavlopoulos, Prodromos Malakasiotis, and Ion Androutsopoulos. 2017. Deeper attention to abusive user content moderation. In *EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, pages 1125–1135, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. GloVe: Global vectors for word representation. In *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. pages 2227–2237. Association for Computational Linguistics (ACL).

Georgios K. Pitsilis, Heri Ramampiaro, and Helge Langseth. 2018. Effective hate-speech detection in Twitter data using recurrent neural networks. *Applied Intelligence*, 48(12):4730–4742.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *arXiv*, 21:1–67.

Alexander Ratner, Stephen H. Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. 2020. Snorkel: rapid training data creation with weak supervision. In *VLDB Journal*, volume 29, pages 709–730. Springer.

Sebastian Ruder and Barbara Plank. 2018. 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)*, 1:1044–1054.

Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR.

Manfred Stede. 2020. Automatic argumentation mining and the role of stance and sentiment. *Journal of Argumentation in Context*, 9(1):19–41.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 07-12-June, pages 1–9.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 2017-Decem, pages 5999–6009.

Claes H. de Vreese, Frank Esser, Toril Aalberg, Carsten Reinemann, and James Stanyer. 2018. Populism as an Expression of Political Communication Content and Style: A New Perspective. *International Journal of Press/Politics*, 23(4):423–438.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation.

Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *32nd International Conference on Machine Learning, ICML 2015*, volume 3, pages 2048–2057. International Machine Learning Society (IMLS).

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. Technical report.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. 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, pages 1415–1420, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ziqi Zhang, David Robinson, and Jonathan Tepper. 2018. 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, pages 745–760. Springer Verlag.

Xinyi Zhou and Reza Zafarani. 2018. Fake News: A Survey of Research, Detection Methods, and Opportunities. *ACM Comput. Surv*, 1(1).