A Novel Stance based Sampling for Imbalanced Data

A Case Study on COVID-19 Healthcare News

Isha Agarwal, Dipti Rana, Aemie Jariwala, Sahil Bondre

Department of Computer Engineering, SVNIT, Surat, India

Abstract—**While the world is suffering from coronavirus pandemic (COVID-19), a parallel battle with Infodemic, the proliferation of fake news online is also taking place. The spread of fake news during this global pandemic COVID-19 has dangerous consequences. This is the driving force behind this study. Relying on incorrect information obtained from the internet or social media can be fatal. According to a World Health Organization survey, at least 800 people have lost their lives because of COVID-19 misinformation during this time, highlighting the accurate automated classification of fake news. However, the data at disposal for classification is imbalanced. The Internet has a vast repository of authentic healthcare news, whereas Fake News on COVID-19 healthcare is not abundant. This imbalance leads to incorrect classification. The paper studies alternative approaches to text sampling. In this paper, we propose a stance based sampling method for balancing news data. The disparity between the title and content of news items is utilized to sample data points selectively and rectify the imbalance. The key findings are that the proposed stance-based sampling strategies enhance categorisation task performance consistently for varying degrees of imbalance. The proposed techniques can better detect misleading news in the health care sector.**

Keywords—*Fake news; healthcare; sampling; stance; COVID-19; imbalance*

I. Introduction

More than half of the global population now owns a smartphone, has internet access, and uses social media. There has been a 13.2 per cent rise in social media users by 2020. During the COVID19 outbreak, there was a tremendous spread of fake news and misinformation on a multitude of health-related topics. The World Health Organization (WHO) coined the term "Infodemic" to characterize the spread of false information. This information apocalypse has deadly implications, which is why a system to identify misleading news is urgently needed. JS Brennen et al. identified the types of misinformation on COVID-19 [1].

Real news articles about health issues outweighed those that had been validated and labeled as fake, causing an imbalance in the news dataset. The most common solution to this problem is sampling to restore data balance. The two-class sampling problem for non-textual numeric data was explored and summarized by Japkowicz and Stephen in 2002 [2]. However, not much contribution has been made to textual data. This research uses stance to present a novel data sampling strategy for rebalancing the classes of news content in the healthcare sector (Fig. 1). In contrast to standard sampling strategies used to improve classification performance, the implications of stance-based classification for false news detection are examined.

The study begins by reviewing the necessary theoretical foundation and academic work in text preprocessing, feature extraction, stance identification, and textual sampling (Section II). Section III introduces a curated dataset for assessing the performance of the proposed algorithms. The training of a stance classifier, which is required for stance-based algorithms, is described in Section IV. The stance-based approaches are discussed in detail in Sections V and VI. The results of the algorithms are presented in Section VII, along with a comparative study of traditional approaches. Finally, a brief conclusion of the paper, along with the future scope of research, is laid out in the concluding section.
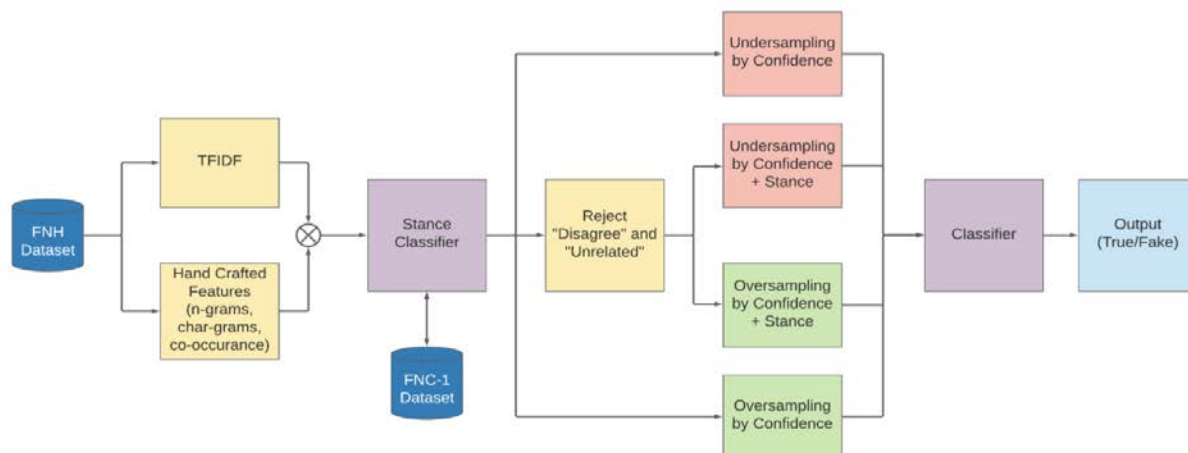

Fig. 1. Block Diagram for Sampling using Stance.

II. BACKGROUND AND RELATED WORK

During the COVID-19 epidemic, various traditional and deep learning techniques for fake news detection are being studied. For training the textual data, important features need to be extracted, and thus, the textual data needs to be first preprocessed, followed by feature extraction.

A. Textual Data Preprocessing

Within various studies and research, apart from tokenisation and stopword removal, authors have performed removal of HTTP URLs special characters [3][4][5]. In the study [6], the authors, in addition to the traditional preprocessing techniques, data augmentation using the back translation technique to increase the existing data is performed. The back-translation technique is the process in which the text is translated to its original language by converting it first into an intermediate language.

B. Feature Extraction

Along with preprocessing, the main task involves feature extraction, after which the model is trained using traditional or deep learning classifiers. Within feature extraction, various methods have been used, to name the popularly used include TF-IDF, GLoVe, and Pre-trained BERT. For TF-IDF, different kinds of features are tested, including uni-gram, bi-gram, character level, etc. The studies [7][8] used these different TF-IDF representations at word-level, n-grams, etc., before feeding them to the classifier and obtained excellent results. Various studies [6][9][10][11][12] applied TF-IDF to convert the textual data into vector space and extract the important features. These studies showed a significant detection of fake news with an accuracy of 80-95%.

The limitation associated with TF-IDF is that it takes into account the occurrence of a particular word and not its grammatical meaning. This is where word-representation such as GLoVE and BERT shine. Stanford developed a global vector for word representation, termed GLoVE [13]. Each word is represented in a meaningful vector space where the cosine distance between two words depicts their similarity. In the studies [14][15], the authors applied an embedding layer using 300-dimensional pre-trained glove vectors. This layer could convert the tweet texts into a meaningful vector space. Dharawat et al. [11] utilised a 100-dimensional pre-trained glove vector along with various classifiers, and similarly, other studies [16][4] employ the same dimension vector for the feature extraction process.

Google developed a pre-training NLP technique, termed BERT [17]. It is based on an understanding of the context and relationship by learning text representation in both directions. There are two main models of BERT - BERT Base and BERT Large and mBERT is the BERT representation for multilingual representation. In the study [18], pre-trained BERT embeddings and mBERT have been utilized to extract features from tweets. Hossain et al. [19] have utilized pre-trained BERT embedding for understanding the similarity between misconceptions and tweets. Cheng et al. [20] used the BERT embedding for converting rumor texts into vector form. After BERT, the LSTM-based variational autoencoder [21] is utilized to extract the important features. With this approach, a sufficient performance score was obtained. Various methods are utilized. However, these three embeddings are commonly used and help with providing efficient performance.

C. Stance Detection

Stance detection is the process of identifying the stance (related, unrelated, etc.) from the textual data. It is identified through understanding the similarity of the headline and body of news content or article [22]. Common approaches involve training a labeled dataset with their stances, but a challenging task in this area includes stance detection without having the target values or no training data.

Lillie et al. investigated the topic of false news identification and stance classification and published their findings [23]. Echo chambers and model organism issues are two examples of difficulties that make collecting high-quality data challenging. Several methods for stance classification and fake news detection have been explored, but it has been difficult to compare their results because of different data and measures. One specific approach is very appropriate and interesting for the thesis project, which is the use of a Hidden Markov Model (HMM) in analysing rumours in microblog data, achieving very promising results. Augenstein et al. experiment with conditional LSTM encoding to build a representation of tweet dependent on target [24]. An additional change includes augmenting the conditional encoding along with bidirectional encoding for stance detection.

D. Sampling Textual Data

Japkowicz et al. studied and unified all the previous approaches for solving the class imbalance problem using sampling and explained the nature of the problem by comparing the per-formance of the learning concept on parameters like complexity, training set size, and degree of imbalance [2]. A critical insight from the study was that class imbalance is not a problem because of the relative size of the small and large class, but it is only a problem when the size of its small class is too little for the complexity of the concept, i.e. when it contains minimal examples per subcluster. When each subcluster of the minority class contains many examples, accuracy remains high no matter the amount of imbalance or complexity of the concept. Textual data is a complex concept to learn, and the data distribution is sparse.

An active learning heuristic and representative sampling strategy is to read through the clustering structure of "uncertain" documents, reducing human effort in text classification tasks [25]. It also provides typical samples from which users can be polled to speed up SVM classifier convergence. This random sample includes more than one unlabelled document. Representative sampling was also compared to SVM active learning and random sampling by Zhao Xu et al. [25].

III. DATASET

For training the model, a curated dataset for fake news in the healthcare domain is required. Within this paper, the FNH dataset has been used. It consists of the following features -

Title, Content, URL, and Publishing Date. This hand-curated dataset has been created using web scraping techniques from various fake/satire and true labeled websites. The statistics for the dataset are presented in the table (Table I), where true news instances supersede the fake news instances, thus creating a high imbalance in the news ecosystem.

TABLE I.   FNH DATASET DISTRIBUTION

	Fake	True
Count	2424	7069

IV. TRAINING THE STANCE CLASSIFIER

For correcting the imbalance existing in the dataset, the stance approach has been chosen. Stance takes into account the textual similarity between the title and body content. Based on the similarity, we can gather its stance value and decide which instances of the particular class need to undergo sampling. This approach provides better insights on choosing instances to undergo sampling than the random traditional approach.

However, the FNH dataset has no stance labelled attribute. Introduction of the stance and its confidence for each instance of the dataset, a stance-labelled dataset is trained. In this paper, the FNC-1 dataset is used as the stance-labelled dataset. The training set is the entire FNC-1 dataset, and the testing set is the FNH dataset. A classifier works with the numerical data, and thus the textual data is represented in vector form.

Algorithm 1: Training Stance Classifier on FNC-1

Input: An annotated list of documents from FNC-1 dataset
Output: Classifier trained to identify stance between title and content of news article
Begin:
1. Create a vocabulary of words V from all text data
2. X := { }
3. N = size(FNC-1)
4. For every document di in FNC-1, i := [1,2,3,...,N]:
   a. ti := di.title, ci := di.content
   b. V_{ti} := TF-IDF(V, ti), V_{ci} := TF-IDF(V, ci)
   c. P_i := co_occurance(ti, ci)
   d. Q_i := n_grams(ti, ci, n)
   e. R_i := char_grams(ti, ci, n)
   f. S_i := word_overlap(ti, ci)
   g. w_ti := word_embedding(ti), w_ci := word_embedding(ci)
   h. T_i := cosine_similarity(w_ti, w_ci)
   i. X_i := [V_ti, V_ci, P_i, Q_i, R_i, S_i, T_i]
   j. append(X, X_i)
5. Y := FNC1.stance_labels
6. Create Instance of Multinomial Naive Bayes Classifier: MNB
7. MNB.train(X, y)
8. Return MNB

End

Along with using TF-IDF (or word vectorization method), a hand-crafted vector space is created to emphasize the correlation between the headline and body of each document in a vectorized format. The hand-crafted vector space is a 28-dimensional vector space, and the distribution is explained in the table (Table II).

TABLE II.   DISTRIBUTION OF VECTOR SPACE IN HAND-CRAFTED FEATURES

Feature	No. of vectors
Binary co-occurrence	2
Binary co-occurrence (stopwords removal)	2
N-grams	3
Char-grams	3
Count-grams	22
Word-overlap	1
Word-embedding	1

The TF-IDF vectors of the headline and the body, each 100 size vector are concatenated along with the handcrafted vectors is concatenated to give a 228 vector space for each document.

Within the FNC-1 dataset, there are five classes, and thus, Multinomial Naive Bayes takes into account of Bayes Theorem and provides the probability for the different classes for a single instance. Thus, the 228 vector space is subjected to a Multinomial Naive Bayes classifier to create the final trained model. The final trained model is then used for predicting the stance and confidence for the FNH dataset.

Algorithm 2: Generating Stance Values for FNH

Input: List of documents from FNH Dataset and an instance of MNB classifier from Algorithm 2
Output: Stance values for each document in the dataset
Begin:
1. X := { }
2. N := size(FNH)
3. For every document di in FNH, i := [1,2,3,...,N]:
   a. ti := di.title, ci := di.content
   b. V_ti := TF-IDF(V, ti), V_ci := TF-IDF(V, ci)
   c. P_i := co_occurance(ti, ci)
   d. Q_i := n_grams(ti, ci, n)
   e. R_i := char_grams(ti, ci, n)
   f. S_i := word_overlap(ti, ci)
   g. w_ti := word_embedding(ti), w_ci = word_embedding(ci)
   h. T_i := cosine_similarity(w_ti, w_ci)
   i. X_i := [V_ti, V_ci, P_i, Q_i, R_i, S_i, T_i]
   j. append(X,X_i)
4. Stance_Labels := MNB.predict(X)
5. return Stance_Labels

End

V. UNDERSAMPLING USING STANCE

For balancing the classes in undersampling, the instances of the majority classes are deleted till it is equal to the instances of minority classes. The deletion of the instances can be random, which is a traditional yet inefficient approach. Deleting the instances based on a systematic algorithm is an efficient approach.

In the previous section, the algorithm to acquire the stance label and confidence for each document is presented. The principle followed for undersampling is that the documents associated with low confidence should be subjected to deletion, which further resolves the imbalance.

In algorithm 3, the majority (true) class is sorted in descending order based on the confidence attribute, and the first N attributes are taken into consideration where N is equal to the number of instances belonging to the minority (fake) class.

Algorithm 3: Undersampling based on Confidence

Input: List of documents from FNH Dataset labeled with stance
Output: Undersampled data for classification task
Begin

1. D := {FNH, Stance_Labels}
2. True_News := D.filter(d where d.label = "True")
3. Fake_News :=D.filter(d where d.label ="Fake")
4. True_News :=sort(True_News, True_News.stance.confidence, reverse=true)
5. N := size(Fake_News)
6. Sampled_True_News := pick_n(True_News, N)
7. Sampled_Data := {Sampled_True_News, Fake_News}
8. return Sampled_Data

End

Along with the confidence attribute, the stance attribute has also been introduced within the FNH dataset. As the undersampling has been performed on the majority (true) class, the removal of stance attributes that are labeled as "disagree" or "unrelated" needs to be performed. The reason being that if a particular document is labeled true, then the headline and body needs to belong to the "agree" or "discuss" stance.

In Algorithm 4, after the deletion based on the stance values, sorting has been performed on the majority (true) class based on the confidence in the descending order. The first N instances are chosen where N is equal to the number of instances of the minority (fake) class.

Algorithm 4: Undersampling based on Stance and Confidence

Input: List of documents from Fake News Dataset labeled with stance
Output: Undersampled data for classification task
Begin

1. D := {FNH, Stance_Labels}
2. True_News := FNH.filter(d where d.label = "True" and d.stance.label != ("Disagree" or "Unrelated"))
3. Fake_News := FNH.filter(d where d.label = "Fake")
4. True_News := sort(True_News, True_News.stance.confidence, reverse=true)
5. N := size(Fake_News)
6. Sampled_True_News := pick_n(True_News, N)
7. Sampled_Data := {Sampled_True_News, Fake_News}
8. return Sampled_Data

End

The accuracy of the minority class is heavily weighted in evaluating performance since, in an unbalanced dataset, the minority class accuracy must improve. Thus, undersampling with stance variations is compared to the baseline, which was the original imbalance data, as well as the traditional undersampling approach.

From the graph (Fig. 2), it can be concluded that both approaches utilizing undersampling using stance supersede the traditional undersampling method and the baseline in performance as seen. Undersampling using stance and confidence performs the best as the imbalance ratio increases. This showcases that randomly choosing instances to undergo undersampling is an inefficient approach compared to utilizing the stance and confidence associated with each document.
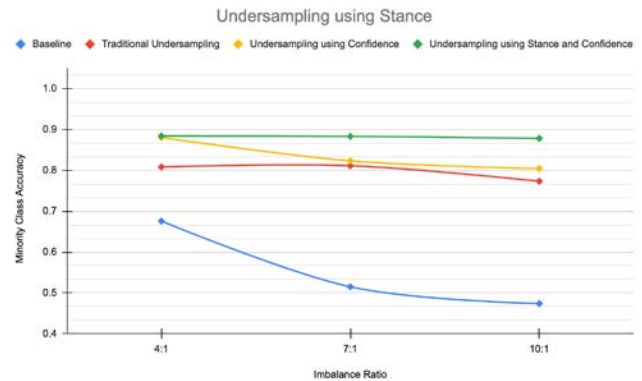

Fig. 2.   Evaluation of Undersampling using Stance.

VI. OVERSAMPLING USING STANCE

To balance the classes in oversampling, the minority class instances are oversampled until they are equal to the majority class instances. The oversampling method involves selecting a subset of minority class instances. These subsets are duplicated in a method that when these oversampled instances are added to the original minority instances, they equal instances of the majority class. The direct duplication could lead to overfitting, and hence it is important to choose an optimal number of subsets that undergo duplication to avoid overfitting.

Oversampling using stance uses the same base principle utilized in undersampling using stance. In Algorithm 5, first, sorting the minority (fake) class instances in descending order based on the confidence is performed.

The k integer which decides the subsets which will undergo oversampling is chosen in a way to avoid overfitting. In the case of the FNH dataset, the k chosen is 100 to keep the direct duplication of the subjects under 100. Choosing k within the range of [10, 50] requires the direct duplication of the subset to be done more than 150 times to resolve the imbalance. This leads to the overfitting of the data. However, choosing the k value to be greater than 100 will increase the time taken as larger subsets are chosen to undergo oversampling.

Once the top k instances are chosen from the minority class, they are subjected to oversampling such that the number of instances of both majority and minority classes is equal.

Algorithm 5: Oversampling based on Confidence

Input: List of documents from FNH Dataset labelled with stance and integer k

Output: Oversampled data for classification task

Begin

1. D := {FNH, Stance_Labels}
2. True_News := D.filter(d where d.label = "True")
3. Fake_News :=D.filter(d where d.label = "Fake")
4. Fake_News := sort(Fake_News, True_News.stance.confidence, reverse=true)
5. Sampling_Examples := pick_n(Fake_News, k)
6. Sampled_Fake_News := Fake_News + oversample(Sampling_Examples)
7. Sample_data := {True_News, Sampled_Fake_News }
8. return Sample_data

End

For oversampling using stance and confidence, the same principle utilized for undersampling using stance and confidence has been used. After rejecting based on the stance value and sorting the minority class instances in the descending order based on their confidence, the first k subsets are chosen, which undergo oversampling (Algorithm 6). The method to choose the value of k has been explained in oversampling using stance.

Algorithm 6: Oversampling based on Stance and Confidence

Input: List of documents from FNH Dataset labeled with stance and integer k
Output: Oversampled data for classification task
Begin

1. D := {FNH, Stance_Labels}
2. True_News := D.filter(d where d.label = "True")
3. True_News := FNH.filter(d where d.label = "Fake" and d.stance.label != ("Disagree" or "Unrelated"))
4. Fake_News := sort(Fake_News, True_News.stance.confidence, reverse=true)
5. Sampling_Examples := pick_n(Fake_News, k)
6. Sampled_Fake_News := Fake_News + oversample(Sampling_Examples)
7. Sample_data := {True_News, Sampled_Fake_News }
8. return Sample_data

End

Oversampling with stance variations is compared to the baseline, which was the original imbalance data, as well as the traditional oversampling approach, and the priority is provided to the accuracy of the minority class. The reason is that within imbalanced data, the model tends to overfit, and the performance of the minority class is low.

Oversampling using stance supersedes the traditional oversampling method and the baseline in performance, as seen in Fig. 3. Oversampling using stance and confidence increases its performance and has a steady increase in the accuracy of the minority class across all imbalance ratios, while the traditional oversampling method scores reduce as it reaches a high imbalance ratio.
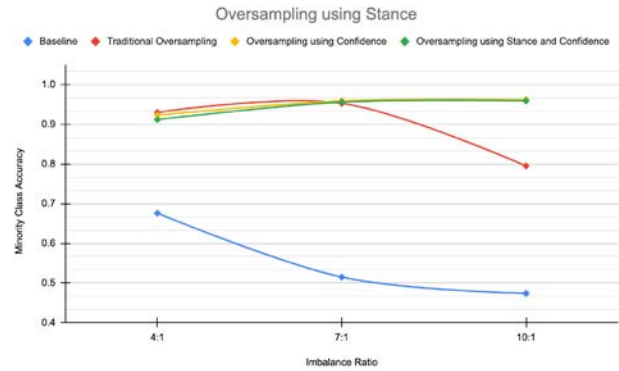

Fig. 3. Evaluation of Oversampling using Stance.

VII. RESULTS

For the evaluation purpose, the MCC score is taken into account. MCC is the only evaluation metric that considers all four quadrants of a confusion matrix, whereas Accuracy and Precision skew toward the positive class. To understand whether the model is overfitting by having high accuracy for the majority and low accuracy for the minority class or the model is balanced effectively, the accuracy of majority and minority class is both taken into account.

The evaluation metric is based on the confusion matrix and the MCC score. This has been done by averaging five trials for each method. The tables (Tables III to V) provide the performance of sampling using stances against the traditional sampling methods for the imbalance ratios 4:1, 7:1, 10:1.

For 4:1 imbalance ratio, it can be observed that oversampling methods supersede the undersampling methods in performance. Within the undersampling methods, the stance methods exceed in performance compared to the traditional method by a huge margin.

TABLE III. SIMULATION OF RESULT FOR RATIO 4:1

Stance	Accuracy Majority Class	Accuracy Minority Class	MCC
Baseline	0.949	0.677	0.659
Traditional Random Undersampling	0.884	0.801	0.685
Traditional Random Oversampling	0.908	0.932	0.841
Undersampling: Confidence	0.785	0.876	0.665
Undersampling: Stance & Confidence	0.872	0.889	0.76
Oversampling: Confidence	0.921	0.919	0.84
Oversampling: Stance & Confidence	0.925	0.918	0.843

Within the oversampling methods, the traditional oversampling method shows a minor improvement in performance compared to the oversampling using stance methods.

TABLE IV. SIMULATION OF RESULT FOR RATIO 7:1

Stance	Accuracy Majority Class	Accuracy Minority Class	MCC
Baseline	0.978	0.495	0.569
Traditional Random Undersampling	0.842	0.814	0.655
Traditional Random Oversampling	0.935	0.953	0.889
Undersampling: Confidence	0.835	0.817	0.653
Undersampling: Stance & Confidence	0.883	0.894	0.777
Oversampling: Confidence	0.947	0.959	0.907
Oversampling: Stance & Confidence	0.948	0.955	0.905

For 7:1 ratio, it follows the similar pattern as 4:1 ratio where oversampling methods supersede the undersampling methods in performance. Within undersampling methods, the undersampling using confidence showed a significant drop while undersampling using stance and confidence supersede in performance by a huge margin.

Within oversampling methods, the difference in performance for oversampling using stance and traditional method is very less. This showcases that with an increase in imbalance ratio, the oversampling using stance shows improvement in their performance.

For 10:1 ratio, both of the oversampling using stance variants supersedes in performance while the performance of traditional oversampling method reduces significantly. Within the undersampling methods, the undersampling using stance and confidence showcases steady improvement in performance with an increase in the imbalance ratio.

TABLE V. SIMULATION OF RESULT FOR RATIO 10:1

Stance	Accuracy Majority Class	Accuracy Minority Class	MCC
Baseline	0.983	0.466	0.558
Traditional Random Undersampling	0.833	0.793	0.625
Traditional Random Oversampling	0.955	0.863	0.840
Undersampling: Confidence	0.798	0.818	0.616
Undersampling: Stance & Confidence	0.872	0.886	0.758
Oversampling: Confidence	0.949	0.958	0.908
Oversampling: Stance & Confidence	0.953	0.959	0.914

VIII. CONCLUSION AND FUTURE WORK

People are led to believe false facts about various health advice and medical treatments as a result of fake news. This creates a pressing need for accurate detection of fake news in healthcare. The proposed framework focuses on improving the performance of fake news detection in order to address these issues. Because the number of true articles in this work outnumbers the number of fake articles, stance has been used for the text sampling method, both undersampling and oversampling.

Understanding the relationship between the headline and the article's content is essential in stance classification. The FNH dataset has been trained to obtain their respective stance labels and confidence. Two approaches have been proposed. Stance-based undersampling and stance-based oversampling were carried out using these variations. These proposed approaches demonstrated a significant improvement in overall detection performance when implemented with various imbalance ratios compared to traditional methods.

Apart from increasing the performance using stance by resolving balance, the broader implications of the paper also highlight the unique method of converting the textual data into vector space highlighting the similarity between the title and body content of the document which further was utilized grabbing the stance and confidence attribute for each document.

Future work can be extended by training different classifiers for stance detection. Experiments can also be further carried out considering the tuning of configuration parameters for the rate of sampling, etc.

REFERENCES

[1] Brennen JS, Simon F, Howard PN, Nielsen RK. Types, sources, and claims of COVID-19 misinformation. Reuters Institute. 2020 Apr 7;7(3):1.

[2] Japkowicz N, Stephen S. The class imbalance problem: A systematic study. Intelligent data analysis. 2002 Jan 1;6(5):429-49.

[3] R.Kaliyar,A.Goswami,P.Narang, "A Hybrid Model for Effective Fake News Detection with a Novel COVID-19 Dataset".

[4] Wani, I. Joshi, S. Khandve, V. Wagh, R. Joshi, "Evaluating Deep Learning Approaches for Covid19 Fake News Detection".

[5] Chen, B. Chen, D. Gao, Q. Chen, C. Huo, X. Meng, W. Ren, Y. Zhou , "Transformer-based Language Model Fine-tuning Methods for COVID-19 Fake News Detection".

[6] J.Ayoub,X.Yang,F.Zhou, "Combat COVID-19 Infodemic Using Explainable Natural Language Processing Models".

[7] M. K. Elhadad, K. F. Li, F. Gebali, "Detecting Misleading Information on COVID-19".

[8] Mahlous, A. Al-Laith , "Fake News Detection in Arabic Tweets during the COVID-19 Pandemic".

[9] A.Koirala , "COVID-19 Fake News Classification using Deep Learning".

[10] P.Patwa,S.Sharma, S.PYKL,V.Guptha, G.Kumari, M.S.Akhtar,A.Ekbal, A.Das,T.Chakraborty, "Fighting an Infodemic: COVID-19 Fake News Dataset".

[11] A.Dharawat,I.Lourentzou,A.Morales,C.Zhai, "Drink bleach or do what now? Covid-HeRA: A dataset for risk-informed health decision making in the presence of COVID19 misinformation".

[12] L. Alsudias, P. Rayson , "COVID-19 and Arabic Twitter: How can Arab World Governments and Public Health Organizations Learn from Social Media?"

[13] J.Pennington, R.Socher, C.Manning , "GloVe: Global Vectors for Word Representation".

[14] M.Elhadad, K.Li, F.Gebali, "An Ensemble Deep Learning Technique to Detect COVID-19 Misleading Information".

[15] S.Kumar, K.M.Carley, "A Fine-Grained Analysis of Misinformation in COVID-19 Tweets".

[16] L.Cui,D.Lee , "CoAID: COVID-19 Healthcare Misinformation Dataset".

[17] J.Devlin, M.W.Chang, K.Lee, K.Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding".

[18] Kar, M. Bhardwaj, S. Samanta, A. P. Azad, "No Rumours Please! A Multi-Indic-Lingual Approach for COVID Fake-Tweet Detection".

[19] T.Hossain, "COVIDLIES: Detecting COVID-19 Misinformation on Social Media".

[20] M. Cheng, S. Wang, X. Yan, T. Yang, W. Wang, Z. Huang, X. Xiao, S. Nazarian, P. Bogdan, "A COVID-19 Rumor Dataset".

[21] M. Cheng, S. Nazarian, P. Bogdan, "VRoC: Variational Autoencoder-aided Multi-task Rumor Classifier Based on Text".

[22] Rajendran, B. Chitturi, P. Poornachandran, "Stance-In-Depth Deep Neural Approach to Stance Classification".

[23] Anders Edelbo Lillie and Emil Refsgaard Middelboe. "Fake news detection using stance classification: A survey". In:arXiv preprint arXiv:1907.00181(2019).

[24] Augenstein, T. Rocktäschel, A. Vlachos, K. Bontcheva , "Stance Detection with Bidirectional Conditional Encoding".

[25] Zhao Xu et al. "Representative sampling for text classification using support vector ma-chines". In:European conference on information retrieval. Springer. 2003, pp. 393–407.