Fake news detection:

An Approach based on Detecting the Stance of Headlines to Articles

Prof. Gourab Nath
Department of Data Science
Praxis Business School
Bengaluru, India
gourab.nath@praxis.ac.in

Anurag Sharma
Department of Data Science
Praxis Business School
Bengaluru, India
anurag.sharma@praxis.ac.in

Pooja N.
Department of Data Science
Praxis Business School
Bengaluru, India
pooja.n@praxis.ac.in

Atul Kumar Pandey
Department of Data Science
Praxis Business School
Bengaluru, India
atul.pandey@praxis.ac.in

Shreyansh Bhalodiya
Department of Data Science
Praxis Business School
Bengaluru, India
shreyansh.bhalodiya@praxis.ac.in

Deepak Sharma
Department of Data Science
Praxis Business School
Bengaluru, India
deepak.sharma@praxis.ac.in

***ABSTRACT*- In recent times, social media has become the main channel for generating fake news .This has become a global threat due to the increase in emerging technologies of connecting people at the fastest.The exponential increase in fake news generation and distribution of inaccurate news creates an immediate need for automatically tagging and detecting such manipulated news articles. However, it is difficult to automate the fake news detection as it requires the model to understand nuances in natural language. Moreover, the majority of the existing fake news detection models treat the problem at hand as a binary classification task, which limits the model's ability to understand how related or unrelated the reported news is when compared to the real news. To address these gaps, we have designed a framework where we integrate the machine learning model and transformer model to accurately predict the stance between a given pair of headlines and article bodies. In this project, we have come up with the applications of NLP for detecting 'fake news' based on the concept of 'Stance detection'. We focused on classifying news articles from unknown sources as either "agreeing" or "disagreeing" by comparing with sources of known credibility. This approach is divided into two categories as content-based learning and context-based learning. The content-based approach is based on extraction of linguistic features such as lexical, syntactic, semantic features, whereas the context-based approach captures the contextual information which includes sequence and collocation of words, negation handling and ambiguity. To overcome increased class misclassification rate, the proposed context-based BERT classification model was evaluated by comparing its performance with machine learning classifier models (Naive Bayes, Support Vector Machine and Random Forest Model) and the outcome showed a better classification of false information for our work. The detection performance was improved in two aspects – a reduction in the detection runtime and an increase in the classification accuracy.**

***Keywords:* Jaccard similarity , glove , n-gram , semantic similarity, Latent dirichlet allocation(LDA), BART summarizer ,BERT-Bidirectional Encoder Representation from Transformers.**

1. INTRODUCTION

Why fake news detection needs of the hour because it has tremendous effects. [2]. During the 2016 US presidential election, various kinds of fake news about the candidates were widely spread in the online social networks, which may have a significant effect on the election results. According to a post-election statistical report [4], online social networks account for more than 41.8% of the fake news data traffic in the election, which is much greater than the data traffic shares of both traditional TV/radio/print medium and online search engines respectively. In the past few years, various social media platforms such as Twitter, Facebook, Instagram, etc. have become very popular since they facilitate the easy acquisition of information and provide a quick platform for information sharing. The availability of unauthentic data on social media platforms has gained massive attention among researchers and become a hot-spot for sharing fake news [3]. Fake news has been an important issue due to its tremendous negative impact. It has increased attention among researchers, journalists, politicians and the general public. In the context of writing style, fake news is written or published with the intent to mislead the people and to damage the image of an agency, entity, person, either for financial or political benefits. In the research context, related synonyms (keywords) often linked with fake news:

Rumor: A rumor is an unverified claim about any event, transmitting from individual to individual in the society. It might imply an occurrence, article, and any social issue of open public concern. It might end up being a socially dangerous phenomenon in any human culture.

Hoax: A hoax is a falsehood deliberately fabricated to masquerade as the truth. Currently, it has been increasing at an alarming rate. Hoax is also known with similar names like prank or jape.

The big technology companies Twitter, Facebook, Google have spotted this danger and have already begun to work on systems to detect the fake news on their platforms. The main objective of this work is to: The goal of this project is, given an article's headline and body text, predict whether that body's text agrees with the headline, disagrees with the headline, discusses the same topics/claim as the headline, or is unrelated to the [1].

2. LITERATURE REVIEW

2.1 Existing approaches for fake news detection:

Existing learning for fake news detection can be generally categorized as (i) News Content-based learning and (ii) Social Context-based learning. News content-based approaches [5] deal with different writing styles of published news articles. In these techniques, our main focus is to extract several features in fake news articles related to both information as well as the writing style.

2.2 Limitations of Machine Learning models for text classification:

Traditional text classification methods based on machine learning have many disadvantages such as dimension explosion, data sparsity, limited generalization ability and so on [7] studies have proved that text classification methods based on deep learning outperform the traditional methods (Machine Learning) when processing large-scale and complex datasets.

2.3 RNN vs Transformers

The turning point for NLP came when transfer learning became possible by training a language model and using the information learned from the language model in many other NLP tasks. This is also referred to as the NLP's ImageNet moment [8].

In addition to transfer learning, Transformers started a new era in the NLP field. Transformers are deep learning models that can handle sequential data but they don't require sequential data to be processed in order, unlike recurrent neural networks (RNNs) [9] Therefore they are parallelizable, reducing the time it takes to train and enabling scientists to train on much larger datasets. Transformers are highly successful in many NLP tasks, including summarization, translation, and classification. BERT is one of the architectures that utilizes Transformers and the model, trained in an unsupervised manner on large datasets, can be utilized in many other NLP tasks [10].

2.4. Need of Language Models

Language models are a critical component of state-of-art systems for modern NLP applications. It turns the qualitative information into quantitative for better prediction and understanding.

2.4.1. BERT Based Approach:

BERT is designed to pretrain deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers[10]. As a result, the pre-trained BERT model can be finetuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task specific architecture modifications [10] .In this paper, we propose a BERT-based deep learning approach for detecting fake news. We utilize BERT as a sentence encoder, which can accurately get the context representation of a sentence. This work is in contrast to previous research works where researchers looked at a text sequence in a unidirectional way (either left to right or right to left for pre-training). Many existing and useful methods had been presented with sequential neural networks to encode the relevant information. Deep neural networks with a bidirectional training approach can be an optimal and accurate solution for the detection of fake news [6]. Our proposed method improves the performance of fake news detection with the powerful ability to capture semantic and long-distance dependencies in sentences.

2.4.2. Limitations of BERT:

At a time only 512 tokens can be given as input in the BERT model. If text length exceeds the limit, then the entire text would be truncated to 512 tokens [11].

3. METHODOLOGY

Did you know Whats App charges its users for sending messages? I was shocked that the free app that we use on a daily basis will charge us now for sending messages. But before getting shocked just think the claim which we entered is real or fake. Maybe it can be fake but how will we identify it? We can simply google search and find websites showing the related news. We can also get many articles showing the news but now what if the claim is fake will newspapers like the Times of India and India Today will show such news obviously not. So, if the claim is fake then we will find very few websites/newspapers showing such news especially since no reputed sites will show fake news. Now let's say we search that claim on google and get 20 articles but if the news is fake then google may show both related and unrelated articles for the same claim. Now what is the meaning of related and unrelated. Related means we search something about WhatsApp and we get news related to WhatsApp only but if we get news other than WhatsApp it is unrelated. Now among searched claims let's say we get 15 related articles and among 15 related articles few articles will agree to that claim, few will disagree and few will discuss the claim but will not take any stance. But among agree, disagree, discuss how you will decide whether the claim is real or fake. Here we have to keep one thing in mind: we will take newspaper sites which are reputed only, we won't take all sites because all reputed sites show only the real claims. This is how our model will work. Further, this model is divided into two parts Baseline model-1 and Baseline model-2. Now what these models will do and how they will function let's see in detail.

3.1. Data Description:

The data set for this project comes from a publicly available platform, Fake News Challenge or FNC-1. This dataset is divided into a training set and a testing set. The ratio of training data over testing data is about 2:1. This dataset provides a total of 75385 articles (headline-body pairs) (split into 49972 articles in the training dataset and 25413 articles in the test dataset). - Each article (or headline-body 25413 pair) is labelled with either 1. Unrelated, 2. Discuss, 3. Agree, or 4. Disagree. Data for the testing part was available in the dataset we took but we scrapped the real-time data from google using google article scraper. The first time we scrapped around 400 articles from google and gave them labels manually. The second time we scrapped around 200 articles but this time we instead of giving labels manually we generated labels through model prediction. Later, the top 55-60 articles were scrapped for each claim from all the links shown by google. But to improve the accuracy of our final model, instead of scrapping the top 50-60 links we made a list of 162 popular and reputed newspapers and then we scrapped articles that were shown in those newspapers. So, from our final scrapper, we again scrapped around 175 headlines with an average of 10 articles per claim ,resulting in 2000 articles. Now we will predict related, unrelated, agree, disagree, discuss articles for each headline and then give real and unreal labels for each headline manually. Now we will select a threshold based on this final data and will predict whether the news is fake or real.

3.2. Baseline model-1

The constructed web scraper will scrape the articles from reputed websites and the claim is searched on google and the articles are extracted. Next, what Baseline model- 1 will do? The main task of the Baseline model-1 is to divide each article into related and unrelated. But how we will decide whether a claim is related or unrelated to the article. In this section, we implement a Feature-oriented technique to determine effective features for detecting the related and unrelated articles. In order to extract features from text information, we introduce linguistic-based techniques through which 7 features were generated. Let's see what each feature does.

3.2.1. Feature creation:

(1) Jaccard similarity

Feature, Jaccard similarity checks which are the common words between headline and article, and it calculates the cosine similarity between them[1,12].

(2) Jaccard similarity Nouns

This feature is extracted from Jaccard similarity. We sort nouns from headlines and articles, then calculate Jaccard similarity between them. We did this to check whether the headline and article are talking about the same subject/ person/Object in the same Nouns.

(3) Latent Dirichlet Allocation (LDA)

Feature, LDA takes the topic similarity between headline and article. For each word in headline and article, it assigns some topic, and then based on those it checks whether headline and article are talking about common topics or different topics[13].

(4) Semantic Similarity

We may get very good Jaccard similarity if we have a lot of common words between headline and article but how can we check whether they mean the same or different. To check that we may use semantic features as one of the features. The semantic similarity feature checks whether the headline and article meant the same [12].

(5) Glove Similarity

The glove is an unsupervised learning algorithm for obtaining vector representations for words. Training is performed on aggregated global word-word co-occurrence statistics from a corpus, and the resulting representations showcase interesting linear substructures of the word vector space [1].

(6) K-L Divergence

Feature, KL divergence can be calculated as the negative sum of the probability of each event multiplied by the log of the probability of the event in Q over the probability of the event in P. The value within the sum is the divergence for a given event. Vector of each headline and article is created and then technique is applied[1].

(7) N-Grams

Feature N-Grams are a set of co-occurring words within a given window size and when computing the n-grams we typically move one word forward (although we can move X words forward in more advanced scenarios) [1,5].

What Next?

We collected headlines, articles and created 7 feature matrices. To predict related and unrelated articles any 2-Class classification model with those 7 features as predictors can be used. Next, as a part of feature selection technique we trained the model with these 7 features and also on a combination of different features. We recorded all the results obtained from different combinations of features into a list. Next, we used the

Random forest model on those 7 features combination list. We are using a random forest model because it gave the best accuracy among the 2 class classification models that we had trained. As a result,the best results were obtained from three combinations of features namely Jaccard similarity nouns, glove similarity, and N-gram.When these features as predictors were used in random forest models to classify the articles into related and unrelated, we achieved 98% model accuracy compared to all other combinations of features. Hence, these 3 features became effective in determining the 2 class labels.

This approach, however, has some drawbacks for 4-class labels. Firstly, Baseline model-1 doesn't capture the contextual meaning of the articles which are considered important in understanding the relation between sentences in the articles. Secondly, Baseline model-1 was not able to clearly differentiate the articles into agree, disagree, discuss, and unrelated labels. Due to this, disagree articles were falsely classified into discuss and unrelated.Hence, we choose 2-class prediction with machine learning model.

To overcome these drawbacks, we proposed Baseline model-2 as an improvement phase. Later, it was integrated with the Baseline model-1 as a complete solution framework for Stance detection.

3.3. Baseline model-2

We aim to design a framework that will be able to take related articles from Baseline model-1 and summarize the articles to classify them into one of the 3 categories- agree, disagree, and discuss. This is an unsupervised learning framework because it uses a Transformer-Based model. Here, we implement the BART model for summarization tasks, and summarized articles are given as an input to the BERT model for sentence-pair classification tasks.

3.3.1. BART model for Summarization task

BART is a Bidirectional Auto-Regressive Transformer that combines the Bidirectional Encoder (i.e., BERT) with Autoregressive decoder into one seq2seq model. We use the BART model to summarize the articles by reducing their length without losing any information. The BART Model with a language modelling head can be used for summarization. Pretraining BART involves token masking (like BERT does), token deletion, text infilling, sentence permutation, and document rotation[14]. Since BART is a pre-trained model, it can be fine-tuned to a more specific task, such as text summarization. From the Machine Learning model, the news articles which are predicted as related are passed through the bidirectional encoder, i.e., the BERT-like encoder. By consequence, articles are looked at from left-to-right and right-to-left, and the subsequent output is used in the autoregressive decoder, which predicts the output based on the encoder input and the output tokens predicted so far. In other words, with BART, we can now both understand the inputs really well and generate new outputs. That is, we can e.g., finetune a model for a text summarization task. Next, we will fine-tune the BART model specific to the text summarization task.

Fine Tune Bart Configuration: Pretrained Model- 'Facebook / Bart-large', with 24-layers, 1024-hidden, 16-heads, 406M parameters, Max-Length =1024.

3.3.2. BERT for Stance detection

Bidirectional Encoder Representations from Transformers. Unlike recent language representation models, BERT is designed to pre-train deep bidirectional representations from the unlabelled text by jointly conditioning on both left and right context in all layers[11]. As a result, the pre-trained BERT model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks [11], we have downstream task to specific task of Sequence Classification model for Multi-label Text classification problem, without substantial task-specific architecture modifications.This section proposes a BERT-based model for text classification, BERT4TC-BERT for Text Classification, via constructing auxiliary sentences and incorporating domain-related knowledge. Here, the summarized articles from the BART model are given as an input to the BERT model. We convert BERT model to supervised learning by training the model with FNC data with labels BERT4TC consists of three parts as follows[11]:

(1) Input layer: Firstly, we input Headlines (claim) and articles parallelly into the BERT model. It aims to build an input sequence for the model by combining the two-input sentences and turn the task into a sentence-pair one. The input layer has three constructing stages namely position embeddings, word embeddings, and segmentation embeddings to convert raw input text into a specific input sequence. Each segmented input sequence includes only 512 tokens [11]. To overcome this limitation, we have used BART summarizer for reducing the article length and thereby avoiding information leakage.

(2)BERT encoder: We only need encoders for text classification tasks. It consists of 12 Transformer blocks and 12 self-attention heads by taking an input of a sequence of no more than 512 tokens and outputting the representations of the sequence. The output representation is a final hidden state vector.

(3) Output layer. It is a classification layer where we train our BERT pre-trained model specific to the Classification task by training the set of articles against target labels. The labels are predicted with a simple SoftMax classifier which calculates the conditional probability distributions over pre-defined categorical labels. Let θ be the set of all trainable parameters for BERT4TC [11], the output layer turns the vector H[CLS] into the conditional probability distributions $P(y_i | H_{[CLS]}, \theta)$ over all categorical labels $y = \{y_1, . . . , y_c\}$ (or $y = \{0, 1, 2\}$) as follows:

$$P(y_i|H_{[CLS]},\theta) = softmax(H_{[CLS]}V^T)$$

$$= \frac{\exp(P(y_i|H_{[CLS]},\theta))}{\sum_{j=1}^{c}\exp(P(y_j|H_{[CLS]},\theta))} \quad (1)$$

where $V \in R^{c\times h}$ is the trainable task-specific parameter matrix and c is the number of labels. Let 't' be the true label of the input sequence 'x', we take the label with the largest $y_x = \text{argmax} (P(y_i |H_{[CLS]} , \theta))$ value as the predicted result and compute a standard calculation loss J(x, θ) based on the canonical cross-entropy function as follows [11] :

$$J(x,\theta) = \begin{cases} -t\ln P(y_x) - (1-t)\ln(1-P(y_x)) & ifc = 2 \\ -\ln P(y_x) & ifc > 2 \end{cases} \quad (2)$$

We use the parameter batch size to denote the number of each training batch. To avoid over-fitting, the regularization strategy Dropout is adopted and the value is always kept at 0.1. BERT4TC uses the default Adam optimizer with beta1=0.9 and beta2=0.999. We fine-tune all trainable parameters from BERT4TC by maximizing the log-probability of the correct label.

*3.3.3. Fine - tune BERT Configuration***:**

Pretrained parameters are fine tuned to perform Sentence-pair Classification task for Stance detection:

Pretrained model name: BERT-Base,Uncased-12-layer, 768-hidden, 12-heads, 110M parameters:

BERTConfiguration: Model Architecture–BertForSequenceClassification

Model arguments:

(1) model class: "Classification Model"
(2) model type: 'Bert'
(3) model name: ' bert-base-uncased'
(4) num labels: 3
(5) args:{'learning_rate':3e-5, 'num_train_epochs': 5, 'reprocess_input_data': True, 'process count': 10, 'train_batch_size': 4, 'eval_batch_size': 4, 'max_seq_length': 512, 'fp16': True}

3.3.4.Evaluation of Classification Model based on:

(1) mcc - Matthew's correlation coefficient (value ranges between –1 to +1, a model with +1 is a perfect model and –1 is a poor model)

(2) eval_loss-Cross Entropy loss, measures the performance of a classification model we calculate a separate loss for each class label per observation and sum the result (2).Cross Entropy loss increases as predicted probability diverges from actual labels.

4. EXPERIMENTAL RESULTS

In this section,we use the FNC dataset to evaluate our models.

4.1..Dataset description

FNC dataset dimension is 75385 ,it was further divided into 49972 as train set, 25413 as test set .

4.2. Baseline model-1: ML model

From Table 1,we can see that ML models performs well for 4-class classification task, but when we see the misclassification rate from Figure:1, Baseline model-1 has a high misclassification rate of 14%. Hence, we propose our final ML model for 2-class classification using Random Forest classifier.

Table 2 and Table 3 shows the Training and Test results respectively , we can notice that model accuracy was improved when we grouped the classes into 2-class compared to previous approach and misclassification rate was also reduced to 1.5%.

TABLE 1: **Test Result on FNC data for 4 class**

MACHINE LEARNING MODEL	MODEL RESULTS 4-CLASS
LogisticRegression, SVC, SGD classifier	87%
QDA , Linear discriminant analysis (LDA)	86.5%
RandomForest,Adaboost, Gaussian Naive bayes	85%
DecisionTree,Xgboost	83%
KNN	80%

TABLE 2: **Train Results on FNC data for 2 class**

Random Forest Classifier (ML model)	Performance metrics			
	Precision	Recall	F1-Score	Train Accuracy
Related	0.96	0.98	0.97	98%
Unrelated	0.99	0.99	0.99	

TABLE 3: **Test Results on FNC data for 2 class**

Random Forest Classifier (ML model)	Performance metrics			
	Precision	Recall	F1-Score	Test accuracy
Related	0.88	0.88	0.88	94%
Unrelated	0.96	0.96	0.96	

4.2. Baseline model-2:Fine tune BERT model

TABLE 4: **Dataset description**

FNC –1	Data set Dimension for 3 class labels			
	Agree	Disagree	Discuss	Total
Train set	3742	1701	7394	12837
Validation set	1193	541	2419	4153
Test set	2237	1069	4643	7949

For fine-tuning the BERT-pretrained model into a text classification task, we use the same FNC dataset for training the model with specific hyperparameters.

TABLE 5 : **BERT model results for 3-class on FNC dataset**

BERT Model Results	Performance metrics					
	F1-score : Agree	F1-score: Disagree	F1-score : Discuss	Accuracy	Evalloss	mcc
Train set	0.98	0.96	0.99	98%	0.062	0.97
Validation set	0.94	0.69	0.94	88 %	0.73	0.78
Test set	0.66	0.46	0.84	75 %	1.8	0.54

To know the performance of the BERT classification model, evaluation loss is the key performance metric and to understand whether or not our model is good or not, (mcc) Matthew's correlation coefficient is the key performance metric. From TABLE 5 , we can notice that Baseline model-2 results in 98% accuracy, performs the task with minimum loss and turns out to be a good model.

4.3. Misclassification rate

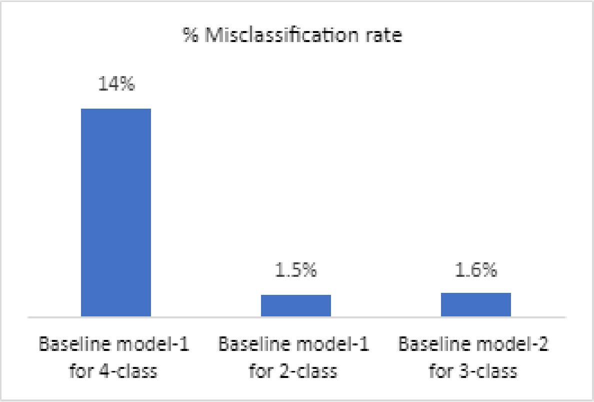

FIGURE 1: **Misclassification rate for Baseline model-1 and Baseline model-2**

5. CONCLUSION & FUTURE WORK

Classifying the "Fake News", we initially performed a content-based approach where we try to extract features at the word level. We had extracted 7 features out of which 3 features Jaccard similarity nouns, glove similarity, and N-gram were selected through the feature selection technique. Later, these features were incorporated into machine learning models for 4-class prediction such as SVM, Logistic regression, Random forest classifier, KNN, QDA, Adaboost classifier, SGD, Xgboost, Decision Tree, LDA, and Gaussian Naive Bayes. As a result, Logistic Regression, SVC, and SGD classifiers performed well but the misclassification rate was more for these models. Hence, we came up with the 2 stage prediction model. i,e. Stage-1 includes Baseline model-1 for 2-class prediction(related and unrelated) and Stage-2 Baseline model-2 for 3-class prediction (agree, disagree, discuss).Random Forest Classifier as a Baseline model-1 was used for 2-class prediction and observed 94% model accuracy. In stage-2, we performed a context-based approach where we try to extract the features at the sentence level..i,e. the entire meaning of the article. Baseline model-2 includes transformer models such as BART summarizer model and BERT Classification model. Since BERT input length is only 512 tokens we used BART summarizer to reduce the article length to avoid information leakage during the process of BERT 3-class prediction. The Baseline model-2 resulted in 98% model accuracy and even the misclassification rate was less compared to 4-class prediction. This approach of breaking down the complex problem into a two-stage problem-solving framework helped us to overcome the potential problem. i.e., to differentiate the articles into one of the four stances.

5.1. Shortfall Identification

As discussed in earlier sections, when we tested our final model with real-time scraped data (2000 articles) for 175 claims. In order to output the model result as Fake or Real news ,we introduced the concept of thresholding for predicted output. Here, we used two conditions: (1) If the claim has predicted frequency count less for agree labels than compared to the sum of disagree and agree labels and (2) If frequency count of discuss is less than disagree, then we output that claim as Fake news, or else Real News.As a result of imposing conditional criteria, 55 claims out of 74 fake claims were correctly classified as fake news and on other hand only 26 claims out of 101 Real claims were classified as Real news. The reason for the high misclassification rate in Real news is due to the presence of an imbalanced dataset(FNC-1) which we had during training the model.Only 21% related articles were available in FNC dataset and that was used for training the 3-class model. Hence, At this stage we restrict our work to 3-class prediction.

There is a scope for improvements in future work. Though our model is working well for labelled dataset.We can improve the same on real-time data by training the BERT model on even more large dataset with equal proportion of real and fake articles.The articles should not only be relevant to the claim but also credible. There are some newspapers which have different writing styles that are written in a manipulative way. We can incorporate this feature in deciding the credibility of the source. This is known as a style-based method where we can assess news intention like is there an intention to mislead the public or not?.Some patterns of fake news style are distinguishable from those of the true news. Fake news text, compared to true news text has higher (i) informality (% swear words), (ii) diversity (% unique verbs), (iii) subjectivity (% report verbs), and is (iv) more emotional (% emotional words).

REFERENCES

[1] Mr. Mohammad Zeeshan Ansari Shaz Akhtar, Assistant Professor Abhinav Kumar Jha ,“AUTOMATIC STANCE DETECTION A PROJECT REPORT”.

[2] Jiawei Zhang1 , Bowen Dong2 , Philip S. Yu2 1 IFM Lab, “FAKEDETECTOR: Effective Fake News Detection with Deep Diffusive Neural Network”.

[3] Rohit Kumar Kaliyar1 ·Anurag Goswami1 · Pratik Narang2 Received: 1 May 2020 / Revised: 24 August 2020 / Accepted: 11 November 2020 / © Springer Science and Business Media, LLC, part of Springer Nature 2021,” Fake BERT: Fake news detection in social media with a BERT-based deep learning approach”.

[7] Hongping Wu1, Yuling Liu1 and Jingwen Wang,” Review of Text Classification Methods on Deep Learning”

[8] Sebastian Ruder. “Nlp’s ImageNet moment has arrived. Gradient”, July, 8, 2018.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, “Attention is all you need”. CoRR, abs/1706.03762, 2017.

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova., “BERT: pre-training of deep bidirectional transformers for language understanding”. CoRR, abs/1810.04805, 2018.

[11] SHANSHAN YU ,Ph.D. Lecture. Senior member of China Computer Federation JINDIAN SU, born in 1980. Ph.D. Ass. Professor. Member of CCF.DA LUO is currently pursuing the master’s degree with the South China University of Technology, Guangzhou, China, “Improving BERT-based Text Classification with Auxiliary Sentence and Domain Knowledge”.

[12] A Survey Goutam Majumder, Partha Pakray National Institute of Technology Mizoram, Aizawl, India. goutam.nita@gmail.com, parthapakray@gmail.com. Alexander Gelbukh Instituto Politécnico Nacional, CIC, Mexico City, Mexico. gelbukh@gelbukh.com David Pinto Benemerita Universidad Autonoma de Puebla, Faculty of Computer Science, Mexico. davideduardopinto@gmail.com , “Semantic Textual Similarity Methods, Tools, and Applications”.

[13] Claus Boye Asmussen* and Charles Møller, “Smart literature review: a practical topic modelling approach to exploratory literature review”.

[14] Mike Lewis*, Yinhan Liu*, Naman Goyal*, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, Luke Zettlemoyer, “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension”.

[4] H.Allcott and M. Gentzkow, "Social media and fake news in the 2016 election”. Journal of Economic Perspectives, 2017.

[5] 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. Springer, Cham, pp 127–138

[6] De S, Sohan FY, Mukherjee A (2018),”Attending sentences to detect satirical fake news”.In: Proceedings of the 27th international conference on computational linguistics, pp 3371–3380