Journal of Intelligent & Fuzzy Systems xx (20xx) x–xx
DOI:10.3233/JIFS-179905
IOS Press

"Bend the truth": Benchmark dataset for fake news detection in urdu language and its evaluation

Maaz Amjad[a], Grigori Sidorov[a,*], Alisa Zhila[a], Helena Gómez-Adorno[b], Ilia Voronkov[c] and Alexander Gelbukh[a]

[a]*Centro de Investigación en Computación (CIC), Instituto Politécnico Nacional, Mexico*
[b]*Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas (IIMAS), Universidad Nacional Autónoma de México, Mexico*
[c]*Moscow Institute of Physics and Technology, Russia*

Abstract. The paper presents a new corpus for fake news detection in the Urdu language along with the baseline classification and its evaluation. With the escalating use of the Internet worldwide and substantially increasing impact produced by the availability of ambiguous information, the challenge to quickly identify fake news in digital media in various languages becomes more acute. We provide a manually assembled and verified dataset containing 900 news articles, 500 annotated as real and 400, as fake, allowing the investigation of automated fake news detection approaches in Urdu. The news articles in the truthful subset come from legitimate news sources, and their validity has been manually verified. In the fake subset, the known difficulty of finding fake news was solved by hiring professional journalists native in Urdu who were instructed to intentionally write deceptive news articles. The dataset contains 5 different topics: (i) Business, (ii) Health, (iii) Showbiz, (iv) Sports, and (v) Technology. To establish our Urdu dataset as a benchmark, we performed baseline classification. We crafted a variety of text representation feature sets including word n-grams, character n-grams, functional word n-grams, and their combinations. After applying a variety of feature weighting schemes, we ran a series of classifiers on the train-test split. The results show sizable performance gains by AdaBoost classifier with 0.87 $\mathrm{F1}_{\mathrm{Fake}}$ and 0.90 $\mathrm{F1}_{\mathrm{Real}}$. We provide the results evaluated against different metrics for a convenient comparison of future research. The dataset is publicly available for research purposes.

Keywords: Fake news detection, urdu corpus, language resources, benchmark dataset, classification, machine learning

1. Introduction

Even though the Urdu language has more than 100 million speakers across the world, it is a resource poor languages in the Natural Language Processing (NLP) domain both from the perspective of NLP tools inaccessibility as well as scarcity of labeled datasets [1]. In this work, we dedicate our attention to assemble a plausible and credible source in the form of Urdu corpus for automatic fake news detection.

In digital media, the epidemic of fake news grows substantially when a change in public opinion is demanded during an important event. Hence, we need to tap natural language processing algorithms to design a system that can determine whether a source is trustworthy or politically inclined with or without human curation. For example, in January 2019 Google showed incorrect Pakistani rupee value against US dollar (exchange price of the dollar) in

[*] Corresponding author. Grigori Sidorov, Mexico City, Mexico. E-mail: sidorov@cic.ipn.mx.

Pakistan [1], the following day stock market in Pakistan was crashed because people started to sell their shares due to the dramatic decline of the stock exchange.

Fake news is painting significant challenges to branch out our society. The availability of information has raised the challenges associated with testing the trustworthiness of the data automatically. For this reason, it is necessary to build systems for controlling the amount of factually incorrect and misleading data on the Web. This deficit can be met by designing computational models for detecting fake news. In turn, this requires sufficient amount of labeled data to apply supervised machine learning approaches. As fake news dissemination can be cross-lingual, it is best to have datasets available in a wide variety of languages.

Therefore, we present a labeled dataset for fake news detection in Urdu language. It contains 500 labeled real news from legitimate news sources and 400 fake news in the corresponding topics: (i) Business, (ii) Health, (iii) Showbiz, (iv) Sports, and (v) Technology.

Additionally, we provide baseline classification methods for fake news detection on this dataset. There are three categories of fake news detection methods [3]: knowledge-based (attempt fact verification), context-based (analyze how the news disseminate in social networks), and style-based (analyze writing style). The problem with implementing the first two approaches for the Urdu language is the unavailability of the NLP tools required for their intermediate feature crafting. However, the style-based approach in its basic form is based on analyzing *n*-gram sequences.

The main contributions of this work are:

- the first corpus for the Urdu language for research on automatic fake news detection containing real news extracted from various legitimate news agencies and fake news written by professional native Urdu-speaking journalists;
- corpus development methodology. This corpus is a unique resource to study style-based fake news detection models deeply;
- the description of the challenges faced in assembling the fake news part;
- statistical metrics for the corpus vocabulary;
- recommendations for most effective feature combination;
- a comparison of supervised learning classifiers and their performance in fake news detection based on linguistic and stilometric features;
- baseline classification results evaluated against a number of metrics with the best results of *0.86* $F1_{\text{Fake}}$ score for fake news detection, *0.89* $F1_{\text{Real}}$ score for real news detection, and *0.95 ROC-AUC*.

The rest of the paper is composed as follows. Section 2 overviews the state-of-the-art work on fake news detection and corpora for other languages. Section 3 describes the methodology we followed for building the corpus along with the annotation guidelines and the corpus statistics. Section 4 describes the classification approach for automatically detecting fake news. In Section 5 we analyze the experimental results. Subsequently, Section 6 present general conclusions and points to the permissible steps of future work.

2. Related work

In this section, we review the literature regarding the automatic analysis of fake news which has been a subject of particularly acute attention. The presence of fake news started with the invention of printing press back in 1439 [2]. However, there are divergent opinions in defining the term "fake news."

Definition 1. Fake News: *Fake news is a news article that is intentionally and verifiably false* [3].

In recent times, there are only two main directions of research to automatically classify fake news: on a conceptual and an operational level. On a conceptual level, fake news have been further divided into three categories [4]: hoaxes, *i.e.*, posting factitious information using social networks alluding to certain news broadcast in its genuine form via reputable news websites; satire, *i.e.*, news that imitate the real content of news with addition of untrue and sarcastic content; and serious fabrications, *i.e.*, misleading news about a celebrity or an event that did not take place.

On an operational level, researchers [5] suggested different approaches, such as an inference task in a Markov random field (MRF) [6], fact-checking, and source-checking. Moreover, fake news detection and deception detection has been used in several studies as a data mining [3] to classify news pieces, posts,

[1] https://www.thenews.com.pk/latest/419799-google-currency-undergoes-glitch

[2] https://www.politico.com/magazine/story/2016/12/fake-news-history-long-violent-214535

and online reviews in publicly available corpora [7, 8].

Fake news pieces contain dogmatic and seditious language to urge users to click on the link to read the full article (known as “clickbait”) [9]. Thus, in the fake news detection task, the linguistic features have been used to capture the different writing styles in news content and sensational headlines [7]. Additionally, social networks, in particular Twitter posts associated with natural disasters have been used for the development of fake news detection model [10].

Linguistic-based features are derived from language to examine various aspects of a language at different levels such as characters, words, sentences, and documents as a whole. There are two primary types of features: common and domain-specific linguistic features. Common linguistic features contain two kind of features: (i) lexical features and (ii) syntactic features. Lexical features including character level and word-level features, such as a total number of words, an average number of characters per word, frequency of words present in the dataset, unique word count, frequency of function words and phrases, parts-of-speech (POS) tags, etc. Syntactic features include sentence-level features such as syntactic dependencies/constituents, clauses, and punctuation. Ivanov and Tutubalina [11] use syntactic clause features for user review analysis.

Domain-specific linguistic features, which are precisely aligned to news domain, are quoted words, external links, number of images, etc. [2]. Furthermore, to find out the deceptive cues in writing styles to flag fake news, features such as author lying-detection features and different types of new features can be created [12].

The reason to attain features is to outline the content of news items mathematically. Model-oriented fake news research opens the door to developing more robust models for fake news detection. Recent studies [2, 3, 13] have suggested different approaches by focusing on extracting several kinds of features and integrating them into supervised classification models such as logistic regression(LR), k-nearest neighbours (kNN), random forest (RF), and support vector machines (SVM), and after that choosing the classifier that outperform other machine learning algorithms.

A recent study suggested a multi-task ordinal regression framework that models the problem of trustworthiness and political ideology detection of entire news content jointly instead of analyzing each news article individually. Furthermore, this study also revealed that joint models over models that tackle the problems separately obtained significantly better results [14].

Additionally, fake news detection has been investigated as a stance detection problem rather than true/false classification [15, 16]. In particular, this approach was adopted in the Fake News Challenge project (FNC1)[3] which reduces problem to checking the relationship between the title and the body of the news: a) the title and the body are clearly related, b) no association between the title and the body, and c) a partial relationship.

The winning team (best performing system) achieved 82.02 accuracy score using machine learning and deep learning approaches [17]. For machines, 2-gram and 3-gram features with TF-IDF weighting scheme using Gradient-Boosted Decision tree were used. For deep learning, word level vectors using *word2vec* embeddings from Google News [18] were applied using a one-dimensional deep convolutional neural network (CNN) on the title and the body text.

News articles can be accumulated using different online sources, such as news agency homepages, search engines, and social networks. Despite this, manually checking the authenticity of a news article requires annotators with domain expertise who conduct deliberate analysis of claims. For fake news detection, datasets in English and Spanish are available. In English the datasets are available, such as BuzzFeedNews [2], BS Detector[4], *Liar* [19], CREDBANK [20] and FakeNewsNet [21]. Likewise, Fake News Corpus Spanish [13] annotated for fake news detection in Spanish. A corpus of social network news feed paraphrases exist for the Russian language, yet it is not annotated for content authenticity [22]. However, to the best of our knowledge, there is still no such resource available in the Urdu language despite the tremendous advancement in research work for this language.

EMILLE[5] Project (Enabling Minority Language Engineering) was the first initiative to assemble a 67 million word corpus of South Asian languages [23]. The Urdu corpus which was collected within this project contained approximately 0.5M spoken Urdu words transcribed from transmissions of BBC Asian Network and BBC Radio. Subsequently, researchers started to make attempts to build resources for resource-poor language, such as the Urdu corpus

[3] http://www.fakenewschallenge.org

[4] https://www.kaggle.com/mrisdal/fake-news

[5] http://www.emille.lancs.ac.uk

for word sense disambiguation [24], the Urdu POS-tagged corpus [25], and the initiative of phonetically rich Urdu corpus for speech recognition [26]. However, these resources do not have annotation suitable for fake news detection.

3. The data: the first corpus of fake news in urdu

In this section, we provide an overview of the data acquisition process as well as the corpus statistics. We assembled real news by crawling thousands of news articles from numerous reliable sources for the time frame from January 2018 to December 2018.

This corpus contain news from five domains: (i) Business, (ii) Health, (iii) Showbiz (entertainment), (iv) Sports, and (v) Technology. This selection of topics is in line with a similar dataset for English language [7] except for the educational domain which presented difficulties in obtaining.

Previous study [2, 27] has almost exclusively focused on providing a more detailed analysis of procedures on how two types of news (real and fake) are collected. It also discussed serious issues associated with fake news corpus. Moreover, some news corpora contain news articles which are a combination of real and fake information. As far as we know, no previous research has illuminated the rigorous criteria for fake news definition and categorization. Although researchers have examined different types of news in creating a corpus, some questions regarding the exact procedure of how they annotated the news pieces remain to be addressed. With this in mind, we introduced an alternative approach to data collection to address this limitation in fake news annotation and applied it to the Urdu language.

This "Bend The Truth" corpus is a unique, reasonably accurate, and reliable source of its kind in the Urdu language for this particular task. Urdu is a national language of Pakistan. This is a binary annotated corpus. The uniqueness about this corpus apart from its language, is that we availed professional journalist services to write fake news stories corresponding to the original real news, just as what takes place in the real life. News agencies used to crawl real news are mentioned in Table 1. The "Bend The Truth" Urdu corpus is publicly available to use for academic research [6].

Table 1
Legitimate websites

Name	URL	Origin
BBC News	www.bbc.com/urdu	England
CNN Urdu	cnnurdu.us	USA
Dawn news	www.dawnnews.tv	Pakistan
Daily Pakistan	dailypakistan.com.pk	Pakistan
Eteemad News	www.etemaaddaily.com	India
Express-News	www.express.pk	Pakistan
Hamariweb	hamariweb.com	Pakistan
Jung News	jang.com.pk	Pakistan
Mashriq News	www.mashriqtv.pk	Pakistan
Nawaiwaqt News	www.nawaiwaqt.com.pk	Pakistan
Roznama Dunya	dunya.com.pk	Pakistan
The daily siasat	urdu.siasat.com	India
Urdu news room	www.urdunewsroom.com	USA
Urdupoint	www.urdupoint.com	Pakistan
Voice of America	www.urduvoa.com	USA
Waqt news	waqtnews.tv	Pakistan

3.1. Data crawling

The Newspaper[7] library for *Python* was used as a web scraper to extract the content of news articles from newspaper web pages. This library offers advanced features to deal with web pages of newspapers and magazines to extract news articles. This capability was essential for obtaining not only the relevant text of Urdu news articles by husking additional obsolete HTML tags but also eliminating Urdu text which did not belong to the news text body (*e.g.*, name of the author, location). Despite that HTML structure of each news source (website) is different, this scrapper performed exceptionally good job dealing with noisy texts, images, and advertisements. For evaluation of the performance of our method, we need balanced corpus and this is why besides fake news news, we also need real news.

3.2. Real news collection

The real news were collected from different mainstream news websites. The major points in the real news data collection and handling procedure were:

- The data was collected and annotated manually.
- The news piece was labeled as real if it fell into one of the following categories:
    1. It was published by a reliable newspaper and prominent news agency.
    2. The same news was found on different newspapers which provided evidence

[6] https://github.com/MaazAmjad/Datasets-for-Urdu-news.git

[7] https://newspaper.readthedocs.io/en/latest

about the authenticity of the news, such as image, date, place of the event, etc.

3. The source of the news is mentioned and that source is reliable. Subsequently, we verified the news source and cross-referencing information among several sources.
4. There is correlation between title and the contents of the news article. To verify correlation between the title and the contents, we had to read all the news articles.

The length of the news pieces in this collection varies because each news agency has a different style of news articles. So, the length of each news content is heterogeneous. Using this methodology we collected 100 news in each of the five domains, for a total of 500 real news.

3.3. Professional crowdsourcing of fake news

The collection of fake news for the corresponding real news was a challenging task. The reason was that it demanded a tremendous amount of work to be done for evaluating fake news. Firstly, there are no websites that offer news validation services for the Urdu language. Consequently, the web scraping approach was out of consideration as it would require manual analysis of hundreds of thousands of news articles for authenticity. Therefore, generating fake news of the corresponding real news was the alternative we chose. For writing fake news, we drew great benefits from professional journalists from various news agencies in Pakistan: Express news, Dawn news, etc. Using the services of professional journalists ensured the quality of the fake news articles and realistically imitated the process that happens in real life when fake news are created.

As our dataset covered news articles in five major domains (sports, business, education, technology), the news cannot be the same from the linguistic point of view. Thus, we tasked journalist who were experts in a corresponding domain.

We provided the journalists with very open-ended instructions to avoid unintentionally introducing any clearly defined patterns that would make the produced news pieces easily distinguishable from the real news. Journalists were asked to keep the same length of the news as the original. For this task, we largely relied on the journalists' expertise.

3.4. Problems in collecting real and fake news

During the Real news collection, there were some problems found, *e.g.*, typing mistakes or word misuse (see the concrete examples below). To avoid such errors, it was required to re-read the whole news corpus and remove such faults.

- Another example, the word "کیساتھ" , which means (*"with"*), is sometimes spelled as "وتھ" by some Indian newspapers.
- Urdu has compound words (*i.e.*, consisting of several tokens), *e.g.*, "خیر و عافیت" (*"with safety"*), "آب و هوا" (*"climate"*), "آباؤ اجداد" (*"forefathers"*), "جدید کاری" (*"modernization"*). Such compound words are split into two or three tokens by the standard tokenizers. However, they are actually a single word, and we needed to be very careful while tokenizing these words. However, in our experiments, we didn't do any additional tokenization step for compound words and used default splitting.
- Some Indian newspapers misreport artists' names. The newspaper mentioned "قرینه" instead of "کرینه" (in English transliterated as *"Katrina"*).
- Some Indian newspapers report grammatical gender (masculine and feminine) differently than Pakistanis newspapers. For example, "تقریب" (in the English language, it means *"occasion"* or *"Event"*). According to the Indian newspaper, *"event"* is masculine. On the other hand, Urdu newspapers report *"event"* as feminine.
- Some newspapers use Roman numerals as "۲۲".
- Some newspaper had typed written mistakes such as "اکٹوبر" instead of "اکتوبر" *"October"*.
- Some Hindi sports newspapers write "Matches" as "میاچس" and "میاتھیس" instead of "میچز" . Additionally, the word "Test Series" is written as "ٹسٹ سریز" instead of "ٹیسٹ سریز".
- In health news, there were some mistakes which completely change the meaning of the sentence. For example, in one health news, it was stated "گٹھیا" which means *"stupid"* instead of "گٹھنا" which means *"ancle"*.

The journalists were asked to read a full news article before writing a fake version of it, which required substantial effort and time to write each fake news.

Table 2
Urdu Corpus for Fake News distribution by topics

Category	Real	Fake
Business	100	50
Health	100	100
Showbiz	100	100
Sports	100	50
Technology	100	100
Totals	**500**	**400**

Table 3
Vocabulary size of distributed corpus

Category	Train		Test	
	Real	Fake	Real	Fake
Business	4,640	1,939	2,822	862
Health	3,825	3,454	2,283	2,091
Showbiz	3,695	3,851	2,919	2,953
Sports	4,948	2,178	3,365	536
Technology	4,494	4,679	2,448	2,458
Total	**13,250**	**10,115**	**8,848**	**6,610**

3.5. Data pre-processing and data cleaning

We enhanced the quality of the text data after extraction with the scrapper by performing additional data cleaning on the plain text of news articles. We took the following steps:

1. All auxiliary character sequences and tokens in Latin alphabet, *e.g.*, special characters such as the description of the images in news, references to images, videos, were discarded manually.
2. However, we did not eliminate punctuation marks from Western Latin character sets.
3. Tokenization (splitting sentences into words/tokens) is performed on the white space character. Sentences with less than two tokens are not included.
4. Ramification of paragraphs into sentences is performed on Urdu sentence end markers, *e.g.*, question mark (*?*), full stop (-).
5. Numerals in the Eastern Arabic-Indic system were converted to Western Arabic to normalize the entire data. Noise from the data in the form of white space tokens, bullets, smiley icons (emojis) is removed.
6. We use the standard *utf-8* codification. Invalid *utf-8* characters were discarded.
7. The title of the news is also included in the corpus as a part of an article.

3.6. Corpus statistics

The Table 2 presents the distribution of the news articles collected from five major domains.

Further, we performed statistical description of the corpus. All the stop words and lemmas are taken into account. All tokens were lower-cased. We calculated the vocabulary size (*i.e.*, the number of unique tokens) for each topic domain. The Table 3 indicates the vocabulary size of the distributed data used for testing and training phase.

The vocabulary overlap between real and fake news articles is calculated as shown in Table 4. The vocabulary overlap in train set is 47.38%, and in the test set is 45.14%. The vocabulary overlap is calculated by the vocabulary (words) present in both news classes (real and fake) divided by the entire dictionary.

4. Fake news detection experiments

In this section, we describe a series of experiments on automatic fake news detection set as a binary classification problem (real or fake). We explore various combinations of feature sets, look at different feature value weighting schemes (scalers), and try out a number of classifiers. This is done to find a best performing baseline classifier for the assembled dataset.

Table 4
Vocabulary overlap within each category in the complete corpus

Category	Train			Test		
	Real	Fake	Overlap	Real	Fake	Overlap
Business	4,640	1,939	34.65%	2,822	862	25.69%
Health	3,825	3,454	55.63%	2,283	2,091	72.41%
Showbiz	3,695	3,851	49.99%	2,919	2,953	46.84%
Sports	4,948	2,178	34.53%	3,365	536	13.93%
Technology	4,494	4,679	56.54%	2,448	2,458	56.49%
Totals	**13,250**	**10,115**	**47.38%**	**8,848**	**6,110**	**45.14%**

Table 5
Domain Distribution in Train and Test subsets

Domain	Train		Test	
	real	fake	real	fake
Business	70	36	30	14
Health	70	70	30	30
Showbiz	70	70	30	30
Sports	70	42	30	8
Technology	70	70	30	30
Totals	**350**	**288**	**150**	**112**

4.1. Dataset split

To prepare the data for the experiments, the corpus was split into train and test sets with 70% and 30% ratio, respectively. In particular, all five domains were distributed proportionally such as 70% news articles of each domain belongs to the train set and the resting 30% belongs to the test set. The Table 5 described the corpus distribution for training and testing sets.

4.2. Features

Several sets of n-gram based features, such as character n-grams, word n-grams, and function words (see below) n-grams, with n varying from 1 to 6 have been used to build the fake news detection models.

Character n-grams. We used sequences of characters of different sizes (from 1 to 6). These features were used to capture morphological and syntactical information embedded in texts. Previous work [2, 13] showed that character n-grams achieved significant improvements in detecting fake news.

Words n-grams. Previous studies on fake news detection [7] have shown that the standard bag-of-words model is considered a baseline to seek out whether a specific selection of words can highlight the attributes of fake news. In this feature set we included experiments with n-gram sizes from 1 (*i.e.*, the standard bag-of-words model) to 6.

Function words n-grams. Function words include articles, prepositions, determiners, conjunctions, and auxiliary verbs. This kind of n-grams are composed of n consecutive function words omitting all the content words in between. Study [2, 13] demonstrated that the use of function words might be useful to separate credible news and fake new. Furthermore, recent work [28] also showed that function word n-grams achieved significant improvements in detecting writing style of authors. They also demonstrated that word and character n-grams without stop words failed to provide better results. We used a standard stop word list for Urdu as function words[8].

4.3. Feature combinations

Combination of different N-gram sizes are important to recognize fake news. The minimum, average and maximum number of features we used in our experiments are 18, 4,079, and 41,125 respectively.

4.4. Weighting schemes

A number of approaches are available to calculate values for n-gram features and their scaling across features. We consider them "weighting schemes". We also used Frequency distribution as Frequency distribution has been used to attempt to understand the lexical structure of a text. Weighting schemes: binary values, raw frequency, relative frequency, normalized frequency, log-entropy weighting, and TF-IDF are investigated.

Raw frequency is the arithmetic count of the number of times an n-gram was encountered.

Relative Frequency. It is a maximum likelihood estimation of probability; divide the count of a word (the frequency) by the total number of words N in a dataset. It estimates the probability of a word being present by the relative frequency of words in a dataset:

$$\hat{P}(X = x) = \frac{f(x)}{N},$$

where X is a discrete numerical variable.

Binary weighting scheme. A binary weighting scheme constrains feature values to only two options, 1 and 0. We followed the scheme per [29], where $w_i = 1$ if $tf_i > 0$ and $w_i = 0$ if $tf_i = 0$, where tf_i is defined as the number of times term i appears in document D.

Normalized frequency. The normalized frequency weighting scheme scales each sample to have a unit norm. This means that each sample (feature vector) is rescaled so that its $L2$ norm equals one. The $L2$ norm [30] is:

$$||x|| = \sqrt{x^T x} = \sqrt{\sum_{i=1}^{d} x_i^2},$$

"Unit $L2$ norm" means that for each data point x, $||x|| = 1$. A vector x can be normalized using $\frac{x}{||x||}$.

[8] https://www.kaggle.com/rtatman/urdu-stopwords-list#stopwords-ur.txt

Log Entropy. Log entropy weight for the term *i* in the document *j* is calculated per [31]:

$$\mathit{Local\ Weight}_{ij}(L_{ij}) = log(tf_{ij}) + 1,$$

$$\mathit{Probability}_{ij}(P_{ij}) = \frac{tf_{ij}}{\sum_{j=1}^{m}(tf_{ij})},$$

$$\mathit{Global\ Weight}_{ij}(G_i) = 1 + \frac{\sum_{j=1}^{m}(P_{ij}) \times log P_{ij}}{log(m+1)},$$

$$\mathit{Final\ Weight}_{ij}(a_{ij}) = (L_{ij}) \times (G_i),$$

where (tf_{ij}) is a number of times term *i* appears in document *j*, *m* is a total number of documents.

TD-IDF. TF-IDF weight of term *i* in document *j* in a corpus of *N* documents is calculated as:

$$\mathit{Weight}_{ij} = tf_{ij} \times log(\frac{N}{df_i}),$$

where (tf_{ij}) is a number of times term *i* appear in document *j* and (df_i) is a number of document containing term *i*.

4.5. Classifiers

We considered a number of machine learning classifiers to find the best performing classifier for fake news identification task on our corpus. These classifiers include Multinomial Naive Bayes (MNB), Bernoulli Naive Bayes (BNB), Support Vector Machines (SVM), Logistic Regression (LR), Random Forests (RF), Decision Tree (DT), and AdaBoost (AB). These classifiers have been used in various NLP tasks and obtained state-of-the-art performance in tasks, such as in opinion mining studies [32], authorship attribution [33], sentiment analysis [34]. We used the Scikit-learn [35] implementation of the above mentioned classifiers with their default parameters.

4.6. Experiments

In this subsection, we describe our approach to the experiment generation. We run an experiment for each feature set, for each weighting scheme, and for each classifier. The total number of experiments are 2880 using different representation of text features.

4.7. Metrics and evaluation

For evaluations, we used the following performance metrics: balanced accuracy, F1_{Real} score, F1_{Fake} score, and ROC-AUC. Balanced accuracy is used as we want to label both fake and real news correctly.

Balanced accuracy is the average of recall obtained on each class. It has been commonly used to deal with imbalanced corpus.

A trivial classification baseline is established by assigning all the news in the test subset to one of the classes. Since dataset contain more real news articles and is essentially the real-world case, we assigned label "real" to all instances in the test subset. Our trivial assignment as all truthful provides the baseline using accuracy score of 0.55 as reference value. We perform 10-fold cross-validation for each experiment on the train subset and run each experiment once on the test subset. No parameter fine tuning is performed.

5. Result analysis

This section presents the analysis of the experimental results and provides recommendations for the set of features, weighting schemes, and classifiers based on the best performing combinations of those.

We present the results as top 10 best performing experimental combinations by F1_{Real} score in Table 6, top 10 best performing experimental combinations by F1_{Fake} score in Table 7, and the experiments that are best performing by both F1_{Fake} score and F1_{Real} score, i.e., those experimental combinations that achieve highest performance both in detecting fake news as fake and real as real, in Table 8.

As it can be seen in the tables, all best performing experiments achieved performance well above the trivial single-class baseline of 0.55 , which indicates that the task of fake news detection can be effectively addressed using *n*-gram features. The maximum F1_{Fake} score was achieved by AdaBoost (often called boosted decision tree) classifier which outperformed other classifiers with the peculiar combinations of character-word 2-grams and 1-grams (*i.e., 2c-1w-0f*) by providing 0.87 F1_{Fake} score on the test set.

Feature Combinations. Our experiments covered combinations of *n*-grams ranging in size from 1 to 6. We observed that performance decreases with the increase of *n*-grams size. Apparently, this is due to the exponential growth of the feature space dimension. For example, for a feature combination *4c-4w-4f*, the total features are 22548, and for the feature combination *6c-6w-6f* the total features are 37208. Our training dataset size, 638 instances, is not enough to

Table 6
Top 10 classification results by $\mathrm{F1}_{\mathrm{Real}}$ score

Character *n*-gram	Word *n*-gram	Func. *n*-gram	Total Features	Weight scheme	Classifier	Balanced Acc	Roc	F1-Fake	F1-Real
2	1	1	5891	TFIDF	Ada Boost	0.88	0.95	0.86	0.90
2	1	2	8690	Binary	Ada Boost	0.88	0.95	0.86	0.90
2	1	2	8690	TFIDF	Ada Boost	0.88	0.95	0.85	0.90
2	1	2	8690	Norm	Ada Boost	0.88	0.95	0.87	0.90
2	1	2	8690	TF	Ada Boost	0.88	0.95	0.87	0.90
2	0	1	1961	Logent	Ada Boost	0.86	0.95	0.85	0.89
2	0	2	4760	Norm	Ada Boost	0.86	0.95	0.85	0.89
2	0	2	4760	TF	Ada Boost	0.86	0.95	0.85	0.89
2	1	0	5596	Binary	Ada Boost	0.87	0.95	0.86	0.89
2	1	0	5596	TFIDF	Ada Boost	0.86	0.95	0.84	0.89

Table 7
Top 10 classification results by $\mathrm{F1}_{\mathrm{Fake}}$ score

Character *n*-gram	Word *n*-gram	Func *n*-gram	Total Features	Weight scheme	Classifier	Balanced Acc.	Roc	F1-Fake	F1-Real
2	1	2	8690	Norm	Ada Boost	0.88	0.95	0.87	0.90
2	1	2	8690	TF	Ada Boost	0.88	0.95	0.87	0.90
2	1	0	5596	Binary	Ada Boost	0.87	0.95	0.86	0.89
2	1	0	5596	Norm	Ada Boost	0.87	0.95	0.86	0.89
2	1	0	5596	TF	Ada Boost	0.87	0.95	0.86	0.89
2	1	1	5891	Binary	Ada Boost	0.87	0.95	0.86	0.89
2	1	1	5891	TFIDF	Ada Boost	0.88	0.95	0.86	0.89
2	1	1	5891	Norm	Ada Boost	0.87	0.95	0.86	0.89
2	1	1	5891	TF	Ada Boost	0.87	0.95	0.86	0.89
2	1	2	8690	Binary	Ada Boost	0.88	0.95	0.86	0.90

Table 8
Top classification results by both $\mathrm{F1}_{\mathrm{Real}}$ and $\mathrm{F1}_{\mathrm{Fake}}$ scores

Character *n*-gram	Word *n*-gram	Func *n*-gram	Total Features	Weight scheme	Classifier	Balanced Acc.	Roc	$\mathrm{F1}_{\mathrm{Fake}}$	$\mathrm{F1}_{\mathrm{Real}}$
2	1	0	5596	Binary	Ada Boost	0.87	0.95	0.86	0.89
2	1	1	5891	TFIDF	Ada Boost	0.88	0.95	0.86	0.90
2	1	2	8690	Binary	Ada Boost	0.88	0.95	0.87	0.90
2	1	2	8690	Norm	Ada Boost	0.88	0.95	0.87	0.90
2	1	2	8690	TF	Ada Boost	0.88	0.95	0.87	0.90

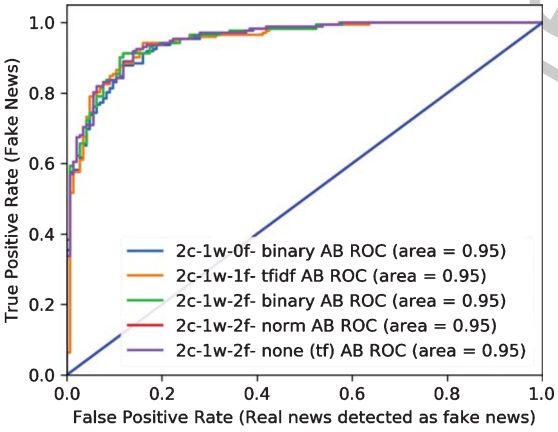


Fig. 1. ROC curves for best performing experimental combinations.

train a classifier in such highly dimensional feature spaces. Therefore, we omit the results for *n*-grams sizes 4 and higher as they are inferior to the performances for smaller *n*-grams.

Figure 1 illustrates ROC-curves for the experimental combinations from Table 8.

Additionally, both Tables 6 and 7 contain feature sets that have only one type of *n*-gram, *e.g.*, *2c-0w-0f* includes only character bi-grams, two types, *e.g.*, *2c-1w-0f* and *2c-0w-1f*, and all three *n*-gram types jointly. In addition, while character *n*-grams can lead to top results individually, there’s no feature combination where it would be absent. Hence, we conclude that character *n*-grams are the most descriptive features for fake news detection in Urdu. Further, as the tables show, their text representation can be enhanced by either of word *n*-grams or function word *n*-grams

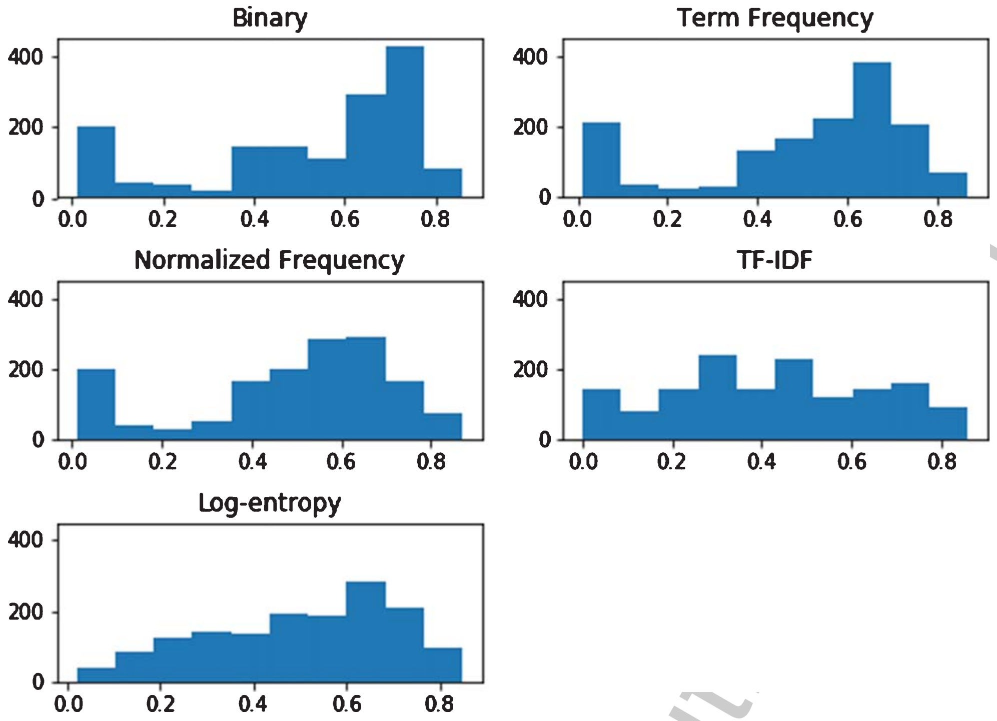


Fig. 2. Weighting scheme performance in terms of F1-score distribution. Y-axes show experiment counts.

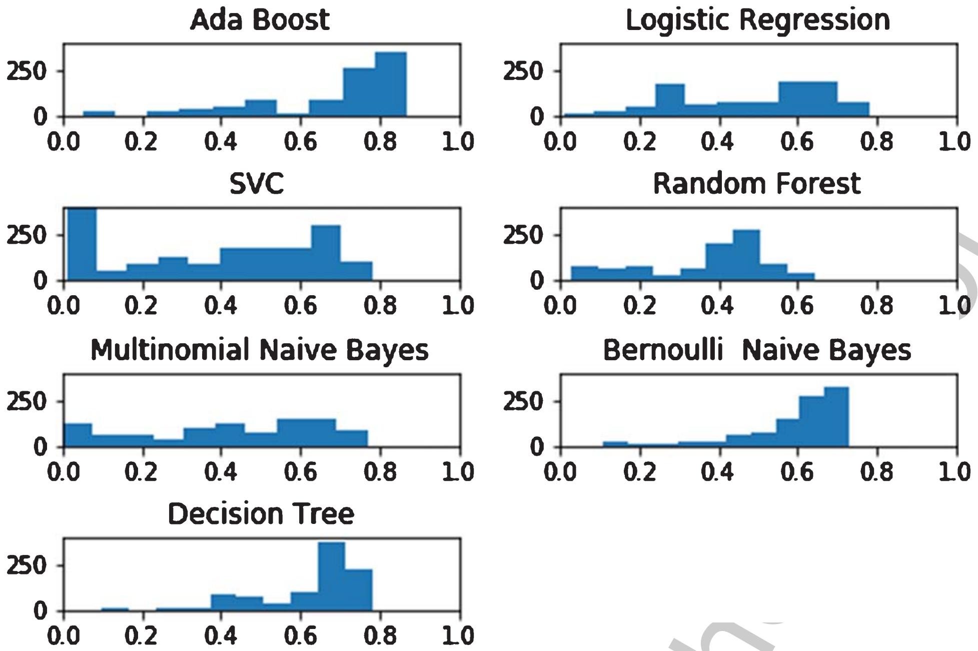
or both. it was also noticed during experiments that the classifiers without function words did not give better results compared with experiments using stop-words. Therefore, in the course of our experiment, stop-words played an important role.

Weighting Schemes. We also considered a number of weighting schemes for feature values. Figure 2 shows performance distribution for the weighting schemes. It can easily be noticed that certain weighting schemes are performing consistently better than the others. Others, such as TF (raw frequency) and TF-IDF are not showing good results. We observed that when using global weighting schemes such as TF-IDF and log-entropy, the classification performance decreases significantly in the majority of the experiments. Global weighting functions measure the importance of an n-gram across the entire collection of documents by reducing the weight of common features and highlighting the uncommon words. For the particular case of fake news detection, global weighting schemes majorly failed. It indicates that common words (character and function words n-grams) provide more relevant information to the classifier (see Tables 6 and 7).

From these distributions as well as from the best performing experimental combinations presented in Tables 6 and 7, we conclude that binary and normalized frequency weighting schemes are most useful in the majority of cases. However, as it can be seen from those tables, other weighting schemes are involved in highly performing runs for some feature combinations.

Classifiers. In the Figure 3, the performance of different classifiers during Our experiments has shown. Tables 6 and 7) results have displayed that some classifiers did a splendid work in differentiating legitimate news content, however, some classifiers presented poor performance. Particularly, we observe that classifiers such as *Adaboost* indicated consistently good performance across domains. DecisionTreeClassifier with maximum depth equal to 1 was chosen as a AdaBoost base. Similarly, bootstrap equal to true was selected as a particular parameter for Random forest trees. Linear kernel was tested with SVM only, because for the sake of time and space and we saw that the linear was enough for the baseline experiments.

Bayes based classifiers showed inferior results to AdaBoost, we haven't performed much analysis on the feature distribution (which is actually our future work and will be done in an upcoming paper). To our understanding, these both classifiers are suitable for


Ada Boost

Logistic Regression

SVC

Random Forest

Multinomial Naive Bayes

Bernoulli Naive Bayes

Decision Tree

Fig. 3. Classifiers Performance.

discrete data such as counts and we ran experiments on both binary features as well as counts to give best settings for each of the classifiers respectively. However, they still showed inferior results compared to AdaBoost in most of the experiments — beyond the paper scope

To gain further insights into the classes that are associated with fake and real content, we evaluate which classes show significant differences between the two groups of news.

The reflective observation from the experiments leads to the following conclusions:

- Combinations of different *n*-gram types obtain better results instead of single *n*-gram type.
- Combination of different feature sets with *n*-grams size 1 to 3 achieved significantly good results as compared with other feature sets.
- The *n*-gram size 1,2,3 achieve significantly better results compared with 4,5,6. However, in other languages, higher order *n*-grams performed well in detecting writing style [13].
- Feature sets with *n*-grams size 4 to 6 dropped the performance of all the classifiers either by combining feature sets or separately. This might be due to the limited dataset size.
- Adaboost algorithm is 87% percent accurate at detecting whether a news article is fake or not, when combining bi-gram characters and unigram function word *n*-grams.

6. Conclusions and future work

The paper concludes by arguing that the automatic detection of fake news is a promising area of research. In this research, a new resource for poor resource languages in the form of a dataset in Urdu language is presented and build a model that can correctly prognosticate the likelihood that a given news article is fake news. To our knowledge, this is the first corpus in the Urdu language for fake news detection, extracted from the internet and annotated manually containing real or fake news.

This is an essential contribution to the development of the Urdu corpus. Importantly, we provide statistics of the complete corpus, casts light on vocabulary size, vocabulary overlap and significant findings by the analysis. The present results confirm with the implementation of machine learning classifiers on lexical features BOW, *n*-grams (with *n* varying from 1 to 6), and in combination with *n*-grams methods. Overall, our results demonstrate the broad implication of the present research by obtaining promising results. On this basis, the main conclusion that can be drawn is that the addition of the new Urdu corpus is a particularly fruitful, reliable resource for the further development of fake news detection models.

In the future, we intend to explore whether the system can be adapted to other languages (it was trained exclusively on Urdu) and whether it can be trained to detect region-specific biases. We will also investigate new features to flag fake news.

Acknowledgments

The work was done with partial support of CONACYT project 240844 and SIP-IPN projects 20195719.

References

[1] C. Cieri, M. Maxwell, S.M. Strassel, J. Tracey, K. Choukri, T. Declerck, S. Goggi and M. Grobelnik, Selection Criteria for Low Resource Language Programs, in: *Proceedings of the 10th. International Conference on Language Resources and Evaluation*, LREC’2016), European Language Resources Association (ELRA), 2016.

[2] M. Potthast, J. Kiesel, K. Reinartz, J. Bevendorff and B. Stein, A Stylometric Inquiry into Hyperpartisan and Fake News, in: *Proceedings of the 56th. Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, (2018), pp. 231–240.

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

[4] V.L. Rubin, Y. Chen and N.J. Conroy, Deception Detection for News: Three Types of Fakes, in: *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*, American Society for Information Science, (2015), pp. 83.

[5] N.J. Conroy, V.L. Rubin and Y. Chen, Automatic Deception Detection: Methods for Finding fake News, *Proceedings of the Association for Information Science and Technology* **52**(1) (2015), 1–4.

[6] D.M. Nguyen, T.H. Do, R. Calderbank and N. Deligiannis, Fake News Detection using Deep Markov Random Fields, in: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, (2019), pp. 1391–1400.

[7] V. Pérez-Rosas, B. Kleinberg, A. Lefevre and R. Mihalcea, Automatic Detection of Fake News, in: *Proceedings of the 27th International Conference on Computational Linguistics, Association for Computational Linguistics*, Santa Fe, New Mexico, USA, (2018), pp. 3391–3401. https://www.aclweb.org/anthology/C18-1287.

[8] M. Aldwairi and A. Alwahedi, Detecting Fake News in Social Media Networks, *Procedia Computer Science* **141** (2018), 215–222. https://linkinghub.elsevier.com/retrieve/pii/S1877050918318210.

[9] Y. Chen, N.J. Conroy and V.L. Rubin, Misleading online content: Recognizing clickbait as false news, in: *Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection, ACM*, (2015), pp. 15–19.

[10] T.H. Nazer, G. Xue, Y. Ji and H. Liu, Intelligent Disaster Response via Social Media Analysis A Survey, *ACM SIGKDD Explorations Newsletter* **19**(1) (2017), 46–59.

[11] V. Ivanov and E. Tutubalina, Clause-Based Approach to Extracting Problem Phrases from User Reviews of Products, (2014), pp. 229–236.

[12] S. Afroz, M. Brennan and R. Greenstadt, Detecting hoaxes, frauds, and deception in writing style online, in: *2012 IEEE Symposium on Security and Privacy*, IEEE, 2012, pp. 461–475.

[13] J.P. Posadas-Durán, H. Gómez-Adorno, G. Sidorov and J. Jaime Moreno Escobar, Detection of Fake News in a New Corpus for the Spanish Language, *Journal of Intelligent & Fuzzy Systems* (2018).

[14] R. Baly, G. Karadzhov, A. Saleh, J. Glass and P. Nakov, Multi-Task Ordinal Regression for Jointly Predicting the Trustworthiness and the Leading Political Ideology of News Media, *arXiv preprint arXiv:1904.00542* (2019).

[15] W. Ferreira and A. Vlachos, Emergent: a Novel Data-Set for Stance Classification, in: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL’2016*, (2016), pp. 1163–1168.

[16] P. Krejzl, B. Hourová and J. Steinberger, Stance Detection in Online Discussions, *arXiv preprint arXiv:1701.00504* (2017).

[17] B. Sean, S. Doug and Y. Pan, Talos targets disinformation with fake news challenge victory, https://blog.talosintelligence.com/2017/06/ (2017).

[18] T. Mikolov, K. Chen, G. Corrado and J. Dean, Efficient estimation of word representations in vector space, *arXiv preprint arXiv:1301.3781* (2013).

[19] W.Y. Wang, ” liar, liar pants on fire”: A new benchmark dataset for fake news detection, *arXiv preprint arXiv:1705.00648* (2017).

[20] T. Mitra and E. Gilbert, Credbank: A large-scale social media corpus with associated credibility annotations, in: *Ninth International AAAI Conference on Web and Social Media*, 2015.

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

[22] E.V. Pronoza, E. Yagunova and A. Pronoza, A New Corpus of the Russian Social Network News Feed Paraphrases: Corpus Construction and Linguistic Feature Analysis, in: *Advances in Computational Intelligence - 16th Mexican International Conference on Artificial Intelligence*, MICAI’2017, (2017), pp. 133–145.

[23] P. Baker, A. Hardie, T. McEnery, H. Cunningham and R. Gaizauskas, EMILLE, A 67 Million Word Corpus of Indic Languages: Data Collection, Mark-up and Harmonisation, in: *Proceedings of the 3rd. Language Resources and Evaluation Conference*, (2002), pp. 819–825.

[24] A. Saeed, R.M.A. Nawab and M. Stevenson, A Word Sense Disambiguation Corpus for Urdu (2018).

[25] A. Muaz, A. Ali and S. Hussain, Analysis and Development of Urdu POS tagged corpus, in: *Proceedings of the 7th. Workshop on Asian Language Resources, IJCNLP*, (2009), pp. 24–29.

[26] A.A. Raza, S. Hussain, H. Sarfraz, I. Ullah and Z. Sarfraz, Design and Development of Phonetically Pich Urdu Speech Corpus, in: *2009 Oriental COCOSDA International Conference on Speech Database and Assessments*, IEEE, (2009), pp. 38–43.

[27] W.Y. Wang, "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection, *arXiv preprint arXiv:1705.00648* (2017).

[28] H. Gómez-Adorno, J.-P. Posadas-Duran, G. Ríos-Toledo, G. Sidorov and G. Sierra, Stylometry-based approach for detecting writing style changes in literary texts, *Computación y Sistemas* **22**(1) (2018), 47–53.

[29] B. Pang, L. Lee and S. Vaithyanathan, Thumbs up?: sentiment classification using machine learning techniques, in: *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, Association for Computational Linguistics, (2002), pp. 79–86.

[30] R.A. Horn and C.R. Johnson, Norms for vectors and matrices, *Ch. 5 in Matrix analysis* (1990), 313–386.

[31] T.K. Landauer, LSA as a theory of meaning, in: *Handbook of latent semantic analysis*, Psychology Press, (2007), pp. 15–46.

[32] G. Sidorov, S. Miranda-Jiménez, F. Viveros-Jiménez, A. Gelbukh, N. Castro-Sánchez, F. Velásquez, I. Díaz-Rangel, S. Suárez-Guerra, A. Trevino and J. Gordon, Empirical Study of Machine Learning based Approach for Opinion Mining in Tweets, in: *Mexican international conference on Artificial intelligence*, MICAI'2012, Springer, (2012), pp. 1–14.

[33] E. Stamatatos, A Survey of Modern Authorship Attribution Methods, *Journal of the American Society for information Science and Technology* **60**(3) (2009), 538–556.

[34] B. Pang and L. Lee, Opinion Mining and Sentiment Analysis, *Foundations and Trends® in Information Retrieval* **2**(1–2) (2008), 1–135.

[35] L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt and G. Varoquaux, API Design for Machine Learning Software: Experiences from the Scikit-learn Project, in: *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, (2013), pp. 108–122.