# Ensembles of Recurrent Networks for Classifying the Relationship of Fake News Titles

Ting Su
t.su.2@research.gla.ac.uk
University of Glasgow, UK

Craig Macdonald
University of Glasgow, UK
craig.macdonald@glasgow.ac.uk

Iadh Ounis
Iadh.Ounis@glasgow.ac.uk
University of Glasgow, UK

## ABSTRACT

Nowadays, everyone can create and publish news and information anonymously online. However, the credibility of such news and information are not guaranteed. To differentiate fake news from genuine news, one can compare a recent news with earlier posted ones. Identified suspicious news can be debunked to stop the fake news from spreading further. In this paper, we investigate the advantages of recurrent neural networks-based language representations (e.g., BERT, BiLSTM) in order to build ensemble classifiers that can accurately predict if one news title is related to, and, additionally disagrees with an earlier news title. Our experiments, on a dataset of 321k news titles created for the WSDM 2019 challenge, show that the BERT-based models significantly outperform BiLSTM, which in-turn significantly outperforms a simpler embedding-based representation. Furthermore, even the state-of-the-art BERT approach can be enhanced when combined with a simple BM25 feature.

**ACM Reference Format:**
Ting Su, Craig Macdonald, and Iadh Ounis. 2019. Ensembles of Recurrent Networks for Classifying the Relationship of Fake News Titles. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '19), July 21–25, 2019, Paris, France.* ACM, New York, NY, USA, 4 pages. https://doi.org/https://doi.org/10.1145/3331184.3331305

## 1 INTRODUCTION

With the increasing popularity of social media, creating and sharing information is no longer a privilege reserved for government and news agencies. Anyone can freely create information and share any content online. However, online information may not be fact-checked before it is published and spread, which makes it easier for misleading, malicious or false information to contaminate online discussions. Such misleading or untrusted information is colloquially referred to as "fake news". As time passes by, shared and spread information may be debunked or shown to be false [17]. Therefore, using other news stories to determine the truthfulness of a news article is one of the possible ways to identify fake news.

However, comparing news with each other consumes both time and effort, as hundreds of stories are created every minute (e.g., an average of 1.7k blogs were posted on WordPress.com every minute in January 2019[1]). To help automate this approach and deploy it on large-scale data, we propose to build an automatic classifier that can identify the relationship between two news article titles. Such a task was the object of the recent WSDM Cup 2019 challenge[2]. In the challenge, participants were asked to predict the relationship between two Chinese news article titles. In particular, gitven two news titles ($t_A$ and $t_B$), a system should classify if the second title ($t_B$) *agrees* ($t_B$ talks about the same news as $t_A$), *disagrees* ($t_B$ refutes the news in $t_A$), or is *unrelated*, to the first title ($t_A$). Among these, the *disagreement* relationship is the most important, since such article pairs either contain fake news, or need to be fact-checked.

Effectively addressing the WSDM Cup 2019 task is the focus of this paper. To do so, we build upon recent advances in recurrent neural networks (RNN) for text processing. Moreover, we note that this task is similar to natural language inference (NLI) in natural language processing (NLP), where sentences are predicted to be logically related or not. Recent NLI methods focus heavily on neural networks (NN) (e.g., [9]). Similarly, we also draw parallels to the task of learning semantic matching between queries and documents (e.g. [1]). Interestingly, deep semantic approaches for this task have been trained in a weakly supervised fashion using the classical BM25 document weighting model [1], or have exhibited a benefit when combined with BM25 [18]. Therefore, we propose to use the ensemble RNN-based (i.e., BiLSTMs [4] and the state-of-the-art BERT model [2]) classification methods together with BM25 representation, in order to effectively classify the relationship between two news titles written in Chinese.

The contributions of this paper are two-fold: by comparing simple-embedding representations with BiLSTM and BERT, we draw best practices in using RNN representations in classifying the relationship of Chinese news titles; Secondly, we examine how the traditional BM25 retrieval score can improve the performance of state-of-the-art deep NN models.

## 2 RELATED WORK

As an emerging challenge, identifying fake news is of both public and scientific interest. Indeed, scholars are developing methods to automatically identify fake news. These methods typically use machine learning techniques to classify fake news among genuine news. There are two aspects that scholars usually focus on when training classifiers: feature engineering and model engineering.

For feature engineering, extracting features from a linguistic aspect is a commonly used approach. Aside from using a typical bag of words (BoW) representation and TF.IDF scores [10], many scholars also used other handcrafted features (e.g., Number of @, #, exclamation marks, first-person pronouns [6]). Deep syntactical analysis approaches [8] are also useful in detecting fake news [3]. However, these methods are often handcrafted and rule-based, where generalisation is not guaranteed. As word embeddings (e.g., Word2Vec approach [11]) have gained popularity for representing language, classic word embeddings methods are widely used to obtain the semantic information of terms in a linguistic manner.

[1] wordpress.com/activity/posting/

[2] kaggle.com/c/fake-news-pair-classification-challenge/

*SIGIR '19, July 21–25, 2019, Paris, France*

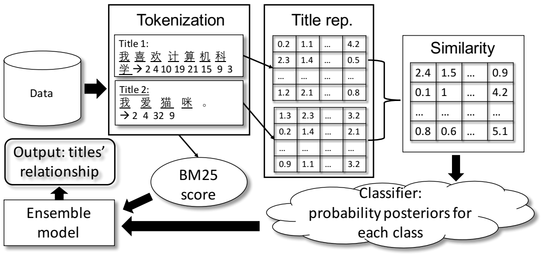

Tokenization
Title rep.
Similarity
Data
Title 1:
我 喜 欢 计 算 机 科 学 → 2 4 10 19 21 15 9 3
Title 2:
我 爱 猫 咪 。
→ 2 4 32 9
Output: titles' relationship
BM25 score
Ensemble model
Classifier: probability posteriors for each class

**Figure 1: The structure and components of our model.**

The aforementioned methods are either task-specific (e.g. TF.IDF and POS are usually trained on the training set, instead of using a large pre-trained model), or too general (e.g. Word2Vec, usually trained using Wikipedia, may have differing term distributions compared to the target dataset). Therefore, in this paper, we propose to use both dynamic word embeddings (i.e., RNN related) methods and a BoW method, to tackle the limitation of using only one method.

In terms of model engineering, traditional classifiers (e.g., support vector machine (SVM), Naive Bayes (NB)) are widely used. Recent work leveraged the learning power of neural networks. Ma et al. [10] showed that using a RNN with a Long Short-Term Memory (LSTM) architecture, with only TF.IDF features, can outperform SVM models with handcrafted features in classifying if a news event, represented by a set of tweets, is fake or not.

However, previous researchers have mainly focused on analysing single articles or tweets [6], or analysing a series of tweets in a generic manner [10], without paying much attention to use the debunking articles to aid the labelling of previous news. In contrast, the recent WSDM Cup 2019 fake news challenge addresses matching new articles with previously stored identified fake news. In this challenge, the winning group, *saigonapps*[3], used a BERT-based language representation and handcrafted features to represent each title pair; these features were then ensembled to produce the final result, obtaining 88.2% accuracy. Similarly, we also build upon BERT, but additionally contrast its use with other language representation methods, from BM25 to BiLSTMs.

## 3 MODEL COMPONENTS

As highlighted in Section 1, this work is concerned with identifying the relationship between two news titles $t_A$ and $t_B$. In particular, given two titles, the classifier $f()$ should make a prediction as to whether the titles are *unrelated*, *agree* or *disagree*, shown as follows:

$$f(t_A, t_B) \rightarrow \{unrelated, agree, disagree\} \qquad (1)$$

The decision of *unrelated* vs. (*agree* || *disagree*) is that of identifying relevance. We build upon standard text similarity approaches, as well as customised classifiers, to determine if the ensembled models can make the *agree* vs. *disagree* decision more effectively.

Figure 1 illustrates an outline of our approach, in four steps: character-level tokenisation; representing terms and titles; similarity calculations; classifiers and ensembles. Table 1 shows combinations of various used representations of terms and titles, similarity calculations and classifiers, leading to different models. The table also lists the abbreviation names given to the resulting models. For example, the ESC model uses an embedding layer to represent terms, the BiLSTM layer to represent titles, subtraction similarity, and the CNN classifier. Note that the ensemble models are not listed in Table 1, but are denoted as model abbr. + BM25.

**Table 1: Models and their components used in this work.**

| Abbr. | Term/Title rep. | Similarity | Classifier |
|---|---|---|---|
| BL | Emb-Concat | Dot | MLP (15 layers) |
| LR(BM25) | BoW | BM25 | LR |
| ESM | Emb-BiLSTM | Subtraction | MLP (2 layers) |
| ESC | Emb-BiLSTM | Subtraction | CNN |
| EDM | Emb-BiLSTM | Dot | MLP (2 layers) |
| EDC | Emb-BiLSTM | Dot | CNN |
| BERT | BERT | - | SoftMax layer |
| BSM | BERT | Subtraction | MLP (2 layers) |
| BSC | BERT | Subtraction | CNN |
| BDM | BERT | Dot | MLP (2 layers) |
| BDC | BERT | Dot | CNN |

### 3.1 Terms and Titles Representation

Firstly, we use a text segmentation method to transform the titles into series of tokens, as the WSDM 2019 Cup dataset is in Mandarin Chinese (for which a number of segmenters are available). However, in our initial experiments, we found that character-level segmentation was the most effective, and hence we only report its results.

Once we have a sequence of tokens for each title, we represent each token as a vector, which are then combined together to represent each title. Below, we describe the three methods (denoted 1(a), 1(b) and 2) used to combine token representations into a title representation, where the resulting representation can be a vector (using the BERT method) or a matrix (using the embedding layer and neural network methods):

**1. Embedding layer within a neural network (denoted as Emb):** Using an embedding layer represents discrete terms using continuous vectors, which carry more meaningful information than a one-hot encoding method. Indeed, using an embedding layer within a neural network allows the vector representation of terms to be adjusted to the current task, since the network learns to minimise the loss on that task. The resulting embedding vectors represent terms semantically, so that similar terms are closer to each other.

To represent titles, we add a layer after the embedding layer to combine tokens into titles. We use two methods, namely:

1(a). **Concatenating vectors vertically (denoted as Concat):** We concatenate the term vectors in a given title into a matrix, following the order of the occurrence of terms within the title, where each row is a vector representation of a term.

1(b). **Bidirectional LSTM (BiLSTM):** LSTM is designed to capture the sequential features within an instance, and is commonly used in NLP (e.g., [7]). However, LSTM is omnidirectional, and hence does not consider future tokens [16]. To address this limitation, we use BiLSTM with the embedding layer to generate a matrix for each title, thereby capturing both the semantic meaning and the two-way sequential information of the title terms.

**2. Bidirectional Encoder Representations from Transformers (BERT):** BERT is a bidirectional, and attention-based neural network language representation model. In particular, it randomly masks 15% of the words in the input data, and uses an attention-based LSTM transformer to predict the masked words, which provides vector representations for the masked words. The title representations are obtained by pooling the final layer of the BERT network. We use a pre-trained Chinese model[4] and perform fine-tuning on our dataset, rather than training a model using our dataset.

[3] github.com/lampts/wsdm19cup

[4] github.com/google-research/bert

**Table 2: Statistics of the WSDM 2019 Cup dataset.**

| Dataset | # *Unrelated* | # *Agree* | # *Disagree* | # Total |
|---|---|---|---|---|
| Training | 198416 | 84626 | 7511 | 276025 |
| Validation | 8831 | 5406 | 291 | 14528 |
| Testing | 20897 | 8347 | 755 | 29999 |

### 3.2 Title Similarity

Using the representations obtained from Section 3.1, we measure the similarity between two titles using the following three methods:

(1) **Cosine similarity (denoted as Dot).** Cosine similarity measures the angle between two representations, which represent the orientation of the subjects between two titles [15].
(2) **Vector Subtraction (denoted as Subtraction).** As mentioned above, an embedding model is able to capture the semantic information of terms. Therefore, we use a subtraction function between two titles' representations to measure the semantic distance. Note that although subtraction is not a commutative operation, it is appropriate in this task, as the relationship between $t_B$ and $t_A$ is an ordered relationship.
(3) **BM25.** BM25 [14] is a weighting model that is traditionally used to score documents, given a query. Due to the need to identify if titles discuss the same news event, we use BM25 to measure title similarity.

### 3.3 Final Classifiers & Ensemble

The outputs of the similarity are combined using two classifiers. We compare using a multilayer perceptron (denoted as MLP) with using a convolutional neural network (denoted as CNN) to classify the relationship of two titles and to output the predicted class.

Finally, integrating BM25 directly into a NN is not practical, as BM25 measures the relevance of titles at the title level, and the inputs to the final stage of NN measures the relationship of two titles at a character level. Therefore, we use a logistic regression classifier (LR), which combines the BM25 score and the NN classifier class posteriors of each title pair as input, to predict the relationship between $t_A$ and $t_B$. In particular, we choose LR because it performs the best after testing other conventional classifiers (e.g. SVM, NB).

## 4 EXPERIMENTAL SETUP

Our experiments address two research questions, namely:

- **RQ1**: Which model is most effective in learning to accurately predict the relationship between pairs of news article titles?
- **RQ2**: Does combining the BM25 relevance score with a NN model improves accuracy in predicting the relationship between pairs of news article titles?

**Dataset.** We use the WSDM 2019 Cup dataset[5], which is a collection of human-written Chinese news title pairs, that are labelled either *unrelated*, *agree*, or *disagree*. All the titles are pooled from Chinese news providers or content creators. The size of the dataset, along with the number of pairs in each class are listed in Table 2.

**Tokenisation.** We use the WordPiece segmenter (implemented in BERT) to segment each Chinese title into characters. Note that any English words in the titles remain as words. We remove the stopwords before tokenisation. We trim each title to be exactly 45 words/characters, in order to enhance the BiLSTM performance (only 0.0003% of titles in the training set exceed this length).

**Embedding.** We apply a Siamese style [12] embedding layer in all the neural network approaches (except models that use BERT for representation), where each term is embedded into 128 dimensions, and $t_A$ and $t_B$ share the same embedding layer.

**BiLSTM.** We use the Keras[6] implementation of bidirectional LSTMs, using 64 units for each layer, and a dropout rate of 0.01.

**BERT.** We begin with the BERT-Base Chinese model (12-layer, 768-hidden, 12-heads, 110M parameters). Following standard practice [2], we fine-tune the BERT model on the training dataset. All other parameters remain at their recommended settings. Moreover, when integrating the output of BERT into other classification methods, we remove its final SoftMax layer, except for the model denoted "BERT" in Table 1, where we keep the final SoftMax layer.

**Classifiers.** We tune all of the hyper-parameters on the training set. We use the Adam optimiser with a learning rate of 0.001, and ReLU [13] as the activation function, for both the MLP and CNN classifiers. For MLP, we use 2 layers with 64 and 16 units in each respective layer. We use 32 filters, 3 kernels, and stride 1 for CNN. We implement our models using the MarchZoo deep text matching toolbox [5][7]. We use the Sage solver, the L2 penalty, and a C regularisation score of 10 for LR.

**Baseline.** We train a neural network with an embedding layer, concatenate the words' vectors in a title to be a 2D matrix, and use Cosine similarity as the similarity function. Finally, we train a 15 layers MLP classifier as the baseline. We use this baseline to show that representing titles using a BiLSTM model is more effective than simply concatenating terms together.

**Evaluation Metrics.** We report accuracy, balanced accuracy (BAC), precision, recall, and F1 scores as evaluation metrics.

## 5 RESULTS AND ANALYSIS

To address the research questions presented in Section 4, we present the results of our news title relationship classification experiments. Table 3 presents the classification results of each model tested on the test set. Note that our data is unbalanced, where the *agree* and *disagree* classes are more important, but are smaller in size. Therefore, we also report the BAC and the accuracy metrics of these two classes, where the performances of models are measured separately.

Firstly, addressing **RQ1**, we evaluate the classification performances of the Emb-BiLSTM models[8]. Table 3 shows that all of the Emb-BiLSTM and BERT-related models outperform the baseline model. They also outperform LR(BM25) in terms of both BAC and accuracy on the *agree* & *disagree* classes. However, the BERT-based models marginally outperform the Emb-BiLSTM models. We postulate that this is because learning an embedding on a small dataset results in less/biased information for representing terms.

Meanwhile, we observe that EDC (accuracy of 0.789) outperforms EDM (accuracy of 0.752) while ESC (accuracy of 0.703) outperforms ESM (accuracy of 0.696). Moreover, EDM and EDC outperform ESM and ESC, respectively. Therefore, for the Emb-BiLSTM-related models, we conclude that the Cosine similarity performs better than subtraction, and that using a CNN classifier performs better than MLP. On the contrary, the performances of the two similarity

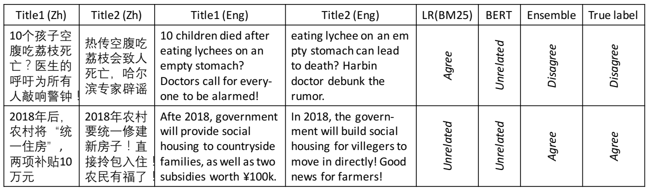
[5] kaggle.com/c/fake-news-pair-classification-challenge/data. Note that the training data is used and separated into training, validation, and testing set, because of the lack of ground truth for the testing data.

[6] keras.io [7] github.com/NTMC-Community/MatchZoo/ [8] We test our models on a portion of the training data, since the ground truth test set is not yet available. Hence, we do not compare our results to the winning group in the WSDM Cup 2019.

**Table 3: Classification scores . Bold denotes the best result in the table. †† denotes that an ensemble model significantly outperforms both the corresponding RNN/BERT model as well as LR(BM25) (McNemar's test, p < 0.01).**

| Model | Acc | BAC | P | R | F1 | *Agree*<br>Acc | *Disagree*<br>Acc |
|---|---|---|---|---|---|---|---|
| BL | 0.632 | 0.692 | 0.71 | 0.62 | 0.67 | 0.712 | 602 |
| LR(BM25) | 0.758 | 0.544 | 0.80 | 0.76 | 0.77 | 0.794 | 0.006 |
| ESM | 0.696 | 0.704 | 0.77 | 0.70 | 0.72 | 0.763 | 0.665 |
| + BM25†† | 0.765 | 0.736 | 0.80 | 0.76 | 0.78 | 0.783 | 0.678 |
| ESC | 0.703 | 0.715 | 0.78 | 0.70 | 0.72 | 0.791 | 0.656 |
| + BM25†† | 0.778 | 0.743 | 0.81 | 0.78 | 0.79 | 0.800 | 0.686 |
| EDM | 0.752 | 0.779 | 0.82 | 0.75 | 0.77 | 0.847 | 0.729 |
| + BM25†† | 0.762 | 0.782 | 0.83 | 0.76 | 0.78 | 0.879 | 0.804 |
| EDC | 0.789 | 0.758 | 0.83 | 0.79 | 0.80 | 0.809 | 0.687 |
| + BM25†† | 0.779 | 0.760 | 0.82 | 0.78 | 0.79 | 0.828 | 0.756 |
| BERT | **0.885** | 0.735 | **0.88** | **0.89** | **0.88** | 0.822 | 0.458 |
| + BM25†† | 0.875 | 0.815 | **0.88** | 0.87 | **0.88** | 0.858 | 0.697 |
| BSM | 0.863 | 0.825 | **0.88** | 0.86 | 0.87 | 0.877 | 0.736 |
| + BM25†† | 0.851 | **0.847** | **0.88** | 0.85 | 0.86 | **0.892** | **0.826** |
| BSC | 0.859 | 0.816 | 0.87 | 0.86 | 0.86 | 0.873 | 0.717 |
| + BM25†† | 0.856 | 0.823 | 0.87 | 0.86 | 0.86 | 0.877 | 0.804 |
| BDM | 0.763 | 0.657 | 0.81 | 0.76 | 0.78 | 0.657 | 0.497 |
| + BM25†† | 0.770 | 0.665 | 0.82 | 0.77 | 0.79 | 0.796 | 0.499 |
| BDC | 0.851 | 0.815 | 0.87 | 0.85 | 0.86 | 0.877 | 0.715 |
| + BM25†† | 0.845 | 0.826 | **0.88** | 0.85 | 0.86 | 0.886 | 0.767 |

**Table 4: Two examples, where LR(BM25) model and BERT model both give the wrong prediction, but the ensemble model gives the correct prediction.**



| Title1 (Zh) | Title2 (Zh) | Title1 (Eng) | Title2 (Eng) | LR(BM25) | BERT | Ensemble | True label |
|---|---|---|---|---|---|---|---|
| 10个孩子空腹吃荔枝死亡？医生的呼吁为所有人敲响警钟！ | 热传空腹吃荔枝会致人死亡，哈尔滨专家辟谣 | 10 children died after eating lychees on an empty stomach? Doctors call for every-one to be alarmed! | eating lychee on an em pty stomach can lead to death? Harbin doctor debunk the rumor. | *Agree* | *Unrelated* | *Disagree* | *Disagree* |
| 2018年后，农村将“统一住房”，两项补贴10万元 | 2018年农村要统一修建新房子！直接拎包入住！农民有福了！ | Afte 2018, government will provide social housing to countryside families, as well as two subsidies worth ¥100k. | In 2018, the govern-ment will build social housing for villegers to move in directly! Good news for farmers! | *Unrelated* | *Unrelated* | *Agree* | *Agree* |

methods used with the BERT-related models are the opposite of that using Emb-BiLSTM (i.e., BSM/BSC outperform BDM/BDC). One possibility is that, as a complex model to learn the representation of words, BERT obtains detailed information about them, and the nature of the title representation models results in over-fitting/under-fitting. We do not observe the same performances with the MLP and CNN methods, as BSM outperforms BSC, but BDC marginally outperforms BDM.

Of all the models presented in Table 3, the BERT model achieves the best accuracy and F1 score. However, the BSM model achieves the best BAC, as well as the best *agree* and *disagree* classes accuracy. Therefore, in response to **RQ1**, we conclude that the BSF model most accurately predicts the *agree* and *disagree* classes in this Chinese news title relationship classification task.

Turning our attention to **RQ2**, Table 3 shows that the BSM model ensemble with BM25 achieves the best BAC score (0.847), and the best *agree* and *disagree* class accuracies (0.892 and 0.826, respectively). Indeed, the BACs of all models increase when BM25 is ensembled, but the accuracy scores do not increase consistently. The observation of increasing BAC is particularly interesting, as the BM25 model alone does not achieve a high BAC, but assists other models to perform better for the smaller classes, namely, *agree* and *disagree*. Indeed, Table 4 presents examples where both the BERT model and the LR(BM25) model predict incorrectly, while the ensemble model predicts correctly. Therefore, regarding **RQ2**, we conclude that including the BM25 score does improve the performance of using the RNN model-related classifiers, especially improving the performance of the *agree* and *disagree* classes.

## 6 CONCLUSIONS

In this paper, we addressed a core task needed for fake news detection as defined by the recent WSDM 2019 Cup fake news challenge. In particular, we investigated various neural network-based representations for detecting news title relationships. Our thorough experiments showed that using BERT for text representation, using the subtraction similarity method and MLP as the classifier predicted the *agree* and *disagree* classes the most accurately. Moreover, when the RNN models are combined in an ensemble with the BM25 similarity, it results in significant improvements to the effectiveness of all models. This finding suggests that a BM25 matching score can aid RNN approaches, and ensemble methods can perform better than each component used alone, arguably because BM25 can better identify obvious similarities that are difficult for the RNN to learn.

## REFERENCES

[1] Mostafa Dehghani, Hamed Zamani, Aliaksei Severyn, Jaap Kamps, and W Bruce Croft. 2017. Neural ranking models with weak supervision. In *Proc. of SIGIR*.

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

[3] Song Feng, Ritwik Banerjee, and Yejin Choi. 2012. Syntactic stylometry for deception detection. In *Proc. of ACL*.

[4] Reza Ghaeini, Sadid A Hasan, Vivek Datla, Joey Liu, Kathy Lee, Ashequl Qadir, Yuan Ling, Aaditya Prakash, Xiaoli Z Fern, and Oladimeji Farri. 2018. Dr-bilstm: Dependent reading bidirectional lstm for natural language inference. *arXiv:1802.05577* (2018).

[5] Jiafeng Guo, Yixing Fan, Xiang Ji, and Xueqi Cheng. 2019. MatchZoo: A Learning, Practicing, and Developing System for Neural Text Matching. In *Proc. of SIGIR*.

[6] Sardar Hamidian and Mona Diab. 2016. Rumor identification and belief investigation on Twitter. In *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*. 3–8.

[7] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *arXiv:1602.02410* (2016).

[8] Tadao Kasami. 1966. An efficient recognition and syntax-analysis algorithm for context-free languages. *Coordinated Science Laboratory Report no. R-257* (1966).

[9] Seonhoon Kim, Jin-Hyuk Hong, Inho Kang, and Nojun Kwak. 2018. Semantic sentence matching with densely-connected recurrent and co-attentive information. *arXiv:1805.11360* (2018).

[10] Jing Ma, Wei Gao, Prasenjit Mitra, Sejeong Kwon, Bernard J Jansen, Kam-Fai Wong, and Meeyoung Cha. 2016. Detecting Rumors from Microblogs with Recurrent Neural Networks.. In *IJCAI*. 3818–3824.

[11] Tomas Mikolov, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013).

[12] Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. In *Proc. of AAAI*.

[13] Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proc. of ICML*. 807–814.

[14] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at TREC-3.

[15] Amit Singhal et al. 2001. Modern information retrieval: A brief overview. *Bulletin of the Technical Committee on Data Engineering* 24, 4 (2001).

[16] Ming Tan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2015. LSTM-based deep learning models for non-factoid answer selection. *arXiv:1511.04108* (2015).

[17] Nguyen Vo and Kyumin Lee. 2018. The rise of guardians: Fact-checking url recommendation to combat fake news. In *Proc. of SIGIR*.

[18] Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-End Open-Domain Question Answering with BERTserini. *arXiv:1902.01718* (2019).