

Multimodal Fake News Detection

Santiago Alonso-Bartolome (santiagoalonsobartolome@gmail.com), Isabel Segura-Bedmar (isegura@inf.uc3m.es)

Computer Science Department, Universidad Carlos III de Madrid, Avenida de la Universidad, 30, Leganés, 28911, Madrid, Spain

Corresponding Author:

Isabel Segura-Bedmar

Computer Science Department, Universidad Carlos III de Madrid, Avenida de la Universidad, 30, Leganés, 28911, Madrid, Spain

Tel: +34 916245961

Email: isegura@inf.uc3m.es

Multimodal Fake News Detection

Santiago Alonso-Bartolome, Isabel Segura-Bedmar*

*Computer Science Department, Universidad Carlos III de Madrid, Avenida de la
Universidad, 30, Leganés, 28911, Madrid, Spain*

Abstract

Over the last years, there has been an unprecedented proliferation of fake news. As a consequence, we are more susceptible to the pernicious impact that misinformation and disinformation spreading can have in different segments of our society. Thus, the development of tools for automatic detection of fake news plays an important role in the prevention of its negative effects. Most attempts to detect and classify false content focus only on using textual information. Multimodal approaches are less frequent and they typically classify news either as true or fake. In this work, we perform a fine-grained classification of fake news on the Fakeddit dataset, using both unimodal and multimodal approaches. Our experiments show that the multimodal approach based on a Convolutional Neural Network (CNN) architecture combining text and image data achieves the best results, with an accuracy of 87%. Some fake news categories such as Manipulated content, Satire or False connection strongly benefit from the use of images. Using images also improves the results of the other categories, but with less impact. Regarding the unimodal approaches using only text, Bidirectional Encoder Representations from Transformers (BERT) is the best model with an accuracy of 78%. Therefore, exploiting both text and image data significantly improves the performance of fake news detection.

Keywords: Fake News Detection, Natural Language Processing, Deep Learning, Multimodal, BERT, CNN

*Corresponding author.

Email addresses: santiagoalonsobartolome@gmail.com (Santiago Alonso-Bartolome), isegura@inf.uc3m.es (Isabel Segura-Bedmar)

1. Introduction

Digital medial has provided a lot of benefits to our modern society such as facilitating social interactions, boosting productivity, improving sharing information, among many others. However, it has also also led to the proliferation of fake news (Finneman & Thomas, 2018), that is, news articles containing false information that has been deliberately created (Hunt & Gentzkow, 2017). The effects of this kind of misinformation and disinformation spreading can be seen in different segments of our society. The *Pizzagate* incident (Hauck, 2017) as well as the mob lynchings that occurred in India (Mishra, 2019) are some of the most tragic examples of the consequences of fake news dissemination. Changes in health behaviour intentions (Greene & Murphy, 2021), an increase in vaccine hesitancy (Islam et al., 2021), and significant economic losses (Brown, 2019) are also some of the negative effects that the spread of fake news may have.

Every day, a huge quantity of digital information is produced, making impossible the detection of fake news by manual fact checking. Due to this, it becomes essential to count with techniques that help us to automate the identification of fake news so that more immediate actions can be taken. During the last years, several studies have already been carried out to perform an automatic detection of fake news (Thota et al., 2018; Choudhary et al., 2021; Singh et al., 2021; Giachanou et al., 2020; Singhal et al., 2019; Wang et al., 2018). Most works have focused only on using textual information (unimodal approaches). Much less effort has been devoted to explore multimodal approaches (Singh et al., 2021; Giachanou et al., 2020; Kumari & Ekbal, 2021), which exploit both texts and images to detect the fake news, obtaining better results than the unimodal approaches. However, these studies typically address the problem of fake news detection as a binary classification task (that is, consisting on classifying news as either true or fake).

Therefore, the main goal of this paper is to study both unimodal and multimodal approaches to deal with a finer-grained classification of fake news. To do

this, we use the Fakeddit dataset (Nakamura et al., 2020), made up of posts from Reddit. The posts were classified into the following six different classes: true, misleading content, manipulated content, false connection, imposter content and satire. We explore several deep learning architectures for text classification such as Convolutional neural network (CNN) (Goodfellow et al., 2016), Bidirectional long short-term memory (BiLSTM) (Hochreiter & Schmidhuber, 1997) and Bidirectional encoder representations from transformers (BERT) (Devlin et al., 2018). As multimodal approach, we propose a CNN architecture that combines both texts and images to classify the fake news.

2. Related work

Since the revival of neural networks in the second decade of the current century, many different applications of deep learning techniques have emerged. Part of Natural Language Processing (NLP) and Computer Vision advances are due to the incorporation of deep neural network approaches (O’Mahony et al., 2019; Deng & Liu, 2018). Fields such as object recognition (Zhao et al., 2019), image captioning (Hossain et al., 2019), sentiment analysis (Tang et al., 2015) or question answering (Sharma & Gupta, 2018), among others, have benefited from the development of deep learning in recent years.

Text classification problems are also one of the tasks for which deep neural networks are being extensively used (Minaee et al., 2021a). Most of these works have been based on unimodal approaches that only exploit texts. More ambitious architectures that combine several modalities of data (such as text and image) have also been tried (Abavisani et al., 2020; Bae et al., 2020; Yu & Jiang, 2019; Viana et al., 2017; Gaspar & Alexandre, 2019). The main intuition behind these multimodal approaches is that many texts are often accompanied by images, and these images may provide useful information to improve the results of the classification task (Baheti, 2020).

We now focus on the recent research on fake news detection, distinguishing between unimodal and multimodal approaches.

2.1. Unimodal approaches for fake news detection

We review the most recent studies for the detection of fake news using only the textual content of the news.

Wani et al. (2021) use the Constraint@AAAI Covid-19 fake news dataset (Patwa et al., 2020), which contains tweets classified as true or fake. Several methods are evaluated: CNN, LSTM, Bi-LSTM + Attention, Hierarchical Attention Network (HAN) (Yang et al., 2016), BERT, and DistilBERT (Sanh et al., 2019), a smaller version of BERT. The best accuracy obtained is 98.41 % by the DistilBERT model when it is pre-trained on a corpus of Covid-19 tweets.

Goldani et al. (2021) use a capsule network model (Sabour et al., 2017) based on CNN and pre-trained word embeddings for fake news classification over the ISOT (Ahmed et al., 2017) and LIAR (Wang, 2017) datasets. The ISOT dataset is made up of fake and true news articles collected from *Reuters* and *Kaggle*, while the LIAR dataset contains short statements classified into the following six classes: pants-fire, false, barely-true, half-true, mostly-true and true. Thus, the authors perform both binary and multi-class fake news classification. The best accuracy obtained with the proposed model is 99.8 % for the ISOT dataset (binary classification) and 39.5 % for the LIAR dataset (multi-class classification).

Girgis et al. (2018) perform fake news classification using the above mentioned LIAR dataset. More concretely, they use three different models: vanilla Recurrent Neural Network (Aggarwal, 2018), Gated Recurrent Unit (GRU) (Chung et al., 2014) and LSTM. The GRU model obtains an accuracy of 21.7 %, slightly outperforming the LSTM (21.66 %) and the vanilla RNN (21.5 %) models.

From this review on approaches using only texts, we can conclude that deep learning architectures provide very high accuracy for the binary classification of fake news, however, the performance is much lower when these methods address a fine-grained classification of fake news. Curiously enough, although BERT is reaching state-of-the-art results in many text classification tasks, it has hardly ever been used for the multiclassification of fake news.

2.2. Multimodal approaches for fake news detection

Singh et al. (2021) study the improvement in performance on the binary classification of fake news when textual and visual features are combined as opposed to using only text or image. They explore several traditional machine learning methods: logistic regression (LR) (Kleinbaum & Klein, 2010), classification and regression tree (CART) (Hastie et al., 2009a), linear discriminant analysis (LDA) (Murphy, 2012), quadratic discriminant analysis (QDA) (Murphy, 2012), k-nearest neighbors (KNN) (Murphy, 2012), naïve Bayes (NB) (Barber, 2012), support vector machine (SVM) (Hastie et al., 2009b) and random forest (RF) (Breiman, 2001). The authors use a Kaggle dataset of fake news (Kaggle, b). Random forest is the best model with an accuracy of 95.18 %.

Giachanou et al. (2020) propose a model to perform multimodal classification of news articles as either true or fake. In order to obtain textual representations, the BERT model (Devlin et al., 2018) is applied. For the visual features, the authors use the VGG (Simonyan & Zisserman, 2014) network with 16 layers followed by a LSTM layer and a mean pooling layer. The dataset used by the authors is retrieved from the FakeNewsNet collection (Shu et al., 2020). More concretely the authors use 2,745 fake news and 2,714 real news collected from the *GossipCop* posts of such collection. The proposed model achieves an F1-score of 79.55 %.

Finally, another recent architecture proposed for multimodal fake news classification can be found in the work carried out by Kumari & Ekbal (2021). The authors propose a model that is made up of four modules: i) ABS-BiLSTM (attention based stacked BiLSTM) for extracting the textual features, ii) ABM-CNN-RNN (attention based CNN-RNN) to obtain the visual representations, ii) MFB (multimodal factorized bilinear pooling), where the feature representations obtained from the previous two modules are fused, and iv) MLP (multi-layer perceptron), which takes as input the fused feature representations provided by the MFB module, and then generates the probabilities for each class (true or fake). In order to evaluate the model, two datasets are used: Twitter (Boididou et al., 2015) and Weibo (Jin et al., 2017). The Twitter dataset contains tweets

along with images and contextual information. The Weibo dataset is made up of tweets, images and social context information. The model obtains an accuracy of 88.3% on the Twitter dataset and an accuracy of 83.2% on the Weibo dataset .

Apart from the previous studies, several authors have proposed fake news classification models and have evaluated them using the Fakeddit dataset. Kaliyar et al. (2020) propose the DeepNet model for a binary classification of fake news. This model is made up of one embedding layer, three convolutional layers, one LSTM layer, seven dense layers, ReLU for activation and finally the softmax function for the binary classification. The model is evaluated on the Fakeddit and BuzzFeed (Kaggle, a) datasets. The BuzzFeed dataset contains news articles collected within a week before the U.S. election and they are classified as either true or fake. The models provides an accuracy of 86.4% on the Fakeddit dataset (binary classification) and 95.2% over the BuzzFeed dataset.

Kirchknopf et al. (2021) use four different modalities of data to perform binary classification of fake news over the Fakeddit dataset. More concretely, the authors use the textual content of the news, the associated comments, the images and the remaining metadata belonging to other modalities. The best accuracy obtained is 95.5%. Li et al. (2021) propose the Entity-Oriented Multi-Modal Alignment and Fusion Network (EMAF) for binary fake news detection. The model is made up of an encapsulating module, a cross-modal alignment module, a cross-model fusion module and a classifier. The authors evaluate the model on the Fakeddit, Weibo and Twitter datasets obtaining accuracies of 92.3%, 97.4% and 80.5 %, respectively.

Xie et al. (2021) propose the Stance Extraction and Reasoning Network (SERN) to obtain stance representations of a post and its associated reply. They combine these stance representations with a multimodal representation of the text and image of a post in order to perform binary fake news classification. The authors use the PHEME dataset (Zubiaga et al., 2017) and a reduced version of the Fakeddit dataset created by them. The PHEME dataset contains 5,802 tweets of which 3,830 are real and 1,972 are false. The accuracies obtained are

96.63 % (Fakeddit) and 76.53 % (PHEME).

Finally, Kang et al. (2021) use an heterogeneous graph named News Detection Graph (NDG) that contains domain nodes, news nodes, source nodes and review nodes. Moreover, they propose an Heterogeneous Deep Convolutional Network (HDGCN) in order to obtain the embeddings of the news nodes in NDG. The authors evaluate this model using reduced versions of the Weibo and Fakeddit datasets. For the Weibo dataset, they obtain an F1 score of 96 %, while for the Fakeddit dataset they obtain F1 scores of 88.5% (binary classification), 85.8 % (three classes) and 83.2% (six classes).

As we can see from this review, most multimodal approaches that were evaluated on the Fakeddit dataset only address the binary classification of fake news. Only one of them (Kang et al., 2021) has dealt with the multi-classification of fake news using a reduced version of this dataset.

To the best of our knowledge, this is the first work that addresses a fine-grained classification of fake news using the whole Fakeddit dataset. Furthermore, contrary to the work proposed in (Kang et al., 2021), which exploits a deep convolutional network, we propose a multimodal approach that simply uses a CNN, obtaining very similar performance.

3. Methods

3.1. Unimodal approaches

Three unimodal models only using the texts are proposed: CNN, BiLSTM and BERT.

3.1.1. Preprocessing for deep learning models

We start pre-processing the documents in the corpus by removing stopwords, punctuations, numbers and multiple spaces. Then, we split each text into tokens and we apply lemmatization. After lemmatization, we transform the texts into sequences of integers. This is done firstly by learning the vocabulary of the corpus and building a dictionary where each word is mapped to a different

integer number. Then, this dictionary is used to transform each text into a sequence of integers. Every non-zero entry in such sequence corresponds to a word in the original text. The original order of the words in the text is respected.

As we need to feed the deep learning models with vectors of the same length, we pad and truncate the sequences of integers so that they have the same number of entries. This has the disadvantage that those vectors that are too long will be truncated and some information will be lost. In order to select the length of the padded\truncated vectors, we computed the percentage of texts that are shorter than 10, 15, 20 and 25 tokens. Figure 1 shows the results for the training, validation and test partitions in each case. We can see that 98 % of the texts are smaller than 15 in length. Since the number of texts that will have to be truncated is very small (less than 2 %) very little information is lost. Therefore, we selected 15 as the length of the vectors after padding and truncating.

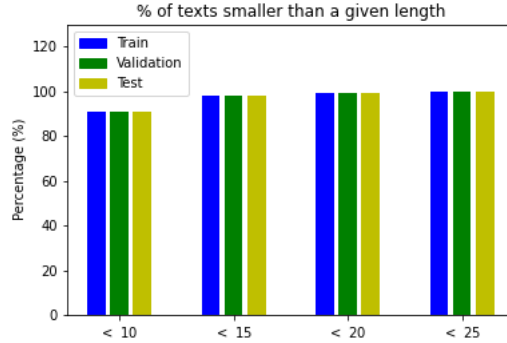


Figure 1: % of texts smaller than a given length

The deep learning architectures use the sequence of word embeddings corresponding to a given text as input. For this reason, an embedding layer will transform each integer value from the input sequence into a vector of word embeddings. In this way, every vectorized document is transformed into matrix of 15 rows and 300 columns (300 being the dimension of the word embeddings). We use both random initialization and pre-trained Glove word embeddings (Pennington et al., 2014). We also compare a dynamic approach (letting the model

further train the word embeddings) and a static approach (not letting the model train the word embeddings).

3.1.2. CNN

We now explain the CNN architecture for the text classification of fake news. As was mentioned above, the first layer is an embedding layer. We initialize the embedding matrix using both random initialization and the pre-trained GloVe word embeddings of dimension 300. We chose this size for the word embeddings over other options (50, 100 or 200) because word embeddings of a larger dimension have been proven to give better results (Patel & Bhattacharyya, 2017).

After the embedding layer, we apply four different filters in a convolutional layer. Each of these filters slides across the (15×300) matrix with the embeddings of the input sequence and generates 50 output channels. The 4 filters have sizes (2×300) , (3×300) , (4×300) and (5×300) , respectively, since these are the typical filter sizes of a CNN for text classification (Voita, 2021). As a consequence the outputs of the previous filters have shapes (14×1) , (13×1) , (12×1) and (11×1) , respectively.

The next step is to pass the outputs obtained from the previous layer through the ReLU activation function. This function is applied element-wise, and therefore, it does not alter the size of the outputs obtained after the previous step. The effect of this function is to set all the negative values to 0 and leave the positive values unchanged.

After going through the ReLU activation, a maxpooling layer is applied that selects the biggest element out of each of the 200 feature maps (50 feature maps per each of the 4 filters). Thus, 200 single numbers are generated.

These 200 numbers are concatenated and the result is passed through 2 dense layers with one ReLU activation in between (Minaee et al., 2021b). The resulting output is a vector of six entries (each entry corresponding to a different class of the Fakeddit dataset) that, after passing through the logsoftmax function, can be used to obtain the predicted class for the corresponding input text.

Early stopping (Brownlee) with the train and validation partitions is used in order to select the appropriate number of epochs. We use the *Adam* optimization algorithm (Kingma & Ba, 2014) for training the model and the *negative log likelihood* as loss function. Figure 2 shows the CNN architecture for text classification.

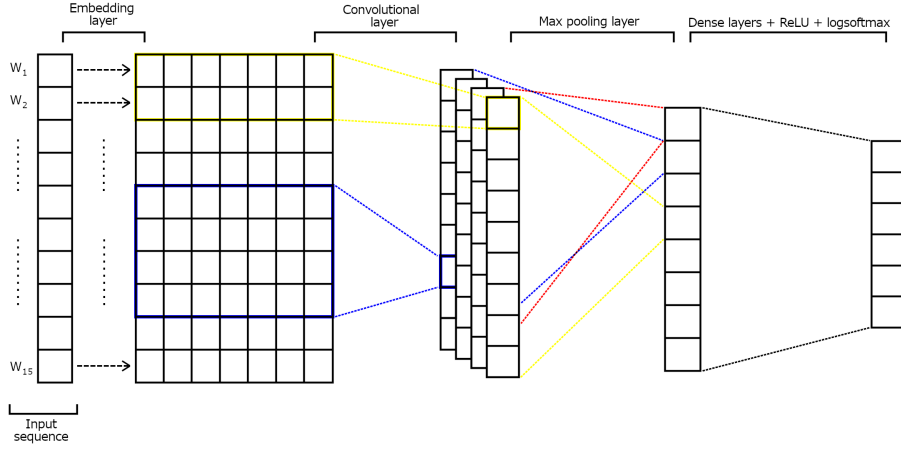


Figure 2: CNN for text classification.

3.1.3. BiLSTM

Actually, this model is a hybrid model that uses a bidirectional LSTM followed by a CNN layer. Firstly, texts are processed as was described above and these input are passed through the same embedding layer that used for the CNN model. Therefore, each input vector of length 15 is transformed into a matrix of shape 15 x 300.

Then, the matrix with the word embeddings goes through a bidirectional LSTM layer with hidden states of length 70. The output of this layer is a matrix of size 15 x 140 that contains two hidden states (corresponding to the two directions of the BiLSTM) for each word embedding. The output of the BiLSTM layer is the input of a convolutional layer, which applies 240 filters of size (3 x 140) therefore it generates 240 output arrays of size (13 x 1). Then,

the ReLU activation is applied followed by a maxpooling layer that selects the largest element within each of the 240 feature maps. Thus, this layer outputs a sequence of 240 numbers.

Similarly to what was done for the CNN model, the output of the maxpooling layer is concatenated and passed through two dense layers with ReLU activation in between. The resulting vector goes through the logsoftmax function and the predicted class is obtained. Figure 3 shows the architecture of the BiLSTM for text classification.

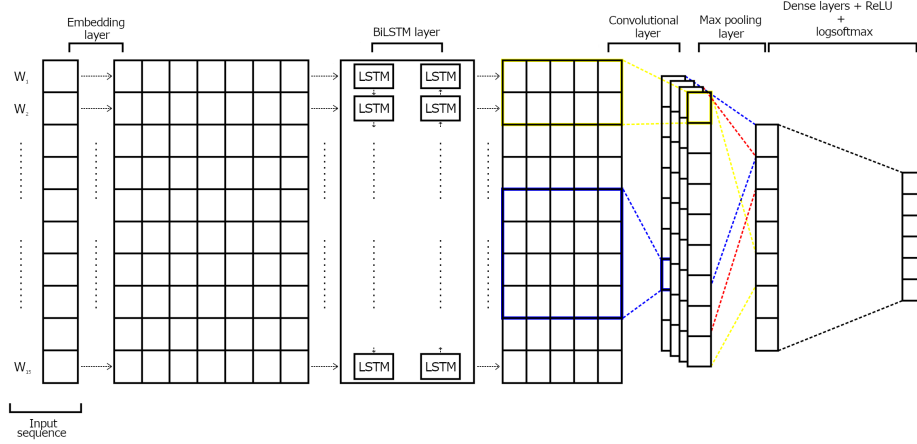


Figure 3: BiLSTM for text classification.

Early stopping is again used for selecting the optimal number of epochs. We use *Adam* as the optimization algorithm and the *negative log likelihood* as the loss function.

3.1.4. BERT

In this case, instead of using random initialization of the pre-trained GloVe embeddings, we now use the vectors provided by BERT to represent the input tokens. As opposed to the GloVe model (Pennington et al., 2014), BERT is taking into account the context of each word (that is, the words that surround it).

For the pre-processing of the texts the steps are similar to those described above. The main differences are that we tokenize the texts by using the BertTokenizer class from the transformers library (Face). This class has its own vocabulary with the mappings between words and ID's so it was not necessary to train a tokenizer with the corpus of texts. We also add the [CLS] and [SEP] tokens at the beginning and at the end of each tokenized sequence. It was also necessary to create an attention mask in order to distinguish what entries in each sequence correspond to real words in the input text and what entries are just 0's resulting from padding the sequences. Thus, the attention mask is composed of 1's (indicating non-padding entries) and 0's (indicating padding entries). We use the BERT base model in its uncased version (12 layers, 768 hidden size, 12 heads and 110 million parameters).

Then, we fine-tune it on our particular problem, that is, the multi-classification of fake news. To do this, we add a linear layer on top of the output of BERT that receives a vector of length 768 and outputs a vector of length 6.

For the training process, we used the *Adam* algorithm for optimization with a learning rate of $2 \cdot 10^{-5}$. We trained the model for two epochs, since the authors of BERT recommended using between two and four epochs for fine-tuning on a specific NLP task (Devlin et al., 2018).

3.2. Multimodal approach

Our multimodal approach uses a CNN that takes as inputs both the text and the image corresponding to the same news. The model outputs a vector of six numbers out of which the predicted class is obtained. In the following lines, we describe the preprocessing steps applied before feeding the data into the network as well as the architecture of the network.

For the input texts, we apply the same preprocessing steps described in 3.1.1. Regarding the preprocessing of the images, we only reshaped them so that all have the same shape (560 x 560). Once the pre-processed data is fed into the network, different operations are applied to text and image. The CNN architecture that we use for the texts is the same that we described in 3.1.2,

except for the fact that we eliminate the last 2 dense layers with ReLU activation in between.

We now describe the CNN model to classify the images. The data first goes through a convolutional layer. Since each image is made up three channels, the number of input cahnnels of this layer is also 3. Besides, it has 6 output channels. Filters of size (5 x 5) are used with stride equal to 1 and no padding. The output for each input image is therefore a collection of 6 matrices of shape (556 x 556). The output of the convolutional layer passes through a nonlinear activation function (ReLU) and then maxpooling is applied with a filter of size (2 x 2) and a stride equal to 2. The resulting output is a set of six matrices of shape (278 x 278). The output from the maxpooling layer passes again through another convolutional layer that has 6 input channels and 3 output channels. The filter size, stride length and padding are the same as those used in the previous convolutional layer. Then the ReLU non-linear activation function and the maxpooling layer are applied again over the feature maps resulting from the convolutional layer. Thus, for a given input (image) we obtain a set of 3 feature maps of shape (137 x 137). Finally, these feature maps are flattened into a vector of length 56307.

The outputs from the operations applied to each text and image are concatenated into a single vector. Then, this vector is passed through 2 dense layers with a ReLU non-linear activation in between. Finally, the logsoftmax function is applied and the logarithm of the probabilities is used in order to compute the predicted class of the given input.

4. Evaluation

4.1. Dataset

In our experiments, we train and test our models using the Fakeddit dataset (Nakamura et al., 2020), which consists of a collection of posts from Reddit users. It includes texts, images, comments and metadata. The texts are the titles of the posts submitted by users while the comment are made by other

users as a answer to a specific post. Thus, the dataset contains over 1 million instances.

One of the main advantages of this dataset is that it can be used to implement systems capable to perform a finer-grained classification of fake news than the usual binary classification, which only distinguishes between true and fake news. In the Fakeddit dataset, each instance has a label which distinguishes five categories of fake news, besides the unique category of true news. We describe briefly each category:

- True: this category indicates that the news is true.
- Manipulated Content: in this case, the content has been manipulated by different means (such as photo editing, for example).
- False Connection: this category corresponds to those samples in which the text and the image are not in accordance.
- Satire \Parody: this category refers to those news in which the meaning of the content is twisted or misinterpreted in a satirical or humorous way.
- Misleading Content: this category corresponds to those news in which the information has been deliberately manipulated or altered in order to mislead the public.
- Imposter Content: in the context of this project, all the news that belong to this category include content generated by bots.

The Fakeddit dataset is divided into training, validation and test partitions. Moreover, there are two different versions of the dataset: the unimodal dataset, whose instances only contains texts, and the multimodal dataset, whose instances have both text and image. Actually, all texts of the multimodal dataset are also included in the unimodal dataset.

Figure 4 shows the distribution of the classes in the unimodal dataset. Figure 5 provides the same information for the multimodal dataset. As we can see, all classes follow a similar distribution in both versions of the dataset (unimodal and

multimodal) as well as in the training, validation and test splits. Moreover, both datasets, unimodal and multimodal, are clearly imbalanced (the classes *true*, *manipulated content* and *false connection* have more instances than the other classes *satire*, *misleading content* and *imposter content*, which are much more underrepresented in both datasets). This imbalance may cause the classification task to be more difficult for those classes with less instances.

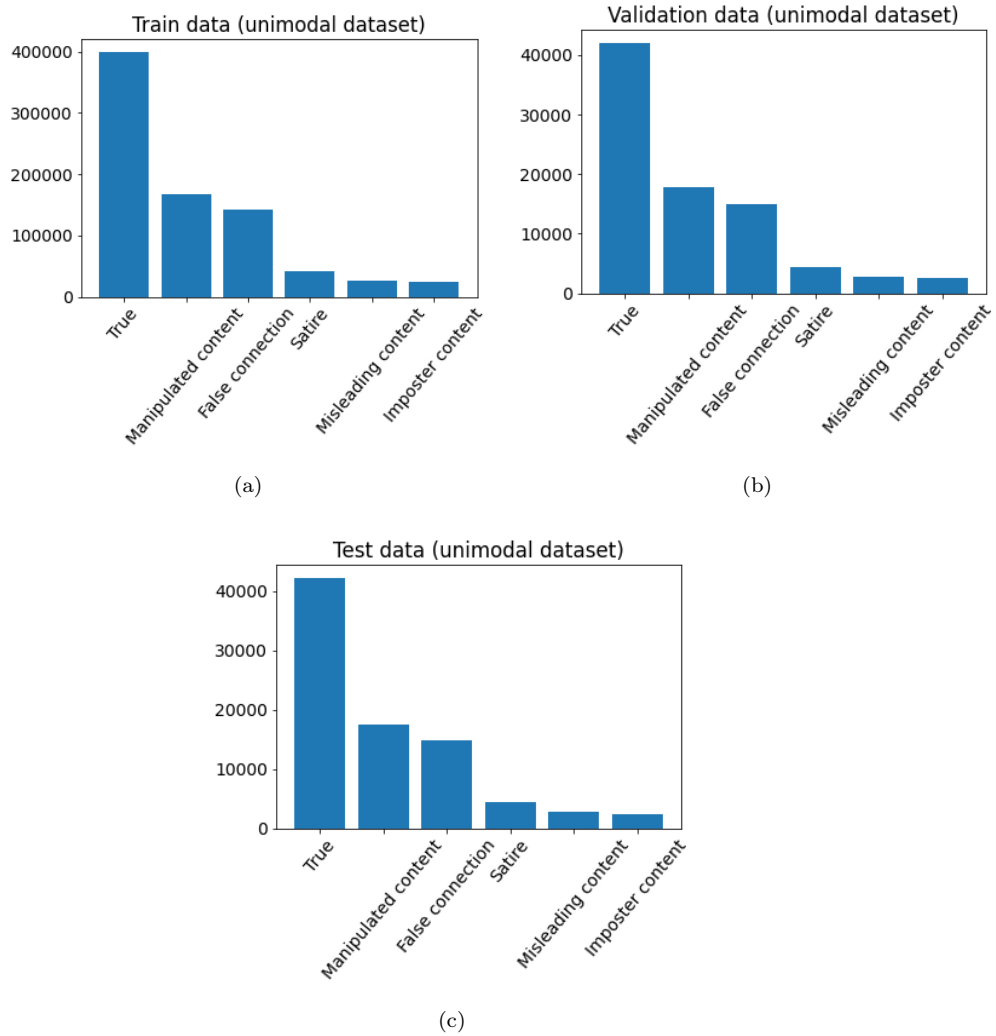


Figure 4: Class distribution (unimodal dataset).

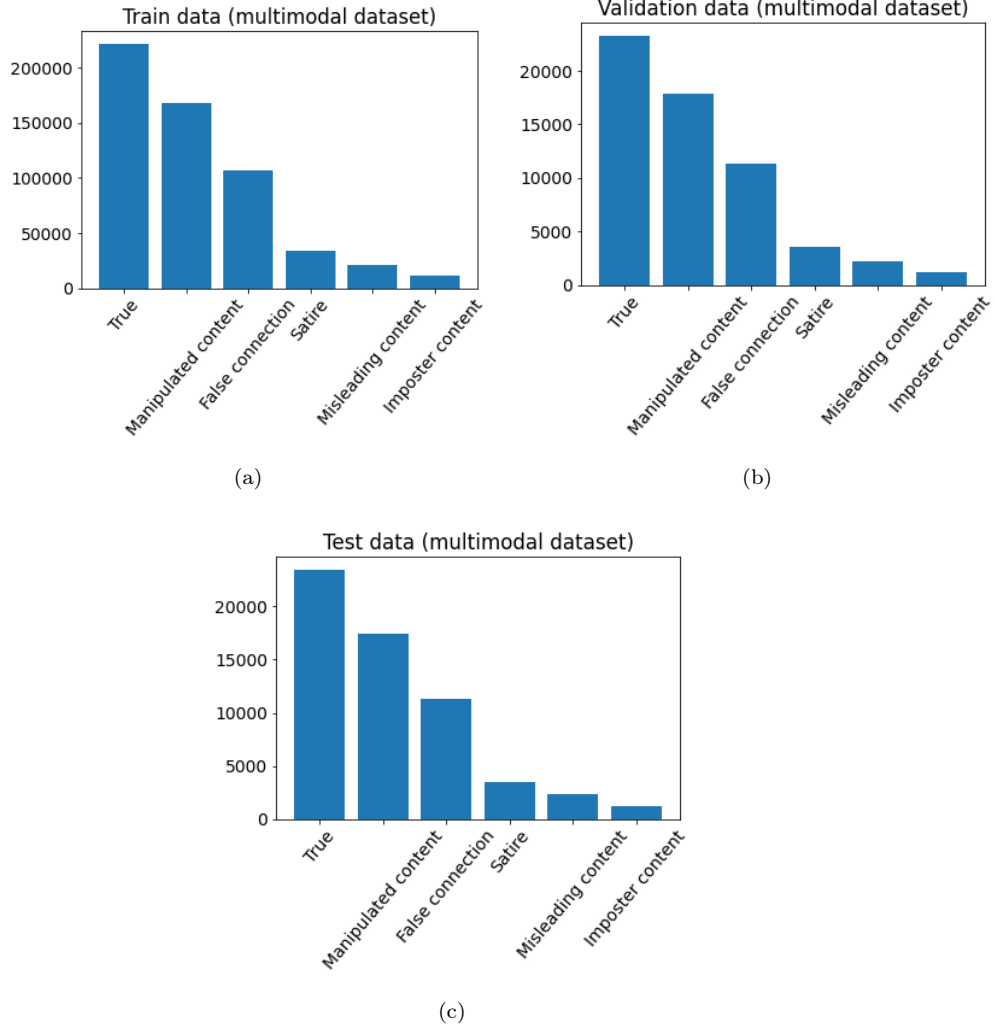


Figure 5: Class distribution (multimodal dataset).

4.2. Results

In this section, we present the results obtained for each model. We report the *recall*, *precision* and *F1* scores obtained by all the models for each class. The *accuracy* is computed over all the classes. It helps us to compare models and find the best approach. Moreover, we are also interested in knowing which model is better at detecting only those news containing false content. For this

reason, we also compute the *micro* and *macro* averages of the *recall*, *precision* and *F1* metrics only over five classes of fake news without the *true* news. We use the $F1_{micro}$ score and the accuracy to compare the performance of the models.

4.2.1. Unimodal approaches

CNN results.

Class	P	R	F1
True	0.71	0.87	0.79
Manipulated content	0.75	0.84	0.79
False connection	0.70	0.48	0.57
Satire	0.63	0.26	0.37
Misleading content	0.71	0.54	0.61
Imposter content	0.72	0.07	0.13
micro-average	0.73	0.62	0.57
macro-average	0.70	0.44	0.49
accuracy	0.72		

Table 1: Results of CNN with random initialization

Our first experiment with CNN uses random initialization to initialize the weights of the embedding layer, which are updated during the training process. This model obtains an accuracy of 72%, a micro F1 of 57% and a macro F1 of 49% (see Table 1). We can also see that True and Manipulated content are the classes with the highest F1 (79%). A possible reason for this could be that they are the majority classes. On the other hand, the model obtains the lowest F1 (13%) for Imposter content, which is the minority class in the dataset (see Fig. 4). Therefore, the results for the different classes appear to be related with the number of instances per class. However, the model achieves an F1 or 61% for the second minority class, Misleading content. As was explained before, the content of this news has been deliberately manipulated. Identifying these

manipulations appears to be easier than detecting humour or sarcasm in news (Satire) or fake news generated by bots (Imposter content).

Interestingly, although the model only exploits the textual content of the news, it achieves an F1 of 57% for classifying the instances of False connection. In these instances, the text and the image are not in accordance.

We also explore CNN with static (see Table 2) and dynamic (see Table 3) GloVe embeddings Pennington et al. (2014). In both models, the embedding layer is initialized with the pretrained Glove vectors. When dynamic training is chosen, these vectors are updated during the training process. On the other hand, if static training is chosen, the vectors are fixed during the training process. The model with dynamic vectors overcomes the one with static vectors, with a slightly improvement in accuracy (roughly one percentage point). However, in terms of micro F1, the static model is better than the dynamic one. Both models provide the same macro F1 (69%). Regarding the classes, there are not significant differences, except for Imposer content. For this class, updating the pre-trained Glove vectors results in a decrease of seven percentage points in F1.

Class	P	R	F1
True	0.76	0.81	0.79
Manipulated content	0.75	0.82	0.79
False connection	0.65	0.59	0.62
Satire	0.60	0.40	0.48
Misleading content	0.71	0.59	0.64
Imposter content	0.35	0.21	0.26
micro-average	0.70	0.67	0.69
macro-average	0.61	0.52	0.56
accuracy	0.73		

Table 2: Results of CNN with static Glove vectors

Class	P	R	F1
True	0.74	0.87	0.80
Manipulated content	0.76	0.83	0.80
False connection	0.71	0.54	0.61
Satire	0.67	0.35	0.46
Misleading content	0.74	0.58	0.65
Imposter content	0.70	0.11	0.19
micro-average	0.74	0.65	0.69
macro-average	0.71	0.48	0.54
accuracy	0.74		

Table 3: Results of CNN with dynamic Glove vectors

We also compare the effect of the pretrained Glove vectors with random initialization (see Table 1). In both dynamic and static approaches, initializing the model with the pretrained GloVe word embeddings gets better results than random initialization. The reason for this is that the GloVe vectors contain information about the relationship between different words that random vectors can not capture.

As the dataset is highly unbalanced, we use the micro F1 to assess and compare the overall performances of the three models. Thus, the best model is a CNN with dynamic Glove vectors. However, the dynamic training takes much more time than the static training (around 6000 to 8000 seconds more). This is due to the fact that, in a dynamic approach, word embeddings are also learned and this increases significantly the training time.

BiLSTM results.

As a second deep learning model, we explore a BiLSTM model. We replicate the same experiments as described for CNN, that is, using random initialization and pre-trained Glove vectors.

Class	P	R	F1
True	0.70	0.88	0.78
Manipulated content	0.73	0.85	0.79
False connection	0.73	0.44	0.55
Satire	0.58	0.25	0.35
Misleading content	0.71	0.54	0.61
Imposter content	0.86	0.08	0.14
micro-average	0.73	0.61	0.66
macro-average	0.74	0.41	0.48
accuracy	0.72		

Table 4: Results of BiLSTM with random initialization

The BiLSTM initialized with random vectors obtains very similar results to those achieved by CNN with random initialization (see Table 1). In fact, both models provides the same accuracy of 0.72. However, in terms of micro F1, the BiLSM model obtains up to 9 points more than the CNN model with random initialization. This improvement may because the BiLSTM improved its scores for Imposter content.

The use of static Glove vectors appears to have a positive effect on the performance of the BiLSTM model (see Table 5). The model shows significant improvements for False connection, Satire, Misleading content, and Imposter, with increases of 6, 12, 3, and 10 points, respectively. Therefore, the pretrained Glove vectors get better results than random initialization.

Class	P	R	F1
True	0.74	0.85	0.79
Manipulated content	0.77	0.82	0.79
False connection	0.68	0.55	0.61
Satire	0.55	0.41	0.47
Misleading content	0.77	0.55	0.64
Imposter content	0.45	0.17	0.24
micro-average	0.72	0.65	0.69
macro-average	0.65	0.50	0.55
accuracy	0.73		

Table 5: Results of BiLSTM with static Glove vectors

If the pretrained Glove vectors are updated during the training of the BiLSTM model, an accuracy of 75% is achieved, that is, two points more than BiLSTM with static Glove vectors. Moreover, this models with dynamic Glove vectors improves the results for all classes, with increases ranging from 1 to four points. In terms of micro F1, using dynamic Glove vectors is the best approach for BiLSTM. Moreover, this model slightly overcomes the CNN model with dynamic Glove vectors, roughly one percentage point. However, as mentioned above, the dynamic training takes much more time than static training.

Class	P	R	F1
True	0.75	0.86	0.80
Manipulated content	0.77	0.84	0.80
False connection	0.72	0.55	0.63
Satire	0.63	0.41	0.50
Misleading content	0.78	0.57	0.66
Imposter content	0.57	0.18	0.28
micro-average	0.74	0.67	0.70
macro-average	0.69	0.51	0.57
accuracy	0.75		

Table 6: Results of BiLSTM with dynamic Glove vectors

BERT results.

Class	P	R	F1
True	0.81	0.86	0.83
Manipulated content	0.80	0.86	0.83
False connection	0.72	0.64	0.68
Satire	0.70	0.53	0.61
Misleading content	0.77	0.70	0.73
Imposter content	0.61	0.28	0.38
micro-average	0.76	0.73	0.74
macro-average	0.72	0.60	0.65
accuracy	0.78		

Table 7: BERT results

Table 7 shows that BERT achieves an accuracy of 78% and a micro F1 of 74%. Therefore, it outperforms all the previous unimodal deep learning approaches. This proves the advantage of the pre-trained contextual text repre-

sentations provided by BERT, as opposed to the context-free GloVe vectors or random initialisation for neural networks.

Moreover, BERT is better in all classes. Comparing the classes, the behaviour of BERT is very similar to the previous deep learning models, that is, the more training instances for a class, the better predictions for it. In this way, True and Manipulated content both get the highest F1 (83%), while the worst performing class is Imposter content (F1=38%). As in previous models, Misleading content gets better scores than Satire, despite the fact that this class is more represented than the first one, Misleading content (see Fig. 5).

4.2.2. Multimodal approach

Class	P	R	F1
True	0.85	0.88	0.86
Manipulated content	1	1	1
False connection	0.77	0.76	0.76
Satire	0.82	0.72	0.77
Misleading content	0.75	0.79	0.77
Imposter content	0.46	0.25	0.32
micro-average	0.88	0.86	0.87
macro-average	0.76	0.70	0.72
accuracy	0.87		

Table 8: Multimodal approach results

Table 8 shows that the multimodal approach obtains an accuracy of 87% and a micro F1 of 72%, which are the highest scores out of all the unimodal models.

As expected, the training set size for each class strongly affects the model scores. While True and Manipulated content, the majority classes, get the highest scores, Imposter content, the minority class, shows the lowest F1 (32%), even six points lower than that provided by BERT for the same class (F1=38%).

Thus, we can say that the image content provides little information for identifying instances of *Imposter content*. Manipulated content shows an F1 of 100%. This is probably due to the fact that the images in this category, have been manipulated. These manipulations may be easily detected by CNN.

As expected, the use of images significantly improves the results for False connection. The multimodal model shows an F1 of 76%, 8 points up than that obtained by BERT, the best unimodal approach, and 15 points up than the unimodal CNN model using only texts. The improvement is even greater for detecting instances of Satire, with an increase of 16 points up than those obtained by BERT and by the unimodal CNN model.

4.2.3. Comparison of the best models

In addition to the deep learning algorithms, we also propose as baseline a Support Vector Machine (SVM), one of the most successful algorithm for text classification. Table 9 shows a comparative of the best models (traditional algorithms, CNN, BiLSTM, BERT and multimodal CNN) according to their accuracy and micro average scores.

Model	P	R	F1	Acc.
SVM	0.71	0.64	0.67	0.72
CNN (Dynamic + GloVe)	0.74	0.65	0.69	0.74
BiLSTM (Dynamic + GloVe)	0.74	0.67	0.70	0.75
BERT	0.76	0.73	0.74	0.78
Multimodal CNN	0.88	0.86	0.87	0.87

Table 9: Comparison of the best models (micro_averages).

In conclusion, we can see that the multimodal CNN outperforms all the unimodal approaches. This proves the usefulness of combining texts and images for a fine-grained fake news classification. Focusing on the unimodal approaches, the BERT model is the best both in terms of accuracy and micro F1 score, which shows the advantage of using contextual word embeddings. The third

best approach is the BiLSTM with dynamic GloVe vectors. Finally, all the deep learning approaches outperform our baseline SVM.

5. Conclusions

Fake news could have a significant negative effect on politics, health and economy. Therefore, it becomes necessary to develop tools that allow for a rapid and reliable detection of misinformation.

Apart from the work carried out by the creators of the Fakeddit dataset (Nakamura et al., 2020), this is, to the best of our knowledge, the only study that addresses a fine-grained classification of fake news by performing a comprehensive comparison of unimodal and multimodal approaches based on the most advanced deep learning techniques.

The multimodal approach overcomes the approaches that only exploit texts. BERT is the best model for the tasks of text classification. Moreover, using dynamic GloVe word embeddings outperforms random initialization for the CNN and BiLSTM architectures.

As future work, we plan to use pre-trained networks to generate the visual representations. In particular, we will use the network VGG , which was pre-trained on a large dataset of images such as *ImageNet*. We also plan to explore different deep learning techniques such as LSTM, BiLSTM, GRU or BERT, as well as, different methods to combine the visual and textual representations. In our current study, we have built our multimodal CNN using an early fusion approach, which consists on creating textual and visual representations, combining then, and then applying a classifier over the resulting combined representation to get the probabilities for each class. Instead of this, we plan to study a late fusion approach, which would require two separate classifiers (one for the textual inputs and the other for the image inputs). The predictions from both classifiers are then combined and the final prediction is obtained.

Acknowledgements

This work was supported by the Madrid Government (Comunidad de Madrid) under the Multiannual Agreement with UC3M in the line of "Fostering Young Doctors Research" (NLP4RARE-CM-UC3M) and in the context of the V PRICIT (Regional Programme of Research and Technological Innovation) and under the Multiannual Agreement with UC3M in the line of "Excellence of University Professors (EPUC3M17).

References

- Abavisani, M., Wu, L., Hu, S., Tetreault, J., & Jaimes, A. (2020). Multimodal Categorization of Crisis Events in Social Media. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 14667–14677). Los Alamitos, CA, USA: IEEE Computer Society.
- Aggarwal, C. C. (2018). Recurrent neural networks. In *Neural Networks and Deep Learning: A Textbook* (pp. 271–313). Cham: Springer International Publishing.
- Ahmed, H., Traore, I., & Saad, S. (2017). Detection of online fake news using n-gram analysis and machine learning techniques. In *International conference on intelligent, secure, and dependable systems in distributed and cloud environments* (pp. 127–138). Springer.
- Bae, K. I., Park, J., Lee, J., Lee, Y., & Lim, C. (2020). Flower classification with modified multimodal convolutional neural networks. *Expert Systems with Applications*, 159, 113455. doi:<https://doi.org/10.1016/j.eswa.2020.113455>.
- Baheti, P. (2020). *Introduction to Multimodal Deep Learning*. Retrieved from <https://heartbeat.comet.ml/introduction-to-multimodal-deep-learning-630b259f9291>. Accessed November 13, 2021.

- Barber, D. (2012). Naive bayes. In *Bayesian Reasoning and Machine Learning* (pp. 243–255). Cambridge: Cambridge University Press. doi:10.1017/CB09780511804779.
- Boididou, C., Andreadou, K., Papadopoulos, S., Dang-Nguyen, D.-T., Boato, G., Riegler, M., Kompatsiaris, Y. et al. (2015). Verifying Multimedia Use at MediaEval 2015. *MediaEval*, 3, 7.
- Breiman, L. (2001). Random Forests. *Machine Learning*, 45, 5–32. doi:<https://doi.org/10.1023/A:1010933404324>.
- Brown, E. (2019). *Online fake news is costing us \$78 billion globally each year*. ZDNet. <https://www.zdnet.com/article/online-fake-news-costing-us-78-billion-globally-each-year/>.
- Brownlee, J. (). A Gentle Introduction to Early Stopping to Avoid Overtraining Neural Networks. <https://machinelearningmastery.com/early-stopping-to-avoid-overtraining-neural-network-models/>. Accessed October 5, 2021.
- Choudhary, M., Chouhan, S. S., Pilli, E. S., & Vipparthi, S. K. (2021). BerConvoNet: A deep learning framework for fake news classification. *Applied Soft Computing*, 110, 107614. doi:<https://doi.org/10.1016/j.asoc.2021.107614>.
- Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, .
- Deng, L., & Liu, Y. (2018). *Deep learning in natural language processing*. (1st ed.). Springer.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, .

- Face, H. (). Transformers. <https://huggingface.co/transformers/>. Accessed October 5, 2021.
- Finneman, T., & Thomas, R. J. (2018). A family of falsehoods: Deception, media hoaxes and fake news. *Newspaper Research Journal*, 39, 350–361. doi:<https://doi.org/10.1177/0739532918796228>.
- Gaspar, A., & Alexandre, L. A. (2019). A multimodal approach to image sentiment analysis. In *International Conference on Intelligent Data Engineering and Automated Learning* (pp. 302–309). Springer, Cham.
- Giachanou, A., Zhang, G., & Rosso, P. (2020). Multimodal multi-image fake news detection. In *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)* (pp. 647–654). IEEE.
- Girgis, S., Amer, E., & Gadallah, M. (2018). Deep Learning Algorithms for Detecting Fake News in Online Text. In *2018 13th International Conference on Computer Engineering and Systems (ICCES)* (pp. 93–97). doi:10.1109/ICCES.2018.8639198.
- Goldani, M. H., Momtazi, S., & Safabakhsh, R. (2021). Detecting fake news with capsule neural networks. *Applied Soft Computing*, 101, 106991.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Convolutional networks. In *Deep learning* (pp. 321–362). Cambridge, MA, USA: MIT press.
- Greene, C. M., & Murphy, G. (2021). Quantifying the effects of fake news on behavior: Evidence from a study of COVID-19 misinformation. *Journal of Experimental Psychology: Applied*, . doi:<http://dx.doi.org/10.1037/xap0000371>.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009a). Additive models, trees, and related methods. In *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (pp. 295–336). New York, NY: Springer New York.

- Hastie, T., Tibshirani, R., & Friedman, J. (2009b). Support vector machines and flexible discriminants. In *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (pp. 417–458). New York, NY: Springer New York. doi:10.1007/978-0-387-84858-7_12.
- Hauck, G. (2017). 'Pizzagate' shooter sentenced to 4 years in prison. CNN. <https://edition.cnn.com/2017/06/22/politics/pizzagate-sentencing/index.html>.
- Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9, 1735–1780. doi:10.1162/neco.1997.9.8.1735.
- Hossain, M. Z., Sohel, F., Shiratuddin, M. F., & Laga, H. (2019). A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CSUR)*, 51, 1–36. doi:<https://doi.org/10.1145/3295748>.
- Hunt, A., & Gentzkow, M. (2017). Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 31, 211–236. doi:10.1257/jep.31.2.211.
- Islam, M., Kamal, A.-H., Kabir, A., Southern, D., Khan, S., Hasan, S., Sarkar, T., Sharmin, S., Das, S., Roy, T., Harun, G. D., Chughtai, A., Homaira, N., & Seale, H. (2021). COVID-19 vaccine rumors and conspiracy theories: The need for cognitive inoculation against misinformation to improve vaccine adherence. *PLOS ONE*, 16, Article e0251605. doi:10.1371/journal.pone.0251605.
- Jin, Z., Cao, J., Guo, H., Zhang, Y., & Luo, J. (2017). Multimodal fusion with recurrent neural networks for rumor detection on microblogs. In *Proceedings of the 25th ACM international conference on Multimedia* (pp. 795–816).
- Kaggle (a). FakeNewsNet. Retrieved from <https://www.kaggle.com/mdepak/fakenewsnet>. Accessed October 14, 2021.
- Kaggle (b). Getting Real about Fake News. Retrieved from <https://www.kaggle.com/mrisdal/fake-news>. Accessed October 13, 2021.

- Kaliyar, R. K., Kumar, P., Kumar, M., Narkhede, M., Namboodiri, S., & Mishra, S. (2020). DeepNet: An Efficient Neural Network for Fake News Detection using News-User Engagements. In *2020 5th International Conference on Computing, Communication and Security (ICCCS)* (pp. 1–6). doi:10.1109/ICCCS49678.2020.9277353.
- Kang, Z., Cao, Y., Shang, Y., Liang, T., Tang, H., & Tong, L. (2021). Fake news detection with heterogenous deep graph convolutional network. In K. Karlapalem, H. Cheng, N. Ramakrishnan, R. K. Agrawal, P. K. Reddy, J. Srivastava, & T. Chakraborty (Eds.), *Advances in Knowledge Discovery and Data Mining* (pp. 408–420). Cham: Springer International Publishing.
- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, .
- Kirchknopf, A., Slijepcevic, D., & Zeppelzauer, M. (2021). Multimodal Detection of Information Disorder from Social Media. *arXiv preprint arXiv:2105.15165*, .
- Kleinbaum, D. G., & Klein, M. (2010). *Logistic Regression*. (3rd ed.). Springer-Verlag New York.
- Kumari, R., & Ekbal, A. (2021). AMFB: Attention based multimodal Factorized Bilinear Pooling for multimodal Fake News Detection. *Expert Systems with Applications*, 184, 115412. doi:<https://doi.org/10.1016/j.eswa.2021.115412>.
- Li, P., Sun, X., Yu, H., Tian, Y., Yao, F., & Xu, G. (2021). Entity-Oriented Multi-Modal Alignment and Fusion Network for Fake News Detection. *IEEE Transactions on Multimedia*, (pp. 1–1). doi:10.1109/TMM.2021.3098988.
- Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021a). Deep learning-based text classification: A comprehen-

- sive review. *ACM Computing Surveys (CSUR)*, 54, 1–40. doi:<https://doi.org/10.1145/3439726>.
- Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021b). Deep Learning-based Text Classification: A Comprehensive Review. *ACM Computing Surveys (CSUR)*, 54, 1–40. doi:<https://doi.org/10.1145/3439726>.
- Mishra, V. (2019). *India’s fake news problem is killing real people*. Asia Times. <https://asiatimes.com/2019/10/indias-fake-news-problem-is-killing-real-people/>.
- Murphy, K. (2012). Kernels. In *Machine Learning: A Probabilistic Perspective* (pp. 479–512). Cambridge, MA: The MIT Press.
- Nakamura, K., Levy, S., & Wang, W. Y. (2020). Fakeddit: A new multi-modal benchmark dataset for fine-grained fake news detection. In *Proceedings of the 12th Language Resources and Evaluation Conference* (pp. 6149–6157). Marseille, France: European Language Resources Association. URL: <https://aclanthology.org/2020.lrec-1.755>.
- O’Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., & Walsh, J. (2019). Deep learning vs. traditional computer vision. In *Science and Information Conference* (pp. 128–144). Springer.
- Patel, K., & Bhattacharyya, P. (2017). Towards lower bounds on number of dimensions for word embeddings. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)* (pp. 31–36).
- Patwa, P., Sharma, S., Pykl, S., Guptha, V., Kumari, G., Akhtar, M. S., Ekbal, A., Das, A., & Chakraborty, T. (2020). Fighting an infodemic: Covid-19 fake news dataset. *arXiv preprint arXiv:2011.03327*, .

- Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543).
- Sabour, S., Frosst, N., & Hinton, G. E. (2017). Dynamic routing between capsules. *arXiv preprint arXiv:1710.09829*, .
- Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, .
- Sharma, Y., & Gupta, S. (2018). Deep learning approaches for question answering system. *Procedia computer science*, 132, 785–794. doi:<https://doi.org/10.1016/j.procs.2018.05.090>.
- Shu, K., Mahudeswaran, D., Wang, S., Lee, D., & Liu, H. (2020). FakeNewsNet: A Data Repository with News Content, Social Context, and Spatiotemporal Information for Studying Fake News on Social Media. *Big data*, 8, 171–188. doi:10.1089/big.2020.0062.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, .
- Singh, V. K., Ghosh, I., & Sonagara, D. (2021). Detecting fake news stories via multimodal analysis. *Journal of the Association for Information Science and Technology*, 72, 3–17.
- Singhal, S., Shah, R. R., Chakraborty, T., Kumaraguru, P., & Satoh, S. (2019). SpotFake: A multi-modal framework for fake news detection. In *2019 IEEE fifth international conference on multimedia big data (BigMM)* (pp. 39–47). IEEE.
- Tang, D., Qin, B., & Liu, T. (2015). Deep learning for sentiment analysis: successful approaches and future challenges. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 5, 292–303. doi:<https://doi.org/10.1002/widm.1171>.

- Thota, A., Tilak, P., Ahluwalia, S., & Lohia, N. (2018). Fake news detection: a deep learning approach. *SMU Data Science Review*, 1, 10.
- Viana, M., Nguyen, Q.-B., Smith, J., & Gabrani, M. (2017). Multimodal Classification of Document Embedded Images. In *International Workshop on Graphics Recognition* (pp. 45–53). Springer.
- Voita, E. (2021). Convolutional Neural Networks for Text. https://lena-voita.github.io/nlp_course/models/convolutional.html. Accessed October 5, 2021.
- Wang, W. Y. (2017). "Liar, Liar Pants on Fire": A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*, .
- Wang, Y., Ma, F., Jin, Z., Yuan, Y., Xun, G., Jha, K., Su, L., & Gao, J. (2018). EANN: Event adversarial neural networks for multi-modal fake news detection. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 849–857).
- Wani, A., Joshi, I., Khandve, S., Wagh, V., & Joshi, R. (2021). Evaluating deep learning approaches for Covid19 fake news detection. In *Combating Online Hostile Posts in Regional Languages during Emergency Situation: First International Workshop, CONSTRAINT 2021, Collocated with AAAI 2021, Virtual Event, February 8, 2021, Revised Selected Papers* (p. 153). Springer Nature.
- Xie, J., Liu, S., Liu, R., Zhang, Y., & Zhu, Y. (2021). SERN: Stance Extraction and Reasoning Network for Fake News Detection. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 2520–2524). doi:10.1109/ICASSP39728.2021.9414787.
- Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016). Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies* (pp. 1480–1489).

- Yu, J., & Jiang, J. (2019). Adapting BERT for Target-Oriented Multimodal Sentiment Classification. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19* (pp. 5408–5414).
- Zhao, Z.-Q., Zheng, P., Xu, S.-T., & Wu, X. (2019). Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, *30*, 3212–3232. doi:10.1109/TNNLS.2018.2876865.
- Zubiaga, A., Liakata, M., & Procter, R. (2017). Exploiting context for rumour detection in social media. In *International Conference on Social Informatics* (pp. 109–123). Springer.