Sentence-Level Propaganda Detection in News Articles with Transfer Learning and BERT-BiLSTM-Capsule Model

George-Alexandru Vlad, Mircea-Adrian Tanase, Cristian Onose and **Dumitru-Clementin Cercel**
Computer Science Department, Faculty of Automatic Control and Computers
University Politehnica of Bucharest, Romania
{georgealexandruvlad, mirceatanase1994, onose.cristian, clementin.cercel}@gmail.com

Abstract

In recent years, the need for communication increased in online social media. Propaganda is a mechanism which was used throughout history to influence public opinion and it is gaining a new dimension with the rising interest of online social media. This paper presents our submission to NLP4IF-2019 Shared Task SLC: Sentence-level Propaganda Detection in news articles. The challenge of this task is to build a robust binary classifier able to provide corresponding propaganda labels, propaganda or non-propaganda. Our model relies on a unified neural network, which consists of several deep leaning modules, namely BERT, BiLSTM and Capsule, to solve the sentence-level propaganda classification problem. In addition, we take a pre-training approach on a somewhat similar task (i.e., emotion classification) improving results against the cold-start model. Among the 26 participant teams in the NLP4IF-2019 Task SLC, our solution ranked 12th with an F_1-score 0.5868 on the official test data. Our proposed solution indicates promising results since our system significantly exceeds the baseline approach of the task organizers by 0.1521 and is slightly lower than the winning system by 0.0454.

1 Introduction

The most widely agreed upon definition of propaganda was formulated by the Institute for Propaganda Analysis (1937) and describes the phenomenon as actions exercised by individuals or groups with the purpose of influencing the opinions of target individuals. This phenomenon was present in the news industry throughout history. However, the concern over the presence of propaganda techniques in news articles has grown exponentially since the rise of social media platforms, especially after the massive impact it had in recent political events, such as the US 2016 elections or Brexit (Barrón-Cedeño et al., 2019a).

Automating the detection of propaganda in news articles is considered very difficult since propaganda uses various techniques (Da San Martino et al., 2019) that, in order to achieve the pursued effect, should not be discovered by the target individuals. The Shared Task of Fine-grained Propaganda Detection of NLP4IF workshop (Da San Martino et al., 2019) consists in two tasks: FLC (Fragment-level Classification) and SLC (Sentence-level Classification). We participated in the SLC task which implied sentence-level classification for the presence of propaganda.

Recently, a series of approaches have been studied in respect to language modeling to obtain a deeper understanding of language (Devlin et al., 2018; Peters et al., 2018; Radford et al., 2018). Thus, the latest solutions of obtaining language representations keep track of the word context to model the relationship between words. Here, we choose to use Bidirectional Encoder Representations from Transformers (BERT) embeddings as it showed performance improvements on a series of Natural Language Processing (NLP) tasks, such as the SQuAD v1.1 and SWAG datasets (Devlin et al., 2018). Moreover, we aim to study the newly developed architecture of Capsule Networks (Sabour et al., 2017) which were first applied in the field of computer vision (Xi et al., 2017). Between the word embeddings generated by BERT and the Capsule layer, we integrate a Bidirectional Long Short-Term Memory (BiLSTM) (Schuster and Paliwal, 1997) layer to capture the semantic features of the human language by cumulating prior and future knowledge for every input token.

In our paper, we analyze the impact of different architectures based on the main components previously mentioned in order to validate our final unified model, namely BERT-BiLSTM-Capsule. Moreover, we study the relationship be-

Proceedings of the 2nd Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda, pages 148–154
Hong Kong, China, November 4, 2019

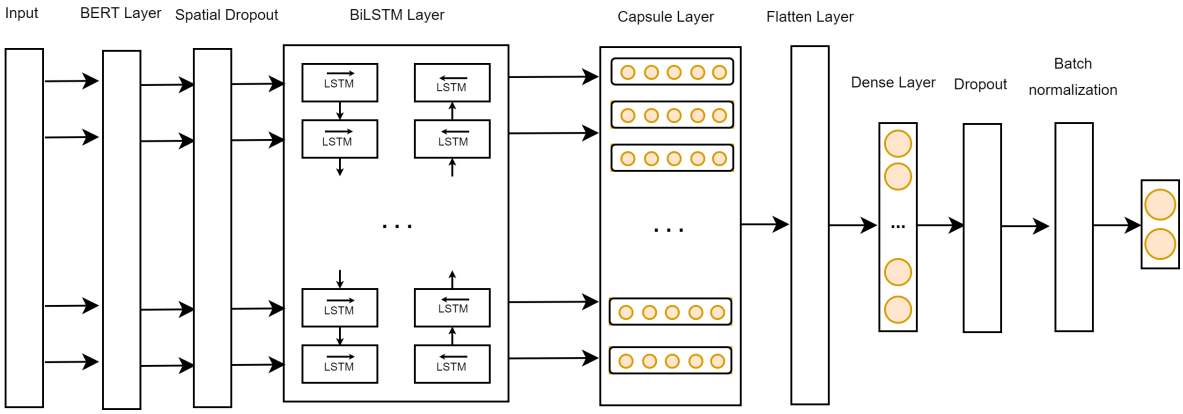

Figure 1: BERT-BiLSTM-Capsule model architecture.

tween emotions and the presence of propaganda by pretraining the BERT-BiLSTM-Capsule model on an emotion labeled dataset. We therefore use the learned weights as a starting point for training on the propaganda dataset.

The remainder of the paper is structured as follows: in Section 2, we present an analysis of the literature on the topic of propaganda detection, in Section 3 we offer an in-depth description of our system and in the Section 4 we present the experimental setup and the results obtained in the SLC challenge. Finally, we present the conclusions of this work.

2 Related work

At first, the task of automated propaganda detection was approached as a subtask of the broader problem imposed by fake news detection (Traylor et al., 2019). The automated detection of fake news has gained a massive interest in the research community with the rise of machine learning algorithms that enabled the development of powerful NLP techniques. One of the consecrated fake news dataset was created by (Shu et al., 2018) and the authors also presented an overview of the data mining based techniques employed for this task and their results in (Shu et al., 2017).

In recent research, propaganda detection in news articles was approached as a standalone problem (Da San Martino et al., 2019). The first part of the task consists of creating a correctly labeled dataset. Some of the earlier works (Rashkin et al., 2017) attempted labeling news outlets as trustworthy or not and considering all the articles published by an outlet as having the same label. This method was proved inaccurate, as propagandistic news outlets also publish objective articles in order to gain readers' trust. Barrón-Cedeño et al. (2019a); Barrón-Cedeno et al. (2019b) designed Proppy, a real time propaganda detection system designed to monitor news sources, which computes a propaganda index using a maximum entropy classifier based on a variety of features including n-grams, readability scores and lexicon features. Baisa et al. (2017) introduced a corpus of more than 5,000 Czech newspaper articles annotated for propaganda use, with a large set of features extracted for each one.

Most recently, Da San Martino et al. (2019) proposed a different annotation level, where not only the articles are labeled individually in a binary way (propagandistic or non-propagandistic), but also each fragment of a sentence containing one of eighteen identified propaganda techniques is labeled accordingly. The authors also test several state-of-the-art NLP models such as BERT, obtaining promising results in both binary classification and identifying individual propagandistic fragments.

3 Methodology

3.1 BERT-BiLSTM-Capsule Model

In this subsection, a detailed description of the BERT-BiLSTM-Capsule model is presented. A high-level overview of our model is illustrated in Figure 1.

BERT Layer. In order to obtain word encodings from the raw sentence, we use BERT (Devlin et al., 2018). The BERT model is based on the Transformer architecture (Vaswani et al., 2017) which follows an encoder-decoder design commonly used in neural machine translation.

BERT model stacks multiple Transformer layers to obtain a deeper representation of the input and applies a masking procedure on the token sequence named Masking Language Model. In contrast to the masking procedure used in Transformer architecture, which performs a sequential masking of the words by replacing the words to be predicted with a mask token, BERT masks a percentage of words at random, determining the bidirectional characteristic of the model. This procedure enables BERT to attain information surrounding the masked word in both directions and also enables a human-like approach in determining a missing word within a context.

BERT model comes in two sizes: BERT-Base (L=12, H=768, A=12, # of parameters=110M) and BERT-Large (L=24, H=1024, A=16, # of parameters=340M), where L means layer, H means hidden, and A means attention heads. In our implementation, we used the BERT-Large model with pretrained weights[1].

The BERT model could take as input a sentence or a pair of sentences depending on the task in hand. The input sentence is represented by a vector of indices, a mapping of the raw sentence words into integer values accordingly to a dictionary based on the BERT vocabulary.

In our model, we use a single sentence as input to the BERT model. We extract the last encoder layer as the output of the BERT layer, which will be further used as input layer to the BiLSTM layer. To decrease the chance of overfitting, we add a spatial dropout layer (Srivastava et al., 2014) after the BERT layer.

BiLSTM Layer. The BiLSTM layer (Schuster and Paliwal, 1997) takes as input the output of the BERT model which returns a sequence $V \in \mathbb{R}^{t \times d}$ where t is the number of encoded tokens returned by the last BERT layer, matching the number of tokens provided as input to the BERT model, and d the dimension of the token encoding. The BiLSTM layer consists of two LSTM layers which processes the input from both left to right and vice versa. Each LSTM produces a sequence of hidden states h which encodes the current token and the prior knowledge of the processed tokens. The resulting hidden states of each LSTM cell for both directions $\overrightarrow{h_l}$ and $\overleftarrow{h_l}$ are concatenated together for each time step $i = 1 \ldots t$ with t the number of input tokens. The resulted sequence of t hidden states $h_i = \overrightarrow{h_l} | \overleftarrow{h_l}$ is then passed to the next layer.

Capsule Layer. The Capsule Networks (Sabour et al., 2017; Hinton et al., 2018) proposed a new approach in selecting the most salient features extracted by precedent layers, acting as a replacement for the more common Max Pooling technique. The Max Pooling step implies dropping the knowledge gathered by activation of several neurons depending on the window of Max Pooling and passing forward only the boldest features, which might imply ignoring relevant information. Capsule Networks not only overcome this disadvantage but also propose a more intuitive approach in determining the presence of concepts by grouping information from a hierarchical standpoint, base concepts validating the existence of more complex ones.

We used a two-layer Capsule Network to determine the relationship between concepts, a primary capsule layer to capture the instantiated parameters from previous layers and a convolutional Capsule layer to determine the routing between capsules.

The primary capsule layer applies a convolutional operation over the sequence of hidden states $x \in \mathbb{R}^{t \times d}$ from the previous layer where t is the number of embedded tokens and d the dimension of the embedding. In our case, depending on the chosen architecture, the embedding sequence x comes from the recurrent layer or directly from the output token embeddings of the BERT layer. Connection between capsules is determined by a procedure called routing-by-agreement.

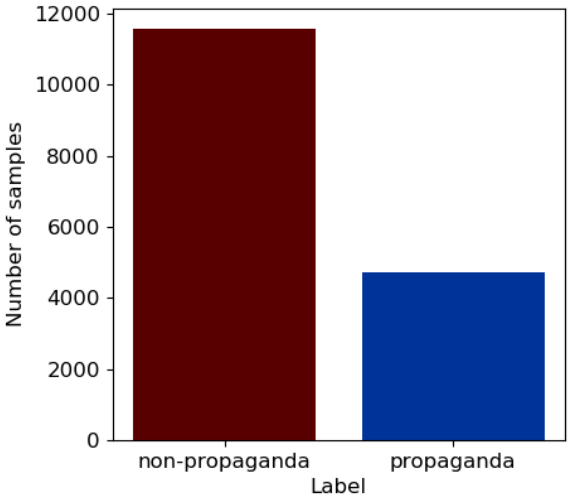

Figure 2: Class label distribution for SLC propaganda dataset.

[1]https://github.com/google-research/bert

Dense Layer. The results of the Capsule layer are flattened, and a dense layer is stacked on top of them. In order to make the model more robust to overfitting, we add both a batch normalization layer as well as a dropout layer. The output is then passed to a final dense layer consisting of 2 neurons, one for each class, propaganda or non-propaganda. Softmax activation is used over the output layer to generate a probability distribution over the two classes.

3.2 BERT-Emotion System

In our proposed model, we freeze the BERT transformer layers to preserve the already pretrained weights and only fine-tune the BiLSTM, Capsule and Dense layers. This procedure is applied with success in the field of computer vision, transferring and freezing the weights of top-performing models becoming a common practice in order to conserve the feature extractive layers. This drastically reduces the computational power required for training step with a slightly lower performance than fine-tuning all the BERT layers (Beltagy et al., 2019).

This procedure is applied in training of the BERT-BiLSTM-Capsule model on both datasets, i.e., propaganda and emotion. After training the BERT-BiLSTM-Capsule model on the emotion dataset, we use the learned weights to initialize the model to be trained on the propaganda task. We will further refer to it as BERT-Emotion.

4 Evaluation

4.1 Data

The SLC task provides a dataset containing 350 articles, annotated for the presence of propaganda with two labels: propaganda and non-propaganda, for the training step.

We use an additional dataset annotated for emotion and perform a transfer learning step to initialize the weights of the BERT-BiLSTM-Capsule model trained on the propaganda task. The emotion dataset is obtained by unifying a series of datasets annotated for different classes of emotions. A solution[2] of unifying multiple emotion datasets was proposed by Bostan and Klinger (2018). To this dataset, we add the Daily dialogue dataset (Li et al., 2017) that contains 11,318 transcribed dialogues manually annotated for 7 emotions: neutral, anger, disgust, fear, happiness, sadness and surprise. The third dataset we use to augment the emotion dataset is the Semeval-2019 Task 3 dataset (Chatterjee et al., 2019) containing 15k records for three emotion classes (i.e., happy, sad and angry) and 15k records not belonging to any of the previously mentioned emotion classes. From the resulted dataset, only the entries annotated for the 4 basic emotions are selected, namely neutral, joy, anger and sadness.

[2]https://github.com/sarnthil/unify-emotion-datasets

4.2 Preprocessing

The provided dataset contains empty strings which are labeled as non-propaganda. We extract all the non-empty entries from the SLC dataset. The obtained dataset contains 16,297 sentences. The distribution between propaganda and non-propaganda classes in the resulted dataset is illustrated in Figure 2.

Because the emotion dataset suffers from severe class imbalance, we decided to restrict the number of samples of the neutral class, which has the highest presence, to 30k entries. The class distribution of the obtained emotion dataset is shown in Figure 3.

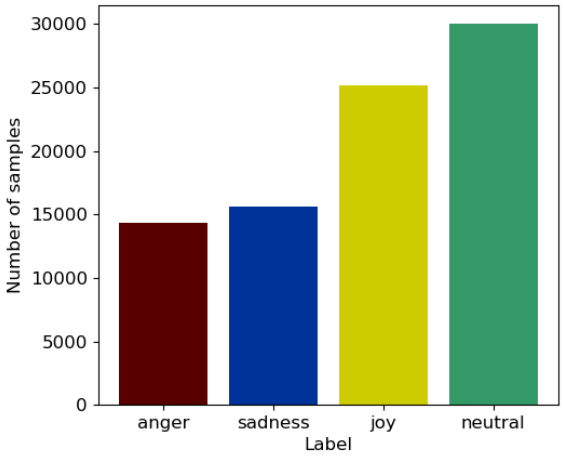

Figure 3: Class label distribution for emotion unified dataset.

We further split both the propaganda and emotion datasets in train and validation sets with the following ratio 0.9/0.1. Because the class distribution is not balanced, we preserve the initial distribution in both splits to keep the validation results relevant for the model's performance.

For the preprocessing step, we use the BERT tokenizer to transpose each word into corresponding index based on the BERT vocabulary. This vocabulary contains entries for 30,522 tokens. The resulting sentence encoding is delimited by the

[CLS] token at the start of the sentence and by the [SEP] token at the end.

4.3 Experimental Settings

During the experiments, we use the Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.001 which decreases with a factor of 0.1 after 7 epochs of no improvement until a lower bound of 10-e5. The BiLSTM hidden size is set to 200 and the second last dense layer has a size of 100. The Dropout technique is used with a probability of 0.12 on the features extracted by capsules and a spatial dropout of 0.1 on the embeddings returned by the BERT layer. For the Capsule layer, we also use 10 capsules of dimension 10. The hyperparameters for our model were chosen empirically.

After performing the stratified splitting of the propaganda dataset into training and validation sets, the class distribution remains unchanged in both splits, the propaganda and non-propaganda classes maintaining the original ratio 0.72/0.28. We use a weighted cross-entropy loss in order to increase the amount of attention paid to samples from an under-represented class. The weights associated for every class are computed as follow:

$$\frac{1}{w_n} = \frac{a_n}{\sum_{t=1}^{n} a_t} \quad (1)$$

where a_n represents the number of samples of class n in training set. A similar approach is used in training the BERT-BiLSTM-Capsule model on the emotion dataset.

4.4 Results

Effect of Various Model Parts. First, we study the impact of each component of our BERT-BiLSTM-Capsule model by removing one layer at a time and retraining the resulted model on the propaganda dataset. The ablation study on the components of our model enables to objectively choose the top performing architecture. Because the F_1 score is the official metric by which the challenge evaluation is made, we assess the performance of each architecture with respect to it.

The results are shown in Table 1. The BERT-BiLSTM-Capsule model outperforms the other architectures by over 2.1% and achieves highest precision. Based on these results, we choose to use the BERT-BiLSTM-Capsule model for the transfer learning step.

Model	Rec.	Prec.	F_1	Acc.
BERT-BiLSTM	**0.8557**	0.8292	0.5909	**0.7723**
BERT-Capsule	0.8506	0.8284	0.5870	0.7687
BERT-BiLSTM-Capsule	0.8126	**0.8508**	**0.6164**	0.7656

Table 1: Ablation study of our BERT-BiLSTM-Capsule model on the validation set. For each metric, the best result is highlighted in bold.

Comparison with our Baselines. We test our proposed solution against two baseline models to validate our BERT-Emotion system. The baseline methods are described below, and we report their results in Table 2.

Model	Rec.	Prec.	F_1	Acc.
XG-Boost	0.6737	0.4862	0.5648	0.6993
BERT-Simple	0.7797	0.8543	0.6086	0.7490
BERT-Emotion	**0.8082**	**0.8618**	**0.6338**	**0.7717**

Table 2: Comparative results against our base models on the validation set. The best results are shown in boldface.

First baseline model is represented by the simple BERT model in which we unfreeze the last dense layer and add another dense layer of size 2 with softmax activation to map the obtained features to the output propaganda classes. We will refer to it as BERT-Simple.

As a second baseline model, we used an XG-Boost classifier (Chen and Guestrin, 2016) based on the following features:

- First, the lemma of the words was extracted and the TF-IDF scores (Jones, 2004) were computed for the n-grams obtained, with n = 1, 2, 3.
- Secondly, parts of speech were tags were extracted using the NLTK Python package[3] and the TF-IDF scores were computed for the tag n-grams obtained, with n = 1, 2, 3.
- Thirdly, TF-IDF scores were computed for character n-grams, with n=1, 2, 3.

[3]https://www.nltk.org/

System	Rec.	Prec.	F_1
Ituorp (1^{nd})	0.6648	0.6028	0.6323
BERT-Emotion (12^{th} place)	**0.5747**	**0.5995**	**0.5868**
SLC baseline	0.4941	0.3880	0.4347

Table 3: Comparative analysis against the official baseline result as well as the best performer of the SLC task. Our result is shown in boldface.

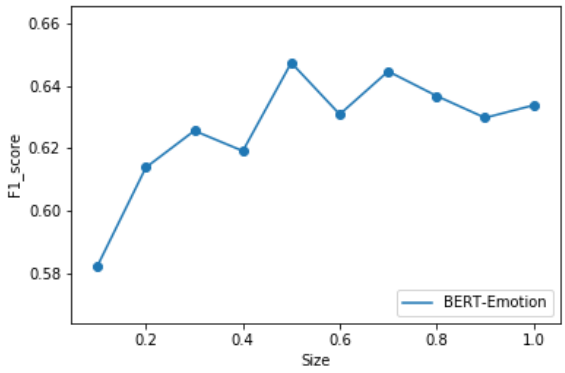

Figure 4: Learning curve on the training set.

- Sentiment analysis features were obtained using the VADER tool (Hutto et al., 2015).
- Other lexical features were added, such as number of characters, words, syllables and the Flesch-Kincaid readability score (Kincaid et al., 1975).

Leaderboard. We submitted for evaluation our BERT-Emotion system and obtained competitive results on the SLC task. In Table 3, we present our results on the test set in comparison to the SLC task baseline and the highest-ranking team.

Effect of Size of the Training Data. In order to determine the correlation between the number of samples provided in training set and the F_1 score obtained on the validation set, we choose to plot the learning curve. Thus, we study the data insufficiency issue for our model and examine the possible need of a larger training dataset in achieving a better performance. We split the training set in 10 blocks, every block employing a percent of the original training dataset between 10% and 100% with a step of 10%. In splitting the original training set, we maintain the original class distribution to keep the relevance of the results. Figure 4 plots the obtained results.

Our model's performance on the validation set is dependent on the dataset size until the 5th block containing 50% of the original dataset, after which the learning curve reaches a plateau. This implies not only that the amount of data provided for training is sufficient but also that our model has a good understanding of the data, being capable to abstract the knowledge needed for the propaganda classification task and successfully generalize the learned information on the new data.

5 Conclusions

In this paper, we described our system (BERT-Emotion) submitted to the Shared Task of Fine-grained Propaganda Detection of the NLP4IF 2019 workshop. We proposed a transfer learning approach by pretraining our BERT-BiLSTM-Capsule model on a distinct task (i.e., emotion classification), procedure which has proven to successfully increase our system's inference ability on the target task (i.e., sentence-level propaganda classification). We based our model on the BERT-Large version for getting word embeddings instead of classical pretrained embeddings and explore the promising design of Capsule Networks.

Our final system obtained substantial improvements against competition official baseline and our baseline systems as well. In the future, we intend to adopt additional contextualized embeddings such as ELMo (Peters et al., 2018) and FLAIR (Akbik et al., 2018) to test the BERT-Emotion performance.

6 Acknowledgments

The work was supported by the Operational Programme Human Capital of the Ministry of European Funds through the Financial Agreement 51675/09.07.2019, SMIS code 125125.

References

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649.

Vít Baisa, Ondrej Herman, and Ales Horák. 2017. Manipulative Propaganda Techniques. In *RASLAN*, pages 111–118.

Alberto Barrón-Cedeño, Giovanni Da San Martino, Israa Jaradat, and Preslav Nakov. 2019a. Proppy: A system to unmask propaganda in online news. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9847–9848.

Alberto Barrón-Cedeno, Israa Jaradat, Giovanni Da San Martino, and Preslav Nakov. 2019b. Proppy: Organizing the news based on their propagandistic content. *Information Processing & Management*.

Iz Beltagy, Arman Cohan, and Kyle Lo. 2019. Scibert: Pretrained contextualized embeddings for scientific text. *arXiv preprint arXiv:1903.10676*.

Laura Ana Maria Bostan and Roman Klinger. 2018. An analysis of annotated corpora for emotion classification in text. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2104–2119.

Ankush Chatterjee, Kedhar Nath Narahari, Meghana Joshi, and Puneet Agrawal. 2019. SemEval-2019 Task 3: EmoContext Contextual Emotion Detection in Text. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 39–48.

Tianqi Chen and Carlos Guestrin. 2016. XGBoost: reliable large-scale tree boosting system. arXiv. *2016a. ISSN*, pages 0146–4833.

Giovanni Da San Martino, Seunghak Yu, Alberto Barrón-Cedeño, Rostislav Petrov, and Preslav Nakov. 2019. Fine-Grained Analysis of Propaganda in News Articles. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, EMNLP-IJCNLP 2019, Hong Kong, China.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Geoffrey E Hinton, Sara Sabour, and Nicholas Frosst. 2018. Matrix capsules with EM routing.

C Hutto, Dennis Folds, and Darren Appling. 2015. Computationally detecting and quantifying the degree of bias in sentence-level text of news stories. In *Proceedings of Second International Conference on Human and Social Analytics*.

Institute for Propaganda Analysis. 1937. How to detect propaganda. *Propaganda Analysis*, 1(2):5–8.

Karen Spärck Jones. 2004. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*.

JP Kincaid, RP Fishburn, R Rogers, and B Chissom. 1975. Derivation of new readability formulas for Navy enlisted personnel (Research Branch Report 8-75). *Memphis, TN: Naval Air Station, Millington, Tennessee*, page 40.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. Dailydialog: A manually labelled multi-turn dialogue dataset. *arXiv preprint arXiv:1710.03957*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning. Technical report, Technical report, OpenAI.

Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2931–2937.

Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. 2017. Dynamic routing between capsules. In *Advances in neural information processing systems*, pages 3856–3866.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Kai Shu, Deepak Mahudeswaran, Suhang Wang, Dongwon Lee, and Huan Liu. 2018. Fakenewsnet: A data repository with news content, social context and dynamic information for studying fake news on social media. *arXiv preprint arXiv:1809.01286*.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958.

Terry Traylor, Jeremy Straub, Nicholas Snell, et al. 2019. Classifying fake news articles using natural language processing to identify in-article attribution as a supervised learning estimator. In *2019 IEEE 13th International Conference on Semantic Computing (ICSC)*, pages 445–449. IEEE.

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*, pages 5998–6008.

Edgar Xi, Selina Bing, and Yang Jin. 2017. Capsule network performance on complex data. *arXiv preprint arXiv:1712.03480*.