I.J. Information Engineering and Electronic Business, **2019, 4, 1-10**
Published Online July 2019 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijieeb.2019.04.01

Natural Language Processing based Hybrid Model for Detecting Fake News Using Content-Based Features and Social Features

Shubham Bauskar, Vijay Badole, Prajal Jain, Meenu Chawla
Department of Computer Science and Engineering, Maulana Azad National Institute of Technology,
Bhopal, 462003, India
Email: shubhamcbauskar@gmail.com, victormanitcse@gmail.com, pjiit2015@gmail.com, chawlam@manit.ac.in

Received: 24 January 2019; Accepted: 22 April 2019; Published: 08 July 2019

Abstract—Internet acts as the best medium for proliferation and diffusion of fake news. Information quality on the internet is a very important issue, but web-scale data hinders the expert's ability to correct much of the inaccurate content or fake content present over these platforms. Thus, a new system of safeguard is needed. Traditional Fake news detection systems are based on content-based features (i.e. analyzing the content of the news) of the news whereas most recent models focus on the social features of news (i.e. how the news is diffused in the network). This paper aims to build a novel machine learning model based on Natural Language Processing (NLP) techniques for the detection of 'fake news' by using both content-based features and social features of news. The proposed model has shown remarkable results and has achieved an average accuracy of 90.62% with F1 Score of 90.33% on a standard dataset.

Index Terms—Fake News Detection, Machine Learning Classifier, Natural Language Processing, Probabilistic Classifiers.

I. Introduction

Fake news refers to false information masqueraded as authentic news. Fake news is mainly of the following types: satirical news, hoax, completely fabricated news and government propaganda (political). It is typically distributed to attract viewers and to generate advertising revenue. The reasons behind fake news include media manipulation and propaganda, political and social influence, provocation and social unrest and financial profit [9]. However, people and groups with potentially malicious agendas have been known to initiate fake news in order to influence events and policies around the world. [10] showed the significant impact of fake news in the context of the 2016 US presidential elections. [16] conducted a survey on how fake news is spread on social media websites like "Facebook" by analyzing the activities of respondents' social media accounts. In 2013, The World Economic Forum warned that the so-called 'digital wildfires', that is, unreliable information going viral online (aka fake news) would be one of the biggest threats faced by society. Given its negative impacts (e.g. It brings needless agitations and social unrest in the society) [14,15], detection of fake news has become an increasingly important issue [11].

This paper presents a novel fake news detection model which uses both content-based and social features for fake news detection. The proposed model has outperformed existing approaches in the literature and obtained higher accuracies than traditional content-based methods on a publically available standard dataset that was recently published [2].

II. Related Works

Extensive research has been done in order to develop an accurate and reliable automatic fake news detector. Traditionally content-based approach has proven effective for the news lacking social information. In [5] authors showed that a simple model based on term frequency-inverse document frequency (tf-idf) offers a baseline accuracy of 88.5%. In [6] authors have used tf-idf with six different machine learning classifiers on a 2000 news dataset, obtaining a 92% accuracy. In [7] authors used syntactic and semantic features of news articles for classifying between genuine and fake news articles using a trigram language model, obtaining an accuracy of 91.5%. But the main problems associated with content-based methods for real-world fake news detection is that these news articles are intentionally written in a style that makes it impervious to word analysis.

Research has been conducted for developing fake news detector using social features of news. In [8] authors showed that Facebook posts can be classified with high accuracy as hoaxes or non-hoaxes on the basis of the users who "like" or "dislike" them and achieved accuracies exceeding 99% even with very small training dataset, authors have used logistic regression and harmonic boolean label crowd-sourcing methods. Although the method proposed by [8] offers very high accuracy, its application is limited just to the use-cases that garner enough social media attention (i.e. the number of likes and dislikes). In [11] authors have proposed a Multi-source Multi-class Fake news Detection (MMFD) framework and also introduced an automated and interpretable way to integrate information from multiple sources, but the accuracy of the model on real-world data is only 38.81%. Similar research has been carried out by authors in [17], they have built an automated system called "FakeNewsTracker" for understanding and detection of fake news. This system collects news contents and social context automatically. In [4] authors have used content-based methods only when the social based methods perform poorly. They have built the model based entirely on only one type of feature at a time and tested this model on real-world data and have obtained an accuracy of 81.7%. Our model is substantially different from the model proposed in [4], instead of relying on a single type of feature of news articles (i.e. either content-based or social features) for prediction of fake news, our model uses both content-based and social features of news articles simultaneously for detection of hoax articles.

III. Methodology

The flowchart of the proposed methodology of the entire process for determining whether the news article is fake or genuine using the content-based and social features of the news article is presented in Fig. 1. This flowchart captures the major part of the developed algorithm and each segment of the flowchart is described in later sections.

A. Data Preprocessing

Data preprocessing consists of various techniques which can be used to convert the data present in raw format (data gathered from various resources which are not feasible for the analysis e.g. news grabbers extract news from the web page of news editors and store the various features of that news, this data need further processing before feeding it to classification algorithms) to clean format (specific formatted data as required by a classification algorithm). Preprocessing refers to the various transformations that are applied to the data before feeding it to the classification algorithm. Data preprocessing increases the efficiency of machine learning algorithms as some of the algorithms require the data to be in a specified format. Data Preprocessing is needed only on the content-based part of the news (*heading, body,* etc.). Social features of news don't require any preprocessing. Each of these content-based parts of news present in the dataset is modeled using the *Bag of Words Model* which is a simplified representation used in Natural Language Processing (NLP). In this model, a text (such as a document or a sentence) is represented as a bag (multiset) of its words, by disregarding the stop words and the ordering of words in the text, but the multiplicity of each word is stored.

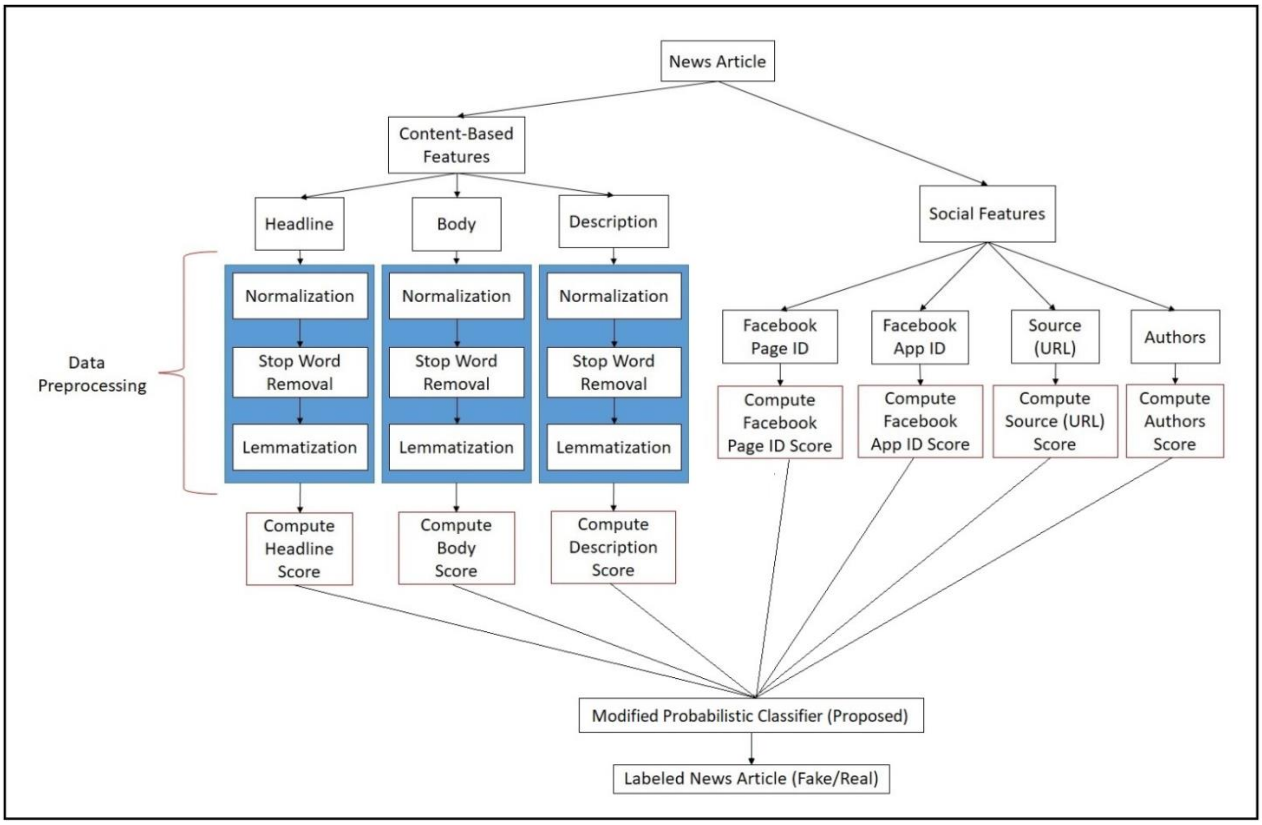

Fig.1. Flowchart of Proposed Methodology

A.1. Normalization

Normalization phase includes various preprocessing steps that are needed to be performed on each word, these steps are described in detail in the subsequent section.

1. Changing Uppercase to Lowercase

This phase of data preprocessing focuses on the *case-folding* operation. *Case-Folding* is a technique to reduce all the letters of the word to lowercase which allows case-insensitive comparisons, all the textual content of the news (present in the corpus, and the news that is to be labeled as fake or real) that is in Uppercase form is converted to Lowercase form. This operation is necessary because all the text present in the news corpus should have a common representation format.

If the two words that possess same meaning and are used in the same context, but they differ in the format of representation, then the matching algorithm would not be able to match both the words and will treat them as a separate entity. e.g. *"Automobile"* and *"automobile"* will be treated as two different words because they differ in their format of representation, which would be incorrect. Thus by reducing *"Automobile"* to lower case "*automobile*" the efficiency of the matching algorithm can be increased.

The main disadvantage associated with *Case-Folding* operation is that many proper nouns are derived from common nouns and can only be distinguished using cases. e.g. *"General Motors"* and *"general motors"* the first word refer to the company and it should be treated differently from the second word, but after applying case-folding operation both words will be in the same format and will be treated as same. Thus case-folding operation leads to loss of information about the proper noun.

2. Removing Special Characters

Special characters as such don't possess any significance during content-based matching and should be omitted before applying the classification algorithm. Although by removing special characters for e.g. "*$*" some context-based information can be lost but there is no loss of content-based information. e.g. let's assume a news A contains a statement "*cost of a book is $100*" now here if the special character i.e. "$" is removed due to preprocessing. The information about the currency will be lost and the statement after preprocessing "*cost of an item is 100*" now may be referring to dollar or rupee or any other currency.

3. Date Format Normalization

Since dates that are mentioned in the news may serve as the most important factor that can further improve the efficiency of the classification model. By using this information we can check whether the news is a derivation of the original news or fabrication of it. E.g. Date can occur in various formats in the news such as dd/mm/yy, mm/dd/yyyy, dd/mm/yyyy. All these forms should be converted to a standard form such that two dates that are mentioned in news and refer to the same day, then these dates should be matched with each other during content-based matching.

e.g. Let's assume that 31.10.18 was mentioned in news A and 10.31.2018 was mentioned in news B, if these date formats are not normalized then during content-based matching these dates will mismatch, but if they are normalized to a standard form that will be used throughout the dataset i.e. convert 10.31.2018 (mm.dd.yyyy) to 31.10.18 (dd.mm.yy) then both these dates will match during content-based classification.

Apart from the above mention transformation, there are many other forms of normalization that can be performed on the content portion of the news such as *Unit Normalization,* etc. This paper only focuses on the above mention transformations for normalization as other preprocessing transformations are not providing significant results.

A.2. Stop Word Removal

Stop words are the most common words in a language (e.g. *a, an, the, was, in,* etc.). These words don't possess any discriminating power and need to be discarded before constructing the bag of words model, because stop words take up more space and increase the processing time, such words do not possess much relevance. The word having higher relevance possess high local frequency (number of time word occurs in the given document that is to be classified) and low document frequency (number of documents in the corpus containing that word). This paper focuses on removing the stop word by calculating its relevancy using the term frequency-inverse document frequency (tf-idf).

Tf-idf is a numerical statistical method that is used to describe how important a particular word is to a document in a collection. For a term *t* present in document *d*, the tf-idf score of that term in that document is given by $tf\text{-}idf_{t,d}$ as shown in equation (1).

$$tf - idf_{t,d} = (1 + \log tf_{t,d}) * \log(\frac{N}{df_t}) \qquad (1)$$

Where, $tf_{t,d}$ = Term Frequency of term *t* in document *d*.

N = Number of the document in the corpus.

df_t = Document Frequency (i.e. Number of Documents in the corpus containing the term *t*).

The inverse of document frequency of the stop word is very low as it occurs in mostly all documents of the corpus, once these stop words are identified they are discarded before modeling content-based features using *Bag of Words Model*.

A.3. Lemmatization

For grammatical reasons, various news editors may use different forms of a word (such as run, ran, running). In addition to this, there is a family of derivationally related words (Terms in different syntactic categories that have the same root and are semantically related) e.g.

photograph, *photography*, *photographic*. Lemmatization is a linguistic transformation, it is the process to group all the inflectional forms of a word so that all of them can be analyzed using a single item and sometimes derivationally related forms of a word to a common base form by removing inflectional endings only and returning the base or dictionary form of a word, which is known as the lemma. This technique uses the vocabulary and the morphological analysis of the word to group all inflected forms. If two sentences only differ in the use of the inflectional forms of the word then these sentences should match with each other but if lemmatization is not performed then the classification model will mismatch such type of sentences.

E.g. *John runs*, *John is running*, *John ran*. Although all these sentences are describing the action *john* does and so all the sentences should match syntactically but due to the presence of more than one inflectional form of the word matching algorithm fails to group them as one. So in order to improve the matching of words that are syntactically same, there is a need to convert all the inflectional form of the word to a common base form that can represent all of them. After lemmatization all these sentences maps to "*John run*", because of all the inflectional form of the word i.e., *run*, *ran*, *running* will now map to a common base *run*. The main disadvantage of lemmatization is the loss of timing information as different inflectional forms of word convey different timing information such as *ran* refers to past, *running* refers to the present. But contextual information is not much relevant as compared to the content-based information for this problem domain.

B. Proposed Model

Given a news article *d* and a set of previously published news articles in the corpus *C* which consists of news that is either labeled as *Real* or *Fake*. The goal of Fake News Detection is to predict whether the *d* is fake or not using a supervised machine learning classifier *M* which is trained over dataset consisting of news articles from *C*.

$$\text{M (d, C)} = \begin{cases} 1, & Real \\ 0, & Fake \end{cases}$$

The goal of this paper is to build a machine learning classification model *M* which will classify the given news documents into a fake and real class. Hence, this problem can be mapped into a binary classification problem and can be solved using a *Modified Probabilistic Classifier*, which incorporates both content-based features and social features of the news article for predicting whether the news is fake or not. *M* uses *7* features f_1, f_2,..., f_7 (i.e. *Headline, Body, Description, Facebook PageID, Facebook AppID, Source, Authors*) of the news articles for demarcation and these features are discussed in detail in Section (IV).

Let us Assume that there is a news article *d* which has to be classified as either fake or real and a machine learning model *M* which will calculate the class scores of *d* with respect to both *Fake* and *Real* class.

The score of *d* for *Fake* class is calculated using equations (4), (10), (12), (14), (16), (18), and (20) and is given by equation (2)

$$Score(d, Fake) = \sum_{i=1}^{7} FeatureScore(f_i, Fake) \quad (2)$$

Where, *FeatureScore*(f_i, *Fake*) is the score of the feature f_i in *d* with respect to the *Fake* class.

The score of *d* for the *Real* class is calculated using equations (7), (11), (13), (15), (17), (19), and (21) and is given by equation (3)

$$Score(d, Real) = \sum_{i=1}^{7} FeatureScore(f_i, Real) \quad (3)$$

Where *FeatureScore*(f_i, *Real*) is the score of the feature f_i in *d* with respect to the *Real* class.

Based on the class scores of document *d* as calculated using equation (2), and (3). *M* will predict the Class label for *d* by using Algorithm 1:-

Algorithm 1: AssignClassLabel( )

Input: Score(d, Real), Score(d, Fake)

Output: ClassLabel

```
ClassLabel ← NULL
if Score(d, Real) >= Score(d, Fake) then
    ClassLabel ← Real
else
    ClassLabel ← Fake
end if
return ClassLabel
```

IV. Mathematical Background Behind The Proposed Model

This section describes the analysis that has been carried out to understand the demarcation of the fake news articles from the genuine news articles, Later in this section, various content-based and social-based features are discussed which best discriminates the fake news articles from genuine news articles. This section also contains the mathematical background of how the proposed model uses these features for identifying the fake news.

A. Content-Based Features

Content-based features of fake news consist of a *headline* and *body*. After preprocessing these features are modeled using the *Bag of Words Model*. These features mostly emphasize on the content portion and don't incorporate any contextual meaning.

A.1. Headline

Headlines are independent of the body of the news. Headlines are the simple representation of the complex text and they are used to capture the summary of the underlying text. They act as a medium to communicate the gist of the news. Headlines play a key role in gaining the attention of potential readers. Headlines of the fake news are generally eye-catching, sensational and exaggerated. Some headlines try to attract the attention of the reader by saying imaginary things e.g. *"This is what your fingerprint says about your destiny"*. Some headlines try to scare or horrify the reader e.g. *"Patient was declared dead but a few minutes later rose from bed in XYZ hospital"*. Some headlines try to create awareness about health risks e.g. *"You drink it every day without knowing that it can cause cancer"*. Some headlines provoke the reader to share the news article e.g. *"95% discount on shoes - share the article to avail your discount"*.

In most of the cases, the only aim of the headline (that belongs to fake article) is to grab the attention of the reader and this types of headlines are loosely connected to the underlying portion of the news article. Thus by examining the headlines of the news articles, its authenticity can be verified.

Let us assume that headline *h* consists of *n* words w_1, w_2, ..., w_n. Then the score of feature *headline* for *Fake* class is given by equation (4).

$$FeatureScore(h, Fake) = \sum_{i=1}^{n} WordScore(w_i, Fake) \quad (4)$$

Where *WordScore*(w_i, *Fake*) is the score of the word $w_i$with respect to *Fake* class. This can be calculated using equation (5).

$$WordScore(w_i, Fake) = \frac{Count(w_i,d)}{|d|} * GW\ (w_i, C_f) \quad (5)$$

Where,

C_f is the set of news articles in *C* that are labeled as *Fake*, *Count*(w_i, *d*) is the number of occurrences of the word w_i in d, Further, *GW*(w_i, C_f) is then the global weight of the word w_i with respect to C_f given by the equation (6).

$$GW\ \left(w_i, C_f\right) = \frac{1+DocumentCount\left(w_i,C_f\right)}{|C_f|} \quad (6)$$

Where, *DocumentCount*(w_i,C_f) is the number of documents in C_f containing the word w_i and $|C_f|$ is the cardinality of the set C_f.

Similarly, the score of the feature *headline* for the *Real* class is given by equation (7).

$$FeatureScore(h, Real) = \sum_{i=1}^{n} WordScore(w_i, Real) \quad (7)$$

Where *WordScore*(w_i, *Real*) is the score of the word $w_i$with respect to the *Real* class. This can be calculated using equation (8).

$$WordScore(w_i, Real) = \frac{Count(w_i,d)}{|d|} * GW\ (w_i, C_r) \quad (8)$$

Where, $C_r$is the set of news articles in corpus C that are labeled as *Real*, *Count*(w_i, *d*) is the number of occurrences of the word w_i in d, Further, *GW*(w_i, C_r) is the global weight of the word $w_i$with respect to C_r given by the equation (9).

$$GW\ (w_i, C_r) = \frac{1+DocumentCount(w_i,C_r)}{|C_r|} \quad (9)$$

Where, *DocumentCount*(w_i, C_r) is the number of documents in C_r containing the word w_i and $|C_r|$ is the cardinality of the set C_r.

A.2. Body

Study of thousands of news articles reveals a *stylistic* difference between genuine and a fake article. Genuine news articles contain more language conveying differentiation whereas fake news articles are expressed with more certainty. Words that are more likely to be used in genuine articles are the words that convey differentiation or express insight or that quantify for e.g. *think, know, consider, not, without, but, instead, against*, whereas the words that are more likely to be used in fake news articles are the words that convey certainty or that express positive emotion or that focus on future for e.g. *always, never, proven, pretty, good, cause, know, ought, gonna, soon*.

Thus by examining the content of the news article, its authenticity can be proved. Let us assume that the Body portion *b* of the news articles consists of *n* words. Then the score of feature *body* for *Fake* class is given by equation (10).

$$FeatureScore(b, Fake) = \sum_{i=1}^{n} WordScore(w_i, Fake) \quad (10)$$

Where *WordScore*(w_i, *Fake*) can be calculated by using the equation (5).

Similarly, the score of the feature *body* for the *Real* class is given by equation (11).

$$FeatureScore(b, Real) = \sum_{i=1}^{n} WordScore(w_i, Real) \quad (11)$$

Where *WordScore*(w_i, *Real*) can be calculated by using the equation (8).

A.3. Description

The description is the gist of the content of the news article. It helps to identify whether the news article is biased towards a particular point of view. Analysis of thousands of Fake news articles that were published reveals that these articles are biased towards a particular topic and in most of the cases, these articles won't reveal the full story. Fake news articles try to play with the emotions of the potential readers and make the reader angry, happy or scared. Thus by analyzing the description or the gist of the news articles its authenticity can be checked.

Let us assume that the description or gist *g* of the news articles consists of *n* words. Then the score of feature *description* for *Fake* class is given by equation (12).

$$FeatureScore(g, Fake) = \sum_{i=1}^{n} WordScore(w_i, Fake) \quad (12)$$

Where *WordScore*(w_i, *Fake*) can be calculated by using the equation (5).

Similarly, the score of a feature *description* for the *Real* class is given by equation (13).

$$FeatureScore(g, Real) = \sum_{i=1}^{n} WordScore(w_i, Real) \quad (13)$$

Where *WordScore*(w_i, *Real*) can be calculated by using the equation (8).

B. Social Features of news

Social features of the news articles tell how the news article is diffused in the network, who has authored this article and who has published it. Social features play a significant role in proving the authenticity of the article. Because the internet acts as a medium for the diffusion of the news. Thus by examining the social features of the news articles, its authenticity can be established.

B.1. Facebook PageID

This feature tells about the Identification number of the Facebook Page on which the news article was shared or posted. Each page that is created on Facebook by the user has a unique identification number and using this identification number one can easily access the information regarding the admins, the members of that page. It is observed that the Facebook page on which fake or hoax article was posted earlier, then there exists a high chance that in future the authors will use the same page to post another fake or hoax article. Let us assume that a news article was posted on the Facebook page having a page identification number F_p. Then the score of feature *Facebook PageID* (F_p) with respect to the *Fake* class is given by the equation (14).

$$FeatureScore\left(F_p, Fake\right) = \frac{1+Count(F_p,C_f)}{|C_f|} \quad (14)$$

Where C_f is the set of news articles in *C* that are labeled as Fake and *Count*(F_p, C_f) is the number of news articles in C_f that are posted on a Facebook page having a page identification number F_p and $|C_f|$ is the cardinality of the set C_f.

Similarly, the score of feature *Facebook PageID* (F_p) with respect to the *Real* class is given by the equation (15).

$$FeatureScore\left(F_p, Real\right) = \frac{1+Count(F_p,C_r)}{|C_r|} \quad (15)$$

Where, $C_r$is the set of news articles in *C* that are labeled as Real and *Count*(F_p, C_r) is the number of news articles in C_r that are posted on a Facebook page having a page identification number F_p and $|C_r|$ is the cardinality of the set C_r.

B.2. Facebook AppID

When a person uses Facebook Login on a website or a mobile App, an ID is created for the specific Facebook App, which is called *App-Scoped* ID. Using this feature we can get to know that from which AppID the news article was posted on Facebook. It was observed that a single Facebook account is used multiple times to post or share the Fake news. Let us assume that a news article was posted on Facebook using App having an identification number F_A. Then the score of feature *Facebook AppID* (F_A) with respect to the *Fake* class is given by the equation (16).

$$FeatureScore(F_A, Fake) = \frac{1+Count(F_A,C_f)}{|C_f|} \quad (16)$$

Where, C_f is the set of news articles in *C* that are labeled as Fake and *Count*(F_A, C_f) is the number of news articles in C_f that are posted on Facebook using App that has the identification number F_A and $|C_f|$ is the cardinality of the set C_f.

Similarly, the score of the feature F_A with respect to the *Real* class is given by the equation (17).

$$FeatureScore(F_A, Real) = \frac{1+Count(F_A,C_r)}{|C_r|} \quad (17)$$

Where, C_r is the set of news articles in *C* that are labeled as Real and *Count*(F_A,C_r) is the number of news articles in C_r that are posted on Facebook using App that has the identification number F_A and $|C_r|$ is the cardinality of the set C_r.

B.3. Source

This feature indicates the publisher of the news article. Generally, those publishers which publish genuine news articles tend to validate the authenticity of the news article before publishing it. Whereas those publishers who don't check the authenticity of the news before publishing may sometimes publish the fake news. Also, those websites which regularly publishes fake news can be easily identified by examining the URL e.g. *"szabadonebredok.info"*, *"mindenegybenblog.hu"* etc. Thus by examining the URL of the published news article, its authenticity can be established. Let us assume that the news article is published by a Source S_k. Then the score of the feature S_k with respect to the *Fake* class is given by equation (18).

$$FeatureScore(S_k, Fake) = \frac{1+Count(S_k,C_f)}{|C_f|} \quad (18)$$

Where C_f is the set of news articles in *C* that are labeled as Fake and *Count*(S_k,C_f) is the number of news articles in C_f that are published by S_k and $|C_f|$ is the cardinality of the set C_f.

Similarly, the score of the feature S_k with respect to the *Real* class is given by equation (19).

$$FeatureScore(S_k, Real) = \frac{1+Count(S_k,C_r)}{|C_r|} \quad (19)$$

Where $C_r$is the set of news articles in *C* that are labeled as Real and *Count*(S_k, C_r) is the number of news articles in C_r that are published by S_k and $|C_r|$ is the cardinality of the set C_r.

B.4. Authors

This feature contains the list of all the authors of the news article. If the author had published fake news earlier then it is more likely that in future the author may publish a piece of fake news to mislead the potential readers. It is found that most authors of fake news are not even journalists.

Let us assume that there are *m* authors (a_1, a_2, ..., a_m) who have written a news article then the score of feature *authors* with respect to the *Fake* class is given by the equation (20).

$$FeatureScore(Author, Fake) = \sum_{i=1}^{m} P_f(a_i, Fake) \quad (20)$$

Where, $P_f(a_i, Fake)$ is the probability of author a_i to generate *Fake* news articles. If a particular author had published a piece of fake news earlier then this author will have a high value of P_f as compared to the author who hasn't published any fake news in the past. $P_f(a_i, Fake)$ is given by equation (21).

$$P_f(a_i, Fake) = \frac{1+Count(a_i,C_f)}{|C_f|} \quad (21)$$

Where, C_f is the set of news articles in *C* that are labeled as Fake and *Count*(a_i, C_f) is the number of news articles in C_f that are authored by a_i and $|C_f|$ is the cardinality of the set C_f.

Similarly, the score of feature *authors* with respect to the *Real* class is given by equation (22).

$$FeatureScore(Author, Real) = \sum_{i=1}^{m} P_r(a_i, Real) \quad (22)$$

Where,$P_r(a_i, Real)$ is the probability of author a_i to generate *genuine* news articles. If a particular author had published a piece of fake news earlier then this author will have a low value of P_r as compared to the author who hasn't published any fake news in the past. $P_r(a_i, Real)$ is given by equation (23).

$$P_r(a_i, Real) = \frac{1+Count(a_i,C_r)}{|C_r|} \quad (23)$$

Where, $C_r$is the set of news articles in *C* that are labeled as Real and *Count*(a_i, C_r) is the number of news articles in C_r that are authored by a_i and $|C_r|$ is the cardinality of the set C_r.

V. RESULTS

A. Dataset

The proposed model has been tested over the publically available dataset *FakeNewsNet* that was recently published [2]. We have used both the *PolitiFact* and *BuzzFeed* datasets which they provide. The *BuzzFeed* dataset consists of 182 news articles (half of them are labeled as fake) labeled on the basis of the expert opinion of the journalists from BuzzFeed. The *PolitiFact* dataset consists of 240 news articles (half of them are labeled as fake) labeled by the well-recognized fact-checking website PolitiFact (http://www.politifact.com/). Both the datasets provide, the content-based features (headline, body) along with social media based features (how this news article was shared/posted on social media websites like Facebook, Twitter).

B. Performance Evaluation of Proposed Model

Most widely used procedure for evaluating the performance of the classification model (in term of accuracy, specificity, and sensitivity) is by using confusion matrix. This paper uses the most widely accepted metrics with the following classification: -

- True Positive (TP): When the news is predicted as fake and it is actually annotated as fake.
- True Negative (TN): When the news is predicted as real and it is actually annotated as real.
- False Positive (FP): When the news is predicted as fake and it is actually annotated as real.
- False Negative (FN): When the news is predicted as real and it is actually annotated as fake.

The performance metrics are defined as follows:-

1. Precision: Precision attempts to answer the question, what proportions of positive identifications were actually correct? Precision is given by equation (24).

$$Precision = \frac{TP}{TP+FP} \quad (24)$$

2. Recall: Recall attempts to answer the question, what proportion of actual positives was identified correctly? The recall is given by equation (25).

$$Recall = \frac{TP}{TP+FN} \quad (25)$$

3. Accuracy: Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. Accuracy is given by equation (26).

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (26)$$

According to *Accuracy Paradox*, accuracy is not a good metric for predictive models when using class-imbalance dataset. This is because a model may have high accuracy but be too crude to be useful.

4. F1 Score: F1 Score is the harmonic mean of Precision and Recall, as precision and recall alone cannot provide the best evaluation of the classification model. F1 Score is given by equation (27).

$$F1\,Score = \frac{(2*Recall*Precision)}{(Recall+Precision)} \quad (27)$$

The confusion matrix is a table which is generally used to describe the performance of the classification model, it is a table layout which allows the visualization of the performance of the classification model. The confusion matrix also is known as the error matrix.

We have used *Shuffle Split Cross-Validation* [12,13] with 5 iterations for testing the proposed model. In each iteration, the dataset has been split in an 80:20 ratio where 80% dataset is randomly selected for training and the remaining 20% is used for testing. The outcomes obtained are as follows:-

1. First Iteration

In the first iteration, the proposed model was trained on 337 news articles chosen randomly from the dataset out of which 168 news articles were labeled as *Fake* and the remaining 169 news articles as *Real*. Remaining 85 news articles were used for testing purpose out of which 43 news articles were labeled as *Fake* and remaining 42 news articles as *Real*. The outcome obtained in the first Iteration is as shown in Table 1.

Table 1. Confusion Matrix for First Iteration

	Fake	Real
Fake	37 (TP)	6 (FN)
Real	2 (FP)	40 (TN)

The accuracy obtained in the first iteration is 90.58%, with precision 94.87%, recall 86.04% and F1 Score of 90.23% as calculated using equations (24), (25), (26), and (27).

2. Second Iteration

In the second iteration, the proposed model was trained on 335 news articles chosen randomly from the dataset out of which 167 news articles were labeled as *Fake* and the remaining 168 news articles as *Real*. Remaining 87 news articles were used for testing purpose out of which 44 news articles were labeled as *Fake* and remaining 43 news articles as *Real*. The outcome obtained in the second Iteration is shown in Table 2.

Table 2. Confusion Matrix for Second Iteration

	Fake	Real
Fake	39 (TP)	5 (FN)
Real	4 (FP)	39 (TN)

The accuracy obtained in the second iteration is 89.65%, with precision 90.69%, recall 88.63% and F1 Score of 89.64% as calculated using equations (24), (25), (26), and (27).

3. Third Iteration

In the third iteration, the proposed model was trained on 332 news articles chosen randomly from the dataset out of which 168 news articles were labeled as *Fake* and the remaining 164 news articles as *Real*. Remaining 90 news articles were used for testing purpose out of which 43 news articles were labeled as *Fake* and remaining 47 news articles as *Real.* The outcome obtained in the third Iteration is as shown in Table 3.

Table 3. Confusion Matrix for the Third Iteration

	Fake	Real
Fake	38 (TP)	5 (FN)
Real	3 (FP)	44 (TN)

The accuracy obtained in the third iteration is 91.11%, with precision 92.68%, recall 88.37% and F1 Score of 90.47% as calculated using equations (24), (25), (26), and (27).

4. Fourth Iteration

In the fourth iteration, the proposed model was trained on 342 news articles chosen randomly from the dataset out of which 170 news articles were labeled as *Fake* and the remaining 172 news articles as *Real*. Remaining 80 news articles were used for testing purpose out of which 41 news articles were labeled as *Fake* and remaining 39 news articles as *Real*. The outcome obtained in the fourth Iteration is shown in Table 4.

Table 4. Confusion Matrix for Fourth Iteration

	Fake	Real
Fake	36 (TP)	5 (FN)
Real	3 (FP)	36 (TN)

The accuracy obtained in the fourth iteration is 90.00%, with precision 92.30%, recall 87.80% and F1 Score of 89.99% as calculated using equations (24), (25), (26), and (27).

5. Fifth Iteration

In the fifth iteration, the proposed model was trained on 337 news articles chosen randomly from the dataset out of which 169 news articles were labeled as *Fake* and the remaining 168 news articles as *Real*. Remaining 85 news articles were used for testing purpose out of which 42 news articles were labeled as *Fake* and remaining 43 news articles as *Real*. The outcome obtained in the fifth Iteration is as shown in Table 5.

Table 5. Confusion Matrix for Fifth Iteration

	Fake	Real
Fake	37 (TP)	5 (FN)
Real	2 (FP)	41 (TN)

The accuracy obtained in the fifth iteration is 91.76%, with precision 94.87%, recall 88.09% and F1 Score of 91.35% as calculated using equations (24), (25), (26), and (27).

The average accuracy obtained by the proposed model is 90.62% ± 0.75%, with precision 93.08% ± 1.60%, recall 87.78% ± 0.91%, and F1 Score of 90.33% ± 0.57%.

C. Comparisons with earlier Proposed Models

To show the advantage of our proposed model and the usefulness of using data preprocessing in this problem domain, we compare our own proposed probabilistic model with some of the earlier proposed approaches for detecting fake news. Since no standard publically available dataset was used by other proposed models the actual comparison of the performance cannot be done. The models proposed in [5,6,7] were only based on content-based features, whereas the model proposed in [8] only considers social-based features for identifying the fake news. A further modification was done in [4] where the model uses content-based features for demarcation only when the social-based features are less in quantity.

Our proposed model uses both content-based and social-based features for the effective demarcation of fake news from genuine news articles and has achieved remarkable accuracy on a publically available standard dataset that was recently published [2].

VI. Conclusion

This paper presents a novel machine learning model based on natural language processing to achieve an automated detection of fake news articles that are circulated in the network. Results were achieved without any sample selection and are impartial. The result indicates that the model is able to detect fake news with remarkable accuracy and also confirms our hypothesis that better results can be obtained by incorporating both the content-based and social media based features of the news article.

In this paper, we have also shown how the data preprocessing steps lead to better accuracy. Using these preprocessing operations, the model developed was more robust and fast.

Our future work includes the implementation of the proposed model on the real-world platform for automated detection of fake news articles using “news grabber” for updating the database in real time.

References

[1] K. Shu, A. Sliva, S. Wang, J. Tang, and H. Liu (2017), “Fake News Detection on Social Media: A Data Mining Perspective”, *ACM SIGKDD Explorations Newsletter*, vol. 19(1), pp. 22-36.

[2] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu (2018), “FakeNewsNet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media”, *arXiv preprint arXiv:1809.01286.*

[3] K. Shu, S. Wang, and H. Liu (2017), “Exploiting Tri-Relationship for Fake News Detection”, *CoRR arXiv preprint arXiv:*1712.07709.

[4] M. L. D. Vedova, E. Tacchini, S. Moret, G. Ballarin, M. DiPierro and L. de Alfaro (2018), “Automatic Online Fake News Detection Combining Content and Social Signals”, *2018 22nd Conference of Open Innovations Association (FRUCT)*, Jyvaskyla, 2018, pp. 272-279.

[5] B. Riedel, I. Augenstein, G. P. Spithourakis, and S. Riedel (2017), “A simple but tough-to-beat baseline for the Fake News Challenge stance detection task”, *arXiv preprint arXiv:1707.03264*.

[6] H. Ahmed, I. Traore, and S. Saad (2017), “Detection of Online Fake News Using N-Gram Analysis and Machine Learning Techniques”, *Intelligent, Secure, and Dependable Systems in Distributed and Cloud Environments, ser. Lecture Notes in Computer Science, Springer*, pp. 127–138.

[7] S. Badaskar, S. Agarwal, and S. Arora (2008), “Identifying Real or Fake Articles: Towards better Language Modeling”, *IJCNLP*, pp. 817–822.

[8] E. Tacchini, G. Ballarin, M. L. Della Vedova, S. Moret, and L. de Alfaro (2017), “Some Like it Hoax: Automated Fake News Detection in Social Networks”, *Proceedings of the Second Workshop on Data Science for Social Good*, Skopje, Macedonia, vol. 1960.

[9] A. Bessi, M. Coletto, G. A. Davidescu, A. Scala, G. Caldarelli, and W. Quattrociocchi (2015), “Science vs Conspiracy: Collective Narratives in the Age of Misinformation”, *PLOS ONE*, vol. 10(2), pp. e0118093.

[10] H. Allcott and M. Gentzkow (2017), “Social Media and Fake News in the 2016 Election”, *Journal of Economic Perspectives*, vol. 31(2), pp. 211–236.

[11] H. Karimi, P. C. Roy, S. S. Sadiya, and J. Tang (2018), “Multi-Source Multi-Class Fake News Detection”, *Proceedings of the 27th International Conference on Computational Linguistics*, New Mexico, USA, pp. 1546–1557.

[12] M. A. Little, G. Varoquaux, S. Saeb, L. Lonini, A. Jayaraman, D. Mohr, and K. Kording (2017), “Using and understanding cross-validation strategies. Perspectives on Saeb et al”, *GigaScience*, vol. 6.

[13] S. Arlot, and A. Celisse (2010), “A survey of cross-validation procedures for model selection”, *Statistics Surveys*, vol. 4, pp. 40-79.

[14] M. Balmas (2012), “When Fake News Becomes Real: Combined Exposure to Multiple News Sources and Political Attitudes of Inefficacy, Alienation, and Cynicism”, *Communication Research*, vol. 41(3), pp. 430-454.

[15] S. Jang, and K. K. Joon (2018). “Third person effects of fake news: Fake news regulation and media literacy interventions”, *Computers in Human Behavior,* vol. 80, pp. 295-302.

[16] A. Guess, J. Nagler, and J. Tucker (2019), “Less than you think: Prevalence and predictors of fake news dissemination on Facebook”, *Science Advances,* vol. 5(1), pp.

[17] K. Shu, D. Mahudeswaran, and H. Liu (2018), “FakeNewsTracker: a tool for fake news collection, detection, and visualization”, *Computational and Mathematical Organization Theory,* vol. 25(1), pp. 60-71.

Authors’ Profiles

Shubham Bauskar was born in Burhanpur, India in 1998. He is currently pursuing a bachelor’s degree in Computer Science and Engineering from Maulana Azad National Institute of Technology, Bhopal, India. His research interests include Machine Learning, Natural Language Processing, and Pattern Recognition.

Vijay Badole was born in Barwani, India in 1996. He is currently pursuing a bachelor’s degree in Computer Science and Engineering from Maulana Azad National Institute of Technology, Bhopal, India. His research interests include Data Structures and Algorithms and Machine Learning.

Prajal Jain was born in Jobat, India in 1997. He is currently pursuing a bachelor’s degree in Computer Science and Engineering from Maulana Azad National Institute of Technology, Bhopal, India. His research interests include Data Structures, Algorithms and Machine Learning.

Dr. Meenu Chawla completed her BE (Computer Technology) from Maulana Azad College of Technology, India in 1990. She did her M. Tech (Computer Science and Engineering) at the Indian Institute of Technology, Kanpur, India, in 1995 and received her Ph.D. in the area of Ad hoc Networks (Computer Science) from Maulana Azad National Institute of Technology, India, in 2012. She has more than 25 years of teaching and research experience. Currently, she is a Professor in the Department of Computer Science and Engineering at Maulana Azad National Institute of Technology, India. She has published more than 50 research papers in the reputed journals and conferences. She is a Member of IEEE, CSI, and ISTE. Her research and teaching interests include Data Structure and Algorithms, Wireless communication and Mobile Computing, Mobile Ad Hoc and Sensor Networks, Cognitive Radio Networks and Big Data.

How to cite this paper: Shubham Bauskar, Vijay Badole, Prajal Jain, Meenu Chawla, " Natural Language Processing based Hybrid Model for Detecting Fake News Using Content-Based Features and Social Features", International Journal of Information Engineering and Electronic Business(IJIEEB), Vol.11, No.4, pp. 1-10, 2019. DOI: 10.5815/ijieeb.2019.04.01