IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 31, 2020, accepted August 24, 2020, date of publication October 1, 2020, date of current version October 22, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.3028131*

# ProSOUL: A Framework to Identify Propaganda From Online Urdu Content

**SOUFIA KAUSAR, BILAL TAHIR, AND MUHAMMAD AMIR MEHMOOD**
Al-Khawarizmi Institute of Computer Science, University of Engineering and Technology, Lahore 54890, Pakistan

Corresponding author: Muhammad Amir Mehmood (amir.mehmood@kics.edu.pk)

This work was supported in part by the Higher Education Commission (HEC) Pakistan, and in part by the Ministry of Planning Development and Reforms under National Center in Big Data and Cloud Computing.

**ABSTRACT** Today, the rapid dissemination of information on digital platforms has seen the emergence of information pollution such as misinformation, disinformation, fake news, and different types of propaganda. Information pollution has become a serious threat to the online digital world and has posed several challenges to social media platforms and governments around the world. In this article, we propose **Pro**paganda **S**potting in **O**nline **U**rdu **L**anguage (**ProSOUL**) - a framework to identify content and sources of propaganda spread in the Urdu language. First, we develop a labelled dataset of 11,574 Urdu news to train the machine learning classifiers. Next, we develop the Linguistic Inquiry and Word Count (LIWC) dictionary to extract psycho-linguistic features of Urdu text. We evaluate the performance of different classifiers by varying n-gram, News Landscape (NELA), Word2Vec, and Bidirectional Encoder Representations from Transformers (BERT) features. Our results show that the combination of NELA, word n-gram, and character n-gram features outperform with 0.91 accuracy for Urdu text classification. In addition, Word2Vec embedding outperforms BERT features in classification of the Urdu text with 0.87 accuracy. Moreover, we develop and classify large scale Urdu content repositories to identify web sources spreading propaganda. Our results show that ProSOUL framework performs best for propaganda detection in the online Urdu news content compared to the general web content. To the best of our knowledge, this is the first study on the detection of propaganda content in the Urdu language.

**INDEX TERMS** Information bias, information pollution, low resource language, propaganda detection.

## I. INTRODUCTION

Recent developments in artificial intelligence, big data, and natural language generation are a double-edged sword. On one hand, applications like text summarization [1], chatbots [2], and automated journalism [3] are assisting humans. On the other hand, these technologies have become effective tools for the generation and dissemination of misinformation. The growth of misinformation in online content and its amplification by social media platforms are posing several critical challenges to society. For instance, fake news and various propaganda techniques are serious threats to democracy [4], journalism [5], health [6], economy [7], and climate change [8]. In general, the propaganda is an expression of opinion or action by individuals or groups deliberately designed to influence the opinions or actions of other individuals or groups concerning predetermined ends [9]. The term *propaganda* is frequently confused with lies, distortion, and deceit [10] but any content with biased messages either intentional or unintentional is propaganda [11]. In literature, the techniques used to spread propaganda are categorized into seven classes [12]. For example, name calling labels the individuals or group with bad names and card stacking method falsifies the facts to overemphasize the agenda. The detailed description of different propaganda techniques is provided in Table 1.

The ramifications of propaganda in the United States of America (USA) elections is a prime example of its impact on societies [13], [14]. The propaganda disseminated by Cambridge Analytica (CA) [15] and Internet Research Agency (IRA) [16] through Facebook shaped the political attitude of citizens to manipulate election results. Similarly, online propaganda has affected the foreign policies of European countries [17]. The conspiracy theories linking 5G technology to coronavirus (COVID-19) pandemic have led to violent riots [18], [19]. This phenomenon is not confined by

The associate editor coordinating the review of this manuscript and approving it for publication was Noor Zaman.

**TABLE 1.** Types of propaganda techniques.

| | Techniques | Description |
|---|---|---|
| 1 | **Card stacking** | Card stacking technique over/under emphasizes the facts by omitting or falsifying the truth. |
| 2 | **Name calling** | Name calling is labelling of individual or group with names, i.e., fascist or terrorist based on their beliefs, nation, or races. |
| 3 | **Glittering generalities** | Glittering generalities use 'abstract words' such as patriotism, rights, or freedom to appeal the emotion of the audience. |
| 4 | **Transfer** | Transfer propaganda highlights the positive or negative qualities of one individual, group, or companies to make second more acceptable. |
| 5 | **Testimonial** | Testimonial technique uses quotes or endorsements from celebrities to strengthen the argument. |
| 6 | **Plain folks** | In plain folks propaganda, speaker wins the confidence of the audience by appearing as a common person. |
| 7 | **Bandwagon** | Bandwagon method groups the target audience with common ties to persuade them to join in and take the course of action that 'everyone else is taking'. |

any specific language. Accordingly, propaganda in regional languages is disseminated by extremist groups to sway the local population towards violent crimes [20], [21]. Anti-state Propaganda by these extremist groups has also threatened national security [22] of some countries. The propaganda news against polio vaccine was published in 100 local Urdu newspapers, resulting in an increased number of polio cases in the propaganda affected areas [23].

The research community has recently started using artificial intelligence tools for the automatic detection of propaganda content. Different systems are developed for automatic propaganda detection by using state-of-the-art classification algorithms. The neural architectures with Bidirectional Encoder Representations from Transformers (BERT) embeddings [24] are used for the detection of propaganda news [25]. The fine-grained analysis of news text is done to identify sentences from content used for propaganda. In another effort, Proppy [26] uses representations of text style, i.e., readability and comprehension, etc., along with textual features for the identification of propaganda news. However, the current development of systems focuses mainly on English language content. In addition, the datasets used in these studies are developed by fetching content from labelled propaganda news sources. The online service of Media Bias/Fact Check (MBFC) [27] manually labels the propaganda news sources according to the biasness of the published content. However, this service labels those sources that are publishing content in the English language only. The adaptation of these methodologies for the content based on the regional languages is strenuous due to the requirement of large scale datasets and essential linguistic resources. Therefore, there is an imperative need to develop systems for low resource languages using transfer learning to avoid the resource scarcity problem.

There are 170 million people in the world who speak Urdu language [28]. As such, no automatic system is developed so far to identify and mitigate the effect of propaganda disseminated in the Urdu language. In this article, we present **Pro**paganda **S**potting in **O**nline **U**rdu **L**anguage (ProSOUL) framework to identify propaganda content in the Urdu language. First, we develop a labelled dataset of Urdu propaganda news containing 11,574 news articles. Next, we develop the Linguistic Inquiry and Word Count (LIWC) for Urdu by translating the English [29] dictionary for psycho-linguistic features. We train and evaluate the performance of state-of-the-art classifiers with n-gram, NEws LAndscape (NELA) [30], Word2Vec [31], and BERT [24] features. Our results describe the best performance of the classifier for Urdu text classification with character n-gram, word n-gram, and NELA features with 0.91 accuracy. However, these features perform poorly for English text with 0.65 accuracy. In addition, Word2Vec performs better for Urdu text compared to BERT features with 0.87 accuracy. We also evaluate the performance of classifiers for classification of test data acquired from sources unseen in training data to penalize the models learning only training data as opposed to actual learning of propaganda features. Our evaluation shows the failure of n-gram features in the classification of data from unseen sources compared to NELA, BERT, and Word2Vec features. Moreover, we identify and assign propaganda scores to web sources disseminating propaganda in Urdu by classifying two large scale repositories of Humkinar-Web [32] and Humkinar-News [33]. We build Humkinar-Web by crawling Urdu content from the World Wide Web (WWW) and Humkinar-News by scraping the news content from manually selected Urdu news websites. The analysis shows that ProSOUL performs best for the propaganda content present in the online news content compared to general web content. Furthermore, our results highlight the need for a dataset from earmarked domains for propaganda detection. Our main contributions in this article are as follows:

1) We develop a labelled dataset of Urdu propaganda news consists of 11,574 news articles. Also, we develop Linguistic Inquiry and Word Count (LIWC) dictionary for Urdu to extract linguistic features of Urdu text.
2) We experiment thoroughly with different variations of classifiers with n-gram, NELA, Word2Vec, and BERT features for English and Urdu text. Our results show that classifier performance can be boosted for the translated text due to simplification of semantic relations by translating algorithms.
3) Our experiment of Urdu text classification on data from unseen sources shows that NELA features can generalize better than currently used n-gram approach.

4) The comparison of BERT and Word2Vec embeddings for English and Urdu text classification describes that performance of these embeddings depends on their training vocabulary. Word2Vec outperforms BERT for the Urdu text classification due to richer vocabulary.
5) We classify the real-world data with ProSOUL to evaluate the performance of our propaganda identification classifier. For testing, we develop two large scale repositories of Humkinar-Web and Humkinar-News by crawling the world wide web. Humkinar-Web and Humkinar-News contain 6.4 and 0.62 million Urdu documents, respectively.
6) We introduce propaganda scoring method for the identification and scoring of online Urdu websites disseminating propaganda. The evaluation of our method on Humkinar-Web and Humkinar-News describes that ProSOUL performs better for online news compared to the general web content. Hence, emphasizing the need for labelled datasets from target domains for propaganda detection.

The rest of the paper is organized as follows: Section II presents related work and Section III describes the dataset. In section IV, we present ProSOUL framework developed for the classification of propaganda content. Classification performance of ProSOUL is reported in Section V. Next, we describe the process of developing Urdu content repositories and propaganda scoring of websites in Section VI. Finally, we conclude our paper in Section VII.

## II. RELATED WORK

Over the years, researchers have studied the general aspects of fake and deceptive information in online content. This includes studying the impact of misinformation on politics [4], [15], [16], society [5], [8], economy [7], and health [18]. Different machine learning algorithms have been developed to automatically detect false news [34]–[37], propaganda [26], [38], [39], toxicity [40], and fact checking [41] from the online web content.

In general, there have been efforts on the automatic detection of misinformation by using simple and advanced linguistic features. The simplistic approach of performing probabilistic matching using Naive Bayes classifier was applied on social media content for the fake news detection with 74% accuracy [37]. Linguistic features such as pronouns, cognitive, emotion, and function words were used for the automatic identification of fake news with 76% accuracy [34]. Another similar study identifies true news from the set of propaganda, satire, and hoax using n-gram features [38]. In this study, the authors performed four-way classification on their Trusted, Satire, Hoax, and Propaganda (TSHP) corpus using maximum entropy classifier. Their results reveal that word n-gram being the topic-dependent feature did not perform well on out-of-domain articles.

Appropriate feature selection for misinformation classification is a challenging task. To address this issue, complex models have been developed by using various combinations of linguistic and user behavioral features. For instance, *Proppy* identified propaganda content from online English news by using a combination of n-gram and NELA linguistic features [26]. In Proppy, the evaluation of maximum entropy classifier with labelled dataset showed that the classifier performed better by including stylistic features of NELA for the propaganda detection. In another approach, the hybrid model of Capture, Score, and Integrate (CSI) [35] used a combination of multiple features such as article text, user response, and source users to detect the fake news. Here, two independent models were developed using text and user features for the classification purpose. Long [36] used a similar approach by integrating two independent Long Short Term Memory (LSTM) models for the user and textual features. In general, hybrid models improved the accuracy of fake news detection by 14.5%.

In literature, pre-trained word embeddings have been used as an effective tool for the detection of propaganda content. Gupta *et al.* [25] used Part of Speech (POS) tags, readability measure, sentiment, topic, emotion, and word embeddings as features for the identification of propaganda content. These features provided the accuracy of 66.9% and 16.4% for sentences and fragments classification, respectively. In addition, a combination of title and document text was used to make context-dependent input pairs to fine-tune BERT embedding to perform fine-grained propaganda detection [39]. Beside automatic feature extraction, the combination of handcrafted text complexity and word embedding features was used to identify sentences containing various kind of propaganda [42].

While the most developed systems focused on the English language content, little progress is made for low resource languages. Baly *et al.* [41] predicted the *factuality* of claims in Arabic text by determining the stance of multiple documents concerning the claim. In this study, hand-crafted features reflecting polarity, refute, similarity, and overlap between the documents were used to achieve the accuracy of 80%. A major obstacle in low resource text document processing is the scarcity of standard and public datasets. To evade the problem, English language resources and datasets were translated to low resource languages. For instance, news toxicity detector was developed by translating English news text to Bulgarian language [40]. The toxicity detector showed the accuracy of 59% with stylometric, NELA, and word embedding features. For Urdu language, Amjad *et al.* [43] used a similar approach to develop fake news dataset by translating the labelled English text. However, translation error highly impacted the classification accuracy of classifiers for the Urdu text. Beside datasets, LIWC dictionary was also translated using Google translate to extract linguistic features of Dutch language [44]. In this study, stem words were converted to fix words with an English dictionary and the quality of the translated text was measured by comparing the automatic and manually translated dictionary. Similarly, the LIWC dictionary was translated into the Filipino language

**TABLE 2.** Dataset statistics.

| Class | Sources | Articles | Train | Test | Avg. token per article | | Token count | |
|---|---|---|---|---|---|---|---|---|
| | | | | | English | Urdu | English | Urdu |
| **Propaganda** | 10 | 5,322 | 3,725 | 1,597 | 935.07 | 1,338.03 | 241,582 | 128,168 |
| **Non - Propaganda** | 94 | 6,252 | 4,376 | 1,876 | 558.28 | 733.57 | 197,289 | 99,666 |
| **Total** | **104** | **11,574** | **8,101** | **3,473** | **731.51** | **1011.4** | **352,510** | **184,221** |

to perform sentiment analysis of Filipino text [45]. In another research effort, the manual analysis of English to Arabic text was performed to check the quality of translated content [46]. Their analysis on 100 randomly selected pages for semantic, grammatical, and syntactic errors showed that the text contains 20.3% inaccurately translated words.

From the above discussion, we conclude that systems are developed for identification of propaganda from English news content. Moreover, English language resources and datasets are translated to build systems for low resource languages. Leveraging previous research efforts, we propose to detect propaganda content in the Urdu language. Compared to the similar work of Proppy [26], we also use n-gram and NELA features for the classification of Urdu text. Furthermore, we analyze the contribution of each individual and combination of NELA features for the Urdu content. We also evaluate and compare the performance of state-of-the-art Word2Vec and BERT embeddings for English and Urdu text classification.

## III. DATASET

ProSOUL trains machine learning classifiers with different linguistic patterns from text to identify propaganda content. To learn different patterns, a reference point of annotated propaganda articles is required. However, any labelled dataset of propaganda content in the Urdu language is not available. Therefore, we develop our labelled dataset[1] by translating the English dataset of *QCRI's propaganda (Qprop)* [26] to the Urdu language. We translate this gold standard dataset because the process of labelling a dataset is technically challenging, labour intensive, and a time-consuming task that often suffers from data sparseness. Qprop was developed by collecting news from various propaganda news sources that were manually labelled by the Media Bias/Fact Check (MBFC) service [27]. MBFC relies on volunteers to score news sources based on their biasness. News sources with propaganda content are flagged separately by volunteers. 94 news sources labelled as *trustworthy* were used to collect non-propaganda and 10 news sources labelled as *propaganda* were used to collect news related to the propaganda class.

After identifying propaganda and non-propaganda news sources, news published by these sources were fetched from the open-source electronic news content repository of Global Database of Events, Language, and Tone (GDELT) [47]. Qprop contains 45,557 non-propaganda and 5,737 propaganda news. Qprop contains highly imbalanced dataset with 11.2% data of propaganda and 88.8% from non-propaganda class. However, for the dataset in the Urdu language, we create a dataset by translating randomly selected 5,322 propaganda and 6,252 non-propaganda news to the Urdu language. We use the online service of Google Translate [48] to translate dataset. The developed dataset contains 184,221 unique tokens. For English news classification, we use the English content of translated pages with the same distribution of 6,252 non-propaganda and 5,322 propaganda news. The details of the datasets are given in Table 2.

In general, machine translation can be used to develop labelled dataset without human assistance. However, machine translation systems are inept to handle substitute words, complex linguistic knowledge, syntactic, and semantic relations. This fact is more eminent when languages with two orthographies like English and Urdu are translated [46]. To ensure the quality of our dataset, we manually analyze 50 translated articles from propaganda and non-propaganda class. We calculate semantic, contextual, translation, and transliteration errors to predict their impact on ProSOUL performance. Semantic error identifies words with ambiguous or different meaning when translated in the Urdu language. In addition, unnecessary, unfamiliar, and incorrectly translated words of English homonyms corrupting the meaning of sentences are also considered as semantic error. Next contextual error is calculated by finding the sentences with altered sense due to misplacement of translated words. Similarly, translation error is calculated by identifying non-translated words. Finally, the words transliterated by Google but not commonly used in Urdu vocabulary are considered as transliteration error. We calculate the percentage of tokens with error for semantic, translation, and transliteration error. The contextual error alters the sentences, hence, we calculate the percentage of sentences with the contextual error. Table 3 shows the percentage of these errors in our translated dataset. The analysis describes the presence of less than one percent of tokens with errors. Moreover, 95.46% of sentences are translated with precise contextual structure.

## IV. ProSOUL FRAMEWORK

In this section, first, we describe the process for developing the LIWC dictionary for Urdu. Next, we explain techniques applied for data pre-processing and feature extraction. Finally, we briefly introduce the machine learning classifiers trained to identify propaganda content.

[1] https://github.com/Bilaltahir098/ProSOUL

**TABLE 3.** Percentage errors in the translated dataset.

| | Articles | Tokens | Sentences | Semantic | Contextual | Translation | Transliteration |
|---|---|---|---|---|---|---|---|
| **Propaganda** | 25 | 25,514 | 966 | 0.10% | 4.45% | 0.09% | 0.07% |
| **Non - Propaganda** | 25 | 27,211 | 1,036 | 0.11% | 4.62% | 0.12% | 0.08% |
| **Total** | **50** | **52,752** | **2,002** | **0.11%** | **4.54%** | **0.11%** | **0.07%** |

**TABLE 4.** LIWC dictionary statistics.

| Category | Standard linguistic dimensions | Psychological processes | Personal concern | Spoken categories |
|---|---|---|---|---|
| **Examples** | *walk, have, were, it* | *brother, cry, ugly, pill* | *goal, music, cash, bury* | *agree, I mean, uh* |
| **English** | 2,070 | 7,244 | 1,229 | 52 |
| **Urdu** | 1,138 | 11,644 | 2,614 | 16 |

### A. LINGUISTIC INQUIRY AND WORD COUNT (LIWC)

This is hardly a novel observation that individuals differ in writing patterns. Even for the content with the same message, people express themselves in distinctive ways [49]. In psychology, the language used is influenced by the underlying emotional or cognitive states. This ultimately results in different word choices. From a theoretical perspective, for propaganda detection, there is a need for a tool to understand the deeper meaning present in the communication by analyzing these word choices. The Linguistic Inquiry and Word Count (LIWC) [29] is a system proposed for the psycho-linguistic analysis of the text. LIWC uses an internal lexicon to classify words into one or more linguistic, psychological, and social processes categories. The sample words from four main categories of standard linguistic dimensions, psychological processes, personal concern, and spoken categories are shown in Table 4. LIWC counts words based on dictionary files. These files contain words in a hierarchical structure with categories and sub-categories. For example, in the standard LIWC dictionary there is a category named 'affect', which has *posemo* and *negemo* as sub-categories, with the latter including the 'sad', 'anxiety', and 'anger' sub-categories. This structure and the words to be included in each category are created from psychological concepts with the help of judges. LIWC is used to classify narcissism [50], emotion [51], fake news [52], and propaganda [26] from the textual content. As LIWC can be used with customized dictionaries, the translated version of dictionaries are developed for linguistic analysis in French [53], Spanish [54], Chinese [55], and Portuguese [56] languages.

Due to the unavailability of LIWC for the Urdu language, we develop a dictionary by translating the original English LIWC dictionary [29]. The English dictionary contains 2,149 fixed and 2,338 stem words belonging to 64 sub-categories like pronoun, number, swear, social, positive words (posemo), and negative words (negemo). First, we expand the stem words to fixed words by extracting the matching terms from the English dictionary [57] with a rich vocabulary of 466,551 words. The expansion of 2,338 English stem words results in 28,580 fixed words. The list of total 30,729 fixed words is translated using online Google Translate [48] service. The translation of English keywords generates 27,020 Urdu and 3,709 non-translated English words. We remove non-translated words from the dictionary. Besides, multiple English keywords are translated into one Urdu keyword. The keywords of *abandoned* and *abandonedly* are translated to one Urdu word of ترک کرنا *(Trk Krna)*. In order to remove duplicates from the dictionary, we assign different sub-categories of English words to one translated Urdu word. Also, we discard any multi-word expressions returned by the online translation service in a similar vein [44]. Our final dictionary of Urdu LIWC contains 7,490 unique words belonging to 62 sub-categories. The example words from Urdu LIWC dictionary are given in Table 5.

We manually analyze the quality of translated LIWC by examining randomly selected translated words. In this article, we utilize the vocabulary of three categories *Emotion, Cognitive words*, and *Topic specificity*. Therefore, we analyze 300 words from each category. We ensure the unbiased selection of words by selecting a word from each sub-category according to their word frequency. For example, sub-categories of emotion named posemo and negemo contain 633 and 684 words, respectively. According to their word frequency, we select 144 posemo and 156 negemo words. In addition, we distribute words into *effective* and *ineffective* vocabulary. The effective vocabulary contains translated and transliterated words that are commonly used in Urdu documents while ineffective vocabulary contains non-existing transliterated words. The translated words in effective vocabulary are correctly translated and labelled words. Similarly, the words transliterated to commonly used Urdu vocabulary are labelled as transliterated words. The mis-labelled class contains translated words with a different meaning in Urdu and cannot be assigned the same label. The result of the analysis is presented in Table 6. Our analysis of LIWC translation shows the presence of 29-40% words from the ineffective vocabulary. However, such words cannot introduce errors in the classification because these words are never used in Urdu documents. Moreover, the effective vocabulary consists of

**TABLE 5.** Urdu LIWC dictionary examples.

| Words | Words (Roman) | Categories | | | | |
|---|---|---|---|---|---|---|
| لاوارث | La waris | affection | negative emotion | sad | cognitive | inhibition |
| درد | Dard | affection | negative emotion | sad | bio | health |
| تم ہو | Tum ho | function | personal pronoun | pronoun | you | - |
| قبولیت | Qabooliat | affection | positive emotion | cognitive | insight | - |
| تسلیم | Tasleem | verb | present | social | coginitive | insight |
| دنیا | Duniya | space | relativity | - | - | - |
| اچانک | Achanak | time | relativity | - | - | - |

**TABLE 6.** Translated LIWC dictionary error analysis.

| Category | Total words | Effective vocabulary | | | Ineffective vocabulary |
|---|---|---|---|---|---|
| | | Translated | Mis-labelled | Transliterated | |
| **Topic specificity** | 300 | 28.3% | 10.3% | 21.6% | 39.6% |
| **Emotion** | 300 | 43.6% | 12.6% | 6.3% | 37.3% |
| **Cognitive words** | 300 | 47.0% | 10.0% | 14.0% | 29.0% |
| **Total** | 900 | 39.6% | 11.0% | 14.0% | 35.4% |

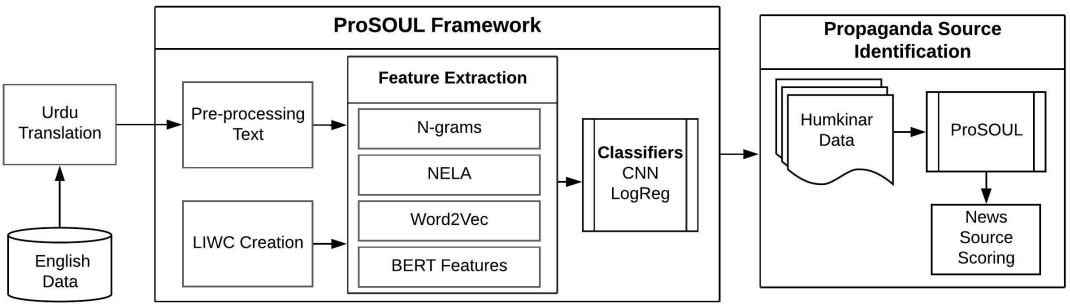



**FIGURE 1.** ProSOUL framework – Architecture.

only 10-12% Urdu words that cannot be assigned the same label after translation. Among emotion, cognitive words, and topic specificity features, we found that cognitive words are least sensitive to errors due to translation.

### B. PRE-PROCESSING AND FEATURE EXTRACTION

The architecture of ProSOUL framework is shown in Figure 1. First, we pre-process the content by removing URLs, non-Urdu words, and stopwords from news text. For this purpose, a pre-defined list of 496 Urdu stop-words is used for text cleaning [58]. This list consists of very frequent words that do not carry any meaning of their own. After pre-processing, we convert the text to machine-understandable vector by extracting character and word level Term Frequency-Inverse Document Frequency (TF-IDF) [59] features. We extract uni-gram, bi-gram, and tri-gram representation of words and characters. TF-IDF features normalize the occurrence and importance of n-gram features. These features are selected because ngram features bring structural information about the sentence sequence, whereas the TF-IDF features bring more refinement for the rare and meaningful terms. We also extract the content based NEws LAndscape (NELA) [30] features to measure stylistic and psycho-linguistic aspects of a news article. NELA contains 130 features categorized into six groups: (i) writing style and complexity, (ii) sentiment and emotion, (iii) LIWC psychology, (iv) topic specific, (v) bias, and (vi) morality. Table 7 provides a description of these groups. We note that for the implementation of NELA features various linguistic resources are required. For example, for readability [60] calculation, the list of syllables is required. However, we could not extract all NELA features due to the unavailability of such linguistic resources in the Urdu language. In particular, we extract the average word length (AvgLen), word count (WordCount), emotion bias (Emotion), number of cognitive process words (Cognitive), topic specificity (TopicSp), and Type-Token Ratio (TTR) from the text using LIWC dictionary. First two features, i.e., AvgLen and

**TABLE 7.** Description of NELA features.

| | Feature category | Description |
|---|---|---|
| 1 | **Writing style and complexity** | swear words, functions words (from LIWC) , average word length, and word count |
| 2 | **Sentiment and emotion** | positive emotion, negative emotion, sad, or anger words (all from LIWC) |
| 3 | **LIWC psychology** | cognitive process words, social words, and achievement words (LIWC) |
| 4 | **Topic specific** | religious, money, time, space, health, and work words (LIWC) |
| 5 | **Bias** | bias lexicons, negative, and positive opinion |
| 6 | **Morality** | features from moral foundation theory, i.e., harm, loyalty, cheating, purity |

WordCount represent the complexity of text whereas emotion and cognitive features extract the bias introduced in the text. Next feature, TopicSp finds the topic of discussion in the text by counting the presence of labelled words using LIWC dictionary. LIWC contains words from a variety of topic including religious, money, time, space, health, and work words etc. Finally, Type-Token Ratio (TTR) [61] is used to measure the vocabulary richness and lexical diversity of the text.

We experiment with word embedding features of pre-learned Word2Vec [62] model. As such, word embedding models *embed* words into a high-dimensional space, representing them as dense vectors of real numbers. Vectors close to each other according to a distance function represent semantically related words. Also, word embeddings have an edge of context-aware learning as compared to n-gram representation. We use a continuous bag of words (CBOW) Word2Vec model with rich vocabulary of 132,246,587 Urdu tokens. In addition, we extract the word embeddings generated using Bidirectional Encoder Representations from Transformers (BERT) [24] which is a bidirectional attention model. BERT captures the context of a word by considering its position and order of words in a sentence. There are several configurations in BERT model to implement various architectures. However, we extract word embeddings of the pre-trained model of BERT-Base. This model contains 12 layers of transformer block with 12 attention layers of each block. Each block contains 768-dimensional hidden layers resulting in 110M total parameters of the model.

### C. MACHINE LEARNING CLASSIFIERS

We train and evaluate the performance of state-of-the-art Logistic Regression (LogReg) classifier [63] with n-gram, NELA, and Word2Vec features. Logistic regression is a statistical and linear classifier which aims to maximize the quality of its predictions using the logistic function to construct a 'map' between textual features and text class. We use *liblinear* as logistic regression solver and L2 for regularization. Also, convolutional neural network (CNN) [64] is trained for the classification of content with BERT embeddings. CNN consists of interconnected layers of artificial neurons that learn the training data by adjusting the weights. Adam optimizer with a dropout rate of 0.2 and Rectified Linear Unit (ReLU) as the activation function are used to train CNN model.

Next, we briefly discuss standard metrics of accuracy, macro precision, macro recall, and macro F-measure used to evaluate the trained classifiers performance.

- **Accuracy:** It calculates the percentage of correctly classified samples out of the total data samples.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

- **Macro precision:** It calculates the per-class average of the correctly classified true positive instances across both the true positive and false positive data samples.

$$Precision_M = \frac{1}{n}(\sum_{i=1}^{n} \frac{TP_i}{TP_i + FP_i}) \tag{2}$$

- **Macro recall**: It computes the per-class average of the truly classified instances across both true positive and true negative samples.

$$Recall_M = \frac{1}{n}(\sum_{i=1}^{n} \frac{TP_i}{TP_i + FN_i}) \tag{3}$$

- **Macro F-measure:** F-measure also called F1-score is a harmonic mean between precision and recall.

$$F_M = \frac{2 * Precision_M * Recall_M}{Precision_M + Recall_M} \tag{4}$$

The interpretation of terms used in evaluation metrics is as follows:

$\boldsymbol{n}$ = Number of classes in dataset
**True Positive (*TP*)** = Actual propaganda sample predicted as propaganda class
**False Positive (*FP*)** = Actual non-propaganda sample predicted as propaganda class
**True Negative (*TN*)** = Actual non-propaganda sample predicted as non-propaganda class
**False Negative (*FN*)** = Actual propaganda sample predicted as non-propaganda class

### D. IMPLEMENTATION

The implementation of pre-processing, feature extraction, and classification modules is done in Python [65] programming language. The library of Natural Language ToolKit (NLTK) [66] is used for stopword removal. Similarly, the implementation of classification models is done using Python language library - Sklearn [67]. Word2Vec features are extracted using gensim [68] library. The implementation

**TABLE 8.** Classifier performance for n-gram and NELA features.

| | | English | Urdu | | | | | |
|---|---|---|---|---|---|---|---|---|
| Top Features | Measure | All | C-ngram | W-ngram | C-ngram + NELA | W-ngram + NELA | W-ngram + C-ngram | All |
| 3 | Accuracy | 0.58 | 0.56 | 0.65 | 0.76 | 0.76 | 0.68 | 0.76 |
| | Precision | 0.57 | 0.73 | 0.67 | 0.75 | 0.76 | 0.69 | 0.75 |
| | Recall | 0.56 | 0.52 | 0.66 | 0.74 | 0.75 | 0.69 | 0.75 |
| | F-measure | 0.55 | 0.39 | 0.65 | 0.75 | 0.75 | 0.68 | 0.75 |
| 10 | Accuracy | 0.59 | 0.58 | 0.69 | 0.75 | 0.77 | 0.71 | 0.77 |
| | Precision | 0.58 | 0.67 | 0.68 | 0.75 | 0.76 | 0.71 | 0.76 |
| | Recall | 0.57 | 0.53 | 0.69 | 0.74 | 0.76 | 0.71 | 0.76 |
| | F-measure | 0.56 | 0.44 | 0.69 | 0.74 | 0.76 | 0.71 | 0.76 |
| 100 | Accuracy | 0.62 | 0.71 | 0.80 | 0.76 | 0.82 | 0.81 | 0.82 |
| | Precision | 0.61 | 0.71 | 0.80 | 0.77 | 0.82 | 0.80 | 0.82 |
| | Recall | 0.61 | 0.71 | 0.80 | 0.75 | 0.82 | 0.80 | 0.82 |
| | F-measure | 0.60 | 0.71 | 0.80 | 0.76 | 0.82 | 0.80 | 0.82 |
| 500 | Accuracy | 0.64 | 0.79 | 0.86 | 0.81 | 0.87 | 0.87 | 0.88 |
| | Precision | 0.64 | 0.79 | 0.86 | 0.81 | 0.87 | 0.87 | 0.88 |
| | Recall | 0.63 | 0.78 | 0.85 | 0.80 | 0.87 | 0.86 | 0.88 |
| | F-measure | 0.63 | 0.79 | 0.86 | 0.81 | 0.86 | 0.86 | 0.88 |
| 1,000 | Accuracy | **0.65** | 0.82 | 0.88 | 0.85 | 0.89 | 0.88 | 0.89 |
| | Precision | 0.64 | 0.83 | 0.88 | 0.85 | 0.89 | 0.89 | 0.89 |
| | Recall | 0.63 | 0.82 | 0.87 | 0.84 | 0.88 | 0.88 | 0.89 |
| | F-measure | 0.63 | 0.82 | 0.88 | 0.84 | 0.89 | 0.88 | 0.89 |
| 10,000 | Accuracy | **0.65** | **0.88** | **0.90** | **0.88** | **0.90** | **0.91** | **0.91** |
| | Precision | **0.65** | **0.88** | **0.91** | **0.89** | **0.91** | **0.91** | **0.91** |
| | Recall | **0.64** | **0.87** | **0.90** | **0.88** | **0.90** | **0.90** | **0.91** |
| | F-measure | **0.64** | **0.87** | **0.90** | **0.88** | **0.90** | **0.91** | **0.91** |

for BERT feature extraction and CNN classifier is done using Python language deep learning framework of Tensorflow [69]. The split ratio of 70:30 is used for training and evaluation of models.

## V. RESULTS

In this section, first, we present results of the propaganda classification by using n-gram and NELA features. Next, we examine the performance of BERT and Word2Vec embedding features for Urdu and English text classification. We also take a deeper look at features that are actually learning the propaganda content. Finally, we conclude with the comparison of different textual features of ProSOUL with another propaganda detection system.

### A. N-GRAM AND NELA FEATURES

In section IV, we describe in detail n-gram and NELA features. The goal of our first experiment is to observe how the textual feature of n-gram and stylistic features of NELA performs for the classification of propaganda and non-propaganda class. In particular, we evaluate the performance of six combinations of NELA, word n-gram (W-ngram), and character n-gram (C-ngram). For the evaluation, we also try 3 to 10,000 C-ngram and W-ngram features. Table 8 illustrates the values of accuracy, precision, recall, and F-measure metrics for English and Urdu content.

We report results of six combinations of n-gram and NELA for different number of top features. These combinations of features are: i) C-ngram, ii) W-ngram, iii) NELA, iv) C-ngram + NELA, v) W-ngram + NELA, and vi) C-ngram + W-ngram + NELA (All). For brevity, in the case of English text classification, only the classification performance of "All" features is provided. We observe that in case of Urdu text classification, the performance of W-ngram feature is significantly better than different combinations of C-ngram and NELA features with accuracy, precision, recall, and F-measure in the range of 0.90-0.91. Also, C-ngram shows comparable performance with 0.88 accuracy. However, we note that when considering "All" features classification performance improves slightly. These performance results highlight that W-ngram feature represents better phrase contextual information of the propaganda class. On the other hand, C-ngram feature does not represent well the complex morphology of the Urdu language. Furthermore, we analyze the most informative features considered by the classifier to distinguish propaganda and non-propaganda articles. We extract the highest weighted features along with assigned weights from the best performing classifier trained with "All" features. Table 9 shows the most important features of Urdu text from propaganda and non-propaganda classes. We observe that the classifier assigns higher weights to W-ngram features as only one C-ngram appears among the

**TABLE 9.** Top 10 features of training dataset.

| C-ngram + W-ngram + NELA (All) | | | | |
|---|---|---|---|---|
| | Non-Propaganda | | Propaganda | |
| | Weights | Features | Weights | Features |
| 1 | -1.53 | رزاق | 1.52 | باراک |
| 2 | -1.37 | ذہن رکھیں | 1.50 | ازاد خیال |
| 3 | -1.31 | گارڈ | 1.45 | خاکہ پیش |
| 4 | -1.17 | ٹریبونل | 1.37 | امداد فراہم |
| 5 | -1.17 | نیلامی | 1.34 | کافی |
| 6 | -1.14 | باڈی | 1.29 | بھگت |
| 7 | -1.12 | اشاعتوں | 1.24 | انگیزی |
| 8 | -1.10 | بدقسمتی | 1.22 | انیشی |
| 9 | -1.09 | امن عمل | 1.21 | انے |
| 10 | -1.08 | غیر مجاز | 1.17 | بروز |

top ten features in row 9 of the propaganda class. We note that in the propaganda class *Barack* (the former president of USA) is assigned the highest weight, however, we point out that a prudent approach should be used while using W-ngram feature as it may learn information that is less relevant at that point of time.

Next, we examine classification results for the English text to understand the impact of automatic translation on the classification. We use the same documents from English and Urdu dataset for the comparison with a different combination of W-ngram, C-ngram, and NELA features. In contrast to the Urdu text, the classifier shows poor performance for the English text with 0.65 accuracy using top 10,000 n-grams. Intuitively, one may expect that classifiers for the Urdu text would perform lesser due to translation errors compared to the English text. However, in general, machine translation simplifies language variations by creating semantic links across languages while translating the text. This simplification of text transforms text data from a sparse semantic space into a dense semantic space resulting in a boosted performance of a text classifier. A similar observation was reported in [70]–[72] for the translated text. In addition, we compare the vocabulary distribution of English and Urdu dataset to analyze the simplification of variations in the translated text. The English dataset contains 352,510 unique tokens compared to 184,221 Urdu tokens extracted from the documents with the same content in the respective languages. This reduction of Urdu vocabulary by a factor of 1.9 after translation results in feature simplification of the translated text. However, this simplification of text can overfit classifiers on the training data and can have an adverse impact on the performance of classifiers when applied on the general world wide web content. In Section VI, we present a detailed analysis of the efficacy of classifiers on real-world content.

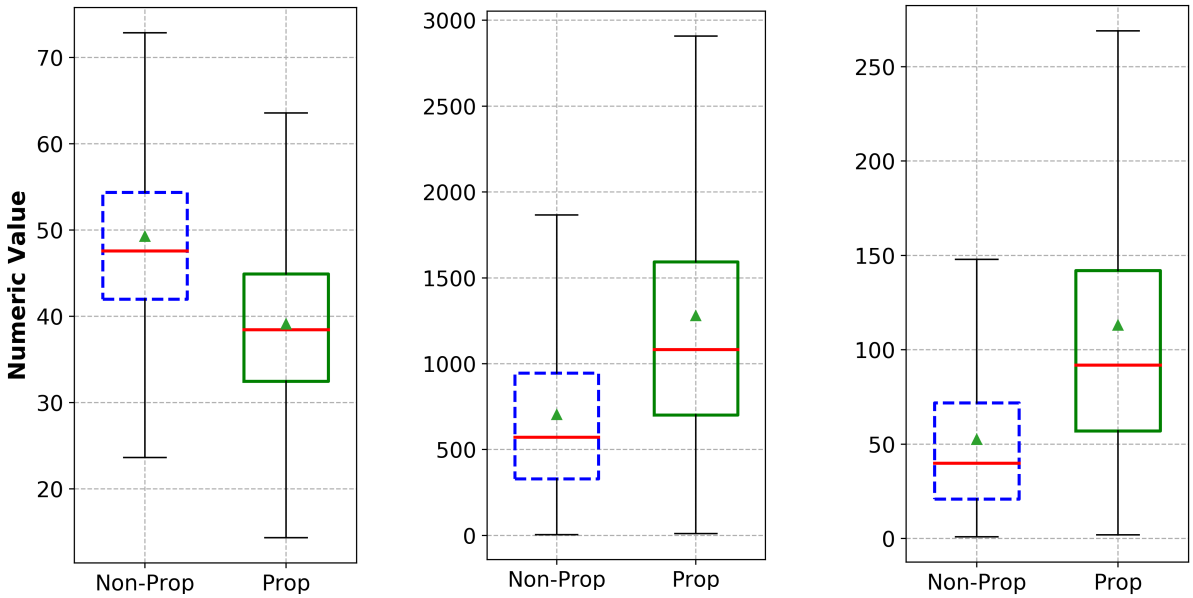
Furthermore, we focus on NELA features of the Urdu text by evaluating the performance of classifiers with all 32 possible combinations of six NELA features as described in Section IV. Table 10 shows the results of accuracy, precision, recall, and F-measure metrics for the best and worst-performing combinations of features. Our first observation is that the individual feature of TTR performs significantly better with 0.70 accuracy compared to the other features. This result indicates that lexical diversity is a contributing feature to distinguish the propaganda content in the Urdu language. The performance of TTR is further improved to 0.74 accuracy when used in combination with Cognitive and WordCount features. Interestingly, TopicSp feature shows the least performance with 0.51 accuracy. This poor performance of TopicSp feature highlights that the propaganda content in Urdu cannot be limited to any single topic. To examine this performance issue further, we analyze the distribution of TTR, WordCount, and Cognitive NELA features in our labelled dataset. Figure 2 shows a boxplot to compare 1st, median, and 3rd quartiles of propaganda and non-propaganda classes for three most informative NELA features. In Figure 2(a), the comparison of TTR feature for the two classes reveals the presence of lexical diversity in the non-propaganda labelled dataset as indicated by their median values of 38 and 47 for the propaganda and non-propaganda classes, respectively. In addition, we found that propaganda articles have 1.88 times more WordCount compared to non-propaganda articles as shown in Figure 2(b). Similarly, propaganda articles contain median 92 cognitive words per article compared to 40 cognitive words in non-propaganda articles, see Figure 2(c). The more presence of these cognitive words providing information like prediction, inferring, labelling, discrepancy, and certainty etc., indicates the presence of propaganda in the text.

### B. WORD EMBEDDING FEATURES

We now examine the classification performance of state-of-the-art Word2Vec and BERT embeddings for Urdu and English text. Recently, word embedding features like BERT and Word2Vec have demonstrated the unprecedented performance for Natural Language Processing (NLP) related tasks. These word embeddings leverage the context-aware learning to capture dependencies in textual sequences. The detailed description of implemented word embedding features for this article is discussed in Section IV. We provide values of accuracy, precision, recall, and F-measure in Table 11, for LogReg and CNN classifier trained with Word2Vec and BERT embeddings, respectively. We find that Word2Vec performs better for the classification of Urdu content with 0.87 accuracy. On the other hand, for English text classification, BERT embeddings outperformed Word2Vec with 0.95 accuracy. The sub-par performance of BERT in case of Urdu text classification is mainly due to the limited vocabulary of Urdu language available in the multilingual BERT [73]. Only 110k *wordpiece* vocabulary across 104 languages including Urdu is used for the training of

**TABLE 10.** Classifier performance for different combinations of NELA features.

| | Minimum performing features | | | | |
|---|---|---|---|---|---|
| | **Features** | **Accuracy** | **Precision** | **Recall** | **F-measure** |
| **1** | **TopicSp** | 0.51 | 0.44 | 0.47 | 0.40 |
| **2** | **TopicSp, AvgLen** | 0.51 | 0.46 | 0.48 | 0.42 |
| **3** | **Emotion, TopicSp, AvgLen** | 0.50 | 0.46 | 0.47 | 0.44 |
| **4** | **Emotion, TopicSp, AvgLen, WordCount** | 0.69 | 0.69 | 0.68 | 0.68 |
| **5** | **Emotion, TopicSp, AvgLen, WordCount, Cognitive** | 0.72 | 0.72 | 0.70 | 0.71 |
| | **Maximum performing features** | | | | |
| **1** | **TTR** | 0.70 | 0.70 | 0.70 | 0.70 |
| **2** | **TTR, Cognitive** | 0.72 | 0.72 | 0.71 | 0.71 |
| **3** | **TTR, Cognitive, WordCount** | 0.74 | 0.74 | 0.73 | 0.73 |
| **4** | **TTR, Cognitive, WordCount, AvgLen** | 0.75 | 0.75 | 0.74 | 0.74 |
| **5** | **TTR, Cognitive, WordCount, AvgLen, Emotion** | 0.75 | 0.75 | 0.74 | 0.74 |
| **6** | **All** | 0.75 | 0.75 | 0.74 | 0.74 |




(a) Token Type Ratio (TTR) (b) Word Count (c) Cognitive Words

**FIGURE 2.** Comparison of features for propaganda (Prop) and non-propaganda (Non-Prop) classes.

BERT model. On the contrary, Word2Vec model contains word embeddings of 132, 246, 587 unique Urdu tokens. Our findings emphasize on building monolingual BERT for the accurate Urdu text classification with a large scale dataset. In addition, the efficacy of multilingual BERT for the Urdu text can be investigated by performing topic-specific fine-tuning.

### C. PROPAGANDA LEARNING–CONTENT VS SOURCE

In order to examine the potential of ProSOUL to identify propaganda content from *unseen* sources, i.e., not learned during training, we evaluate the performance of classifiers by varying the number of sources in the training data as done in Barrón-Cedeno *et al.* [26]. In particular, our framework needs to learn the features of propaganda content independent of the source to avoid overfitting to the training data. With respect to the training data, we randomly select 5 and 47 sources of propaganda and non-propaganda

**TABLE 11.** Classification results of BERT and Word2Vec features.

| | English | | Urdu | |
|---|---|---|---|---|
| **Model** | **CNN** | **LogReg** | **CNN** | **LogReg** |
| **Features** | BERT Features | Word2Vec | BERT Features | Word2Vec |
| **Accuracy** | **0.95** | 0.87 | 0.82 | **0.87** |
| **Precision** | **0.95** | 0.86 | 0.77 | **0.84** |
| **Recall** | **0.94** | 0.86 | **0.88** | 0.83 |
| **F-measure** | **0.95** | 0.86 | 0.82 | **0.83** |

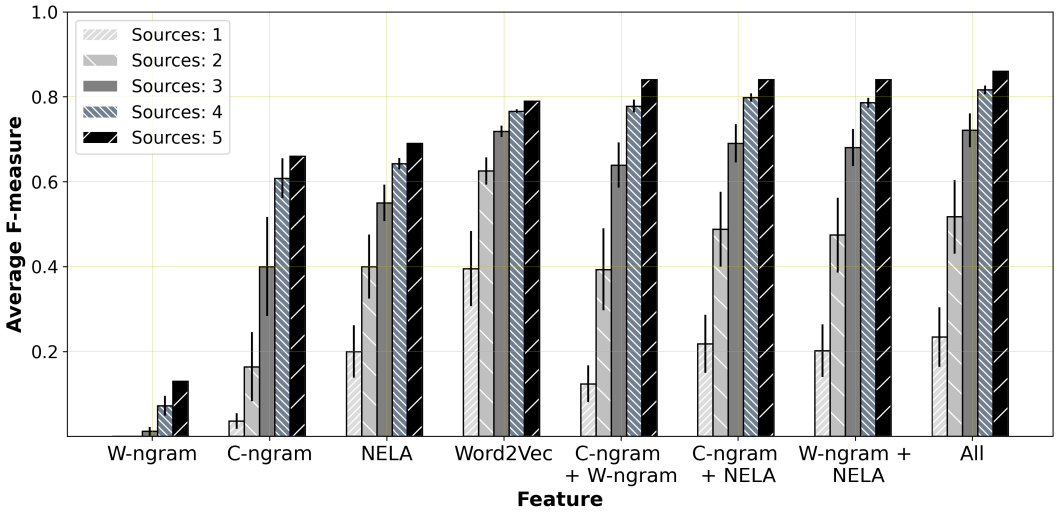



**FIGURE 3.** Average F-measure of propaganda class for propaganda learning – content vs source.

**TABLE 12.** Dataset for propaganda vs source learning.

| Class | Train | | Test | |
|---|---|---|---|---|
| | Sources | Articles | Sources | Articles |
| Propaganda | 5 | 2,552 | 5 | 2,770 |
| Non - Propaganda | 47 | 3,043 | 47 | 3,221 |
| **Total** | **52** | **5,595** | **52** | **5,991** |

articles, respectively. The content from the remaining sources is used for testing of classifiers. Table 12 shows the number of sources and articles for training and testing classifiers for both classes. For the training of classifiers, let $s_1$, $s_2$ ... $s_5$ be the propaganda content sources for the training data. In addition, we select $k \leq 5$ sources and use only documents belonging to selected sources for the training of classifiers. While we vary the number of propaganda sources in the training phase, the distribution of propaganda sources in testing did not change. Similarly, for the non-propaganda class sources in the training/testing phases did not change. Here, for the evaluation, the harmonically balanced metric of F-measure is used for all 16 possible combinations of sources. As such, we are only interested in the classification of propaganda articles, therefore, we calculate the F-measure value of the propaganda class only. Figure 3 shows the mean and standard deviation of the F-measure value for all possible combinations of C-ngram, W-ngram, and NELA features along with Word2Vec embeddings against the different number of sources. Note that no value of standard deviation is available with 5 number of sources as only one combination is possible. Interestingly, our results show the failure of C-ngram and W-ngram features for the identification of propaganda content from unseen sources. As we increase training sources from 1 to 5, F-measure values for C-ngram, W-ngram, and NELA features vary from 0 to 0.68. Moreover, compared to individual features, the combination of C-ngram, W-ngram, and NELA features show much better performance as indicated by the F-measure value of 0.84 with 5 sources. Thus, we find that individual features are not much effective in identifying propaganda content for the Urdu language. Finally, Word2Vec also shows competitive performance with F-measure values in the range of 0.4 to 0.83.

### D. ProSOUL AND PROPPY–COMPARISON

Next, we compare our ProSOUL framework with a similar propaganda detection system Proppy (see Section II for details). Table 13 shows a comparison of dataset, features, classifiers, and results of both systems. Note that the comparison of only Qprop dataset is presented due to its common use in both systems. First, we find that compared to 11,574 samples of ProSOUL, Proppy uses a large dataset of 51,294 samples of English news text with the highly imbalanced distribution of propaganda and non-propaganda class with 11.2% and 88.8% articles, respectively. Interestingly, on the contrary, ProSOUL utilizes dataset with 46% propaganda and 54% non-propaganda articles of Urdu language. Second, feature comparison of both system reveals that Proppy extracts n-gram, NELA, lexicon, and readability features, whereas ProSOUL leverages only n-gram, six NELA, and word embedding features due to linguistic resource scarcity for the Urdu language. Finally, with respect to classification results, Proppy achieves maximum F-measure value of 0.82 with an individual feature of C-ngram compared to ProSOUL with 0.91 F-measure with the combination of "All" features. When comparing the performance of stylistic and vocabulary features, both Proppy and ProSOUL show better performance for stylistic features in the identification of the propaganda content from *unseen* sources, also reported in Section V. In particular, this indicates that stylistic features exhibit better propaganda detection capabilities irrespective of the language of the text.

## VI. PROPAGANDA SOURCES IDENTIFICATION

In this section, we describe the process of crawling world wide web (WWW) to develop two large scale Urdu content

**TABLE 13.** ProSOUL and Proppy comparison.

| | Component | Proppy | ProSOUL |
|---|---|---|---|
| Dataset | Samples | 51,294 | 11,574 |
| | Sources | 104 | 104 |
| | Language | English | Urdu |
| Features | N-gram | Character, Word | Character, Word |
| | NELA | Extensive list of features | 6 features |
| | Lexicon | 18 lexicons | - |
| | Readability | 3 readability measures | - |
| | Word embeddings | - | BERT, Word2Vec |
| Classifiers | Model | Max Entropy | Logistic regression, CNN |
| Results (F-measure) | C - ngram | 0.82 | 0.87 |
| | W - ngram | 0.75 | 0.90 |
| | Nela | 0.51 | 0.74 |
| | Lexicon | 0.44 | - |
| | Readability | 0.21 | - |
| | All | 0.79 | 0.91 |
| | BERT | - | 0.82 |
| | Word2Vec | - | 0.83 |

**TABLE 14.** Humkinar web and news repository statistics.

| Dataset | Complete | | | Filtered | | |
|---|---|---|---|---|---|---|
| | Webpages | Websites | Tokens | Webpages | Websites | Token |
| Humkinar-Web | 6,399,476 | 7,922 | 18,986,146 | 6,307,775 | 1,818 | 17,363,653 |
| Humkinar-News | 622,838 | 35 | 1,844,838 | 613,434 | 26 | 1,844,831 |

repositories. We also present our methodology to identify and assign propaganda scores to webpages. Finally, we discuss the results of our propaganda scoring method on Urdu content repositories.

### A. URDU CONTENT REPOSITORIES

Just like English language content, online Urdu content is generated and published by a large number of websites on daily basis. The discovery of websites spreading propaganda content on the WWW is a computationally challenging task. In particular, it requires a large scale dataset representative of the Urdu content on the WWW. Due to the absence of such dataset, we crawl the WWW for three years (2016-2019) to develop Urdu content repository ''Humkinar-Web''. The implementation details of our crawling system are provided in [74]. In addition to crawling, high-quality of the corpus is ensured by selecting only those webpages containing at least 256 bytes of Urdu content. For this purpose, we enhance an open-source library Boilerpipe [75] to extract the main content of crawled webpages.

In general, Boilerpipe uses ''ArticleExtractor'' to extract the main content of a webpage. However, we observe that it also selects other noise content such as headings, sidebars, etc., along with the main content. To remove such noise, we modify the Boilerpipe algorithm by introducing a rule-based algorithm Web-AM [76]. The Web-AM exploits tree structure of Boilerpipe selected HTML. It removes the noise by using the observation that the webpage has the main content of large length with simple formatting and noise contains short text with rich formatting. Following this observation, Web-AM extracts the content from the node with the maximum number of characters and its neighbouring nodes present at the same tree level. Next, the extracted main content of a webpage is parsed using open-source language detection library Compact Language Detector 2 (CLD2) [77] to identify the text of different languages. We provide 'UTF-8' encoded text to CLD2 and get three levels of information, i.e., content-language, content bytes, and percentage. Additionally, for multi-lingual text, CLD2 provides top 3 languages in the webpage after ranking them according to each language content bytes. This information of content bytes and percentage for each language is used to filter out webpages with less than 256 bytes of Urdu content. After filtering, the Humkinar-Web repository contains 6.4 million Urdu webpages from 7,922 websites. We note that, during the crawling process, we did not confine the crawler to pre-determined Urdu websites. The crawled 7,922 websites meeting the threshold of 256 bytes of Urdu are explored during the crawling process. As such, Humkinar-Web contains content from a variety of domains including news, sports, health, religion, entertainment, and books etc. In addition, in order to test the efficacy of ProSOUL, we build another repository ''Humkinar-News'' containing 0.62 million news webpages from 35 manually selected propaganda-free websites. These propaganda-free websites are selected because the assignment of propaganda label by ProSOUL to these websites will prompt the failure of the framework on the general world wide web data. Similar to Humkinar-Web, we apply Web-AM algorithm and threshold of 256 bytes on Humkinar-News to ensure high-quality content. Moreover, we remove websites from Humkinar-Web

**TABLE 15.** Propaganda score bins.

| Bin number | Bin score | Probability range | Level |
|---|---|---|---|
| **1** | 0.0 | 0.0-0.2 | No |
| **2** | 0.2 | 0.2-0.4 | Low |
| **3** | 0.4 | 0.4-0.6 | Medium |
| **4** | 0.6 | 0.6-0.8 | High |
| **5** | 0.8 | 0.8-1.0 | Critical |

**TABLE 16.** Distribution of propaganda in Humkinar.

| | Number of articles | |
|---|---|---|
| **Class** | **Humkinar-Web** | **Humkinar-News** |
| **Propaganda** | 610,932 (9.7%) | 35,760 (5.8%) |
| **Non-Propaganda** | 5,696,843 (90.3%) | 577,674 (94.2%) |
| **Total** | **6,307,775** | **613,434** |

and Humkinar-News containing less than 100 documents. Table 14 shows statistics of complete and filtered dataset.

### B. PROPAGANDA SCORE–RESULTS

As a next step, we need to calculate propaganda scores of websites. For this, we define $Propaganda_{Score}$ of a website by taking mean propaganda score assigned by ProSOUL to all webpages of that website. First, all documents of a website present in a given repository are classified using ProSOUL. The classifier calculates the similarity of textual features of the webpage with labelled data and provides the probability score of being *propaganda* and *non-propaganda* class. By using this probability score, we assign *BinScore* to each document according to different levels of propaganda severity as described in Table 15. The propaganda score is divided into five levels: i) No, ii) Low, iii) Medium, iv) High, v) and Critical. After assigning a score to an individual document, the average of all documents of a website is calculated. This mean value is assigned as the final propaganda score of a website. Our equation of calculating the propaganda score is derived from the standard weighted averaging equation as described in Equation 5. The standard equation is mapped to Equation 6 for propaganda score calculation.

$$Weighted_{Average} = \frac{\sum_{i=1}^{n}(Weight_i * Term_i)}{n} \tag{5}$$

where

$n$ = Number of terms

$Weight_{\mathrm{i}}$ = Weight of ith term

$Term_{\mathrm{i}}$ = ith term

$$Propaganda_{Score} = \frac{\sum_{i=1}^{5}(NDocs_i * BinScore_i)}{Total_{Articles}} \tag{6}$$

where

$i$ = Bin Number

$NDocs_{\mathrm{i}}$ = Number of documents in i

$BinScore_{\mathrm{i}}$ = Bin Score of i

$Total_{\mathrm{Articles}}$ = Total Number of Articles

The goal of our work is to find the distribution of propaganda and non-propaganda content in Humkinar-Web and Humkinar-News repositories by using ProSOUL. The webpages in both repositories are classified using best performing classifier with the combination of "All" features. For assigning propaganda and non-propaganda labels to a webpage, classifier probability output of 0.5 is used as a binary threshold. Table 16 depicts the distribution of propaganda and non-propaganda webpages in Humkinar-Web and Humkinar-News. Our results show that Humkinar-News contains only 5.86% propaganda webpages compared to 9.86% propaganda webpages in Humkinar-Web. To explore this issue in-depth, the propaganda score for each website is calculated and the propaganda level is assigned according to bins in Table 15. Figure 4 shows the frequency of websites from Humkinar-Web and Humkinar-News associated to each propaganda level. Interestingly, all websites in Humkinar-News are assigned the label of "No" propaganda. This result highlights the accurate classification of news websites from the WWW by ProSOUL. However, for Humkinar-Web, 1194 (65.7%) websites are labelled as "No" propaganda and 624 (34.3%) websites are assigned a score in the range of 0.2-0.8. As such, Humkinar-Web contains websites from a variety of domains, these results are further investigated to analyze the domain dependency of ProSOUL. For this purpose, we analyze Humkinar-Web by manually classifying randomly selected 270 websites (News:87, Blog:58, Religious:94, Others:31) having different propaganda levels as shown in Table 17. Our manual classification reveals that Humkinar-Web contains the majority of websites (88.5%) from news, blogs, and religious domains. Therefore, websites from remaining domains are grouped into "Others" class. Table 17 also shows the propaganda levels of websites belonging to News, Blogs, Religious, and Others domains. These results show that 16 (19.5%) news websites and 27 (46.5%) blogs are assigned propaganda scores. During the manual investigation of these websites, we note the presence of political propaganda content which indicates the efficacy of ProSOUL on general news and blogs websites. On the other hand, the manual analysis of 56 (59.6%) religious websites – classified as propaganda websites by ProSOUL – reveals that there was no propaganda content present on those websites. The misclassification of these religious websites by ProSOUL is likely due to the difference in text complexity and vocabulary in news and religious content. Our analysis highlights that ProSOUL is capable of successfully identifying propaganda content from news websites and blogs with higher precision. In addition, we note that compare to other propaganda methods, ProSOUL identifies websites using *name calling* [10] propaganda method with more accuracy.

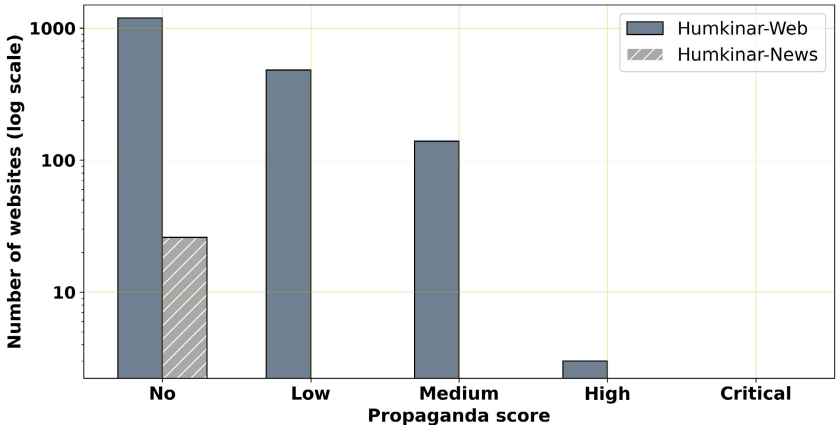



**FIGURE 4.** Distribution of score for Humkinar-Web and Humkinar-News websites.

**TABLE 17.** Domain vs propaganda level of Humkinar-Web.

| | | Domain | | | | |
|---|---|---|---|---|---|---|
| **Bin** | **Level** | **News** | **Blog** | **Religious** | **Others** | **Total** |
| **1** | No | 71 | 31 | 38 | 20 | **160** |
| **2** | Low | 11 | 18 | 32 | 8 | **69** |
| **3** | Medium | 5 | 9 | 21 | 3 | **38** |
| **4** | High | 0 | 0 | 3 | 0 | **3** |
| **5** | Critical | 0 | 0 | 0 | 0 | **0** |
| | **Total** | **87** | **58** | **94** | **31** | **270** |

## VII. CONCLUSION

In this article, we present ProSOUL (Propaganda Spotting in Online Urdu Language) framework to identify propaganda content in the Urdu language. First, a labelled dataset of Urdu propaganda content containing 5,322 propaganda and 6,252 non-propaganda news articles is developed by translating open-source English language dataset of *Qprop*. Our manual analysis of translated dataset reveals that 95.4% of sentences from articles are translated with the correct context. In addition, we build a linguistic dictionary of LIWC to extract psycho-linguistic features of Urdu text. Next, a detailed analysis of classifiers with n-gram, NELA, Word2Vec, and BERT features shows the best performance with 0.91 accuracy for the combination of word n-gram, character n-gram, and NELA features. For Urdu text classification, word embedding features of Word2Vec performs better than BERT features due to small amount of *useful vocabulary* of Urdu in BERT embeddings. We also evaluate the performance of different classifiers on test data acquired from *unseen* sources to study how different propaganda features are learned by classifiers. Our evaluation shows the failure of n-gram features in the classification of propaganda content in case of unseen sources. Further exploration of classification results for Urdu and English text highlights the better performance of the classifier for Urdu text due to simplification of semantic relation in the translated data. Moreover, we test ProSOUL on two large scale repositories of Humkinar-Web and Humkinar-News containing 6.4 and 0.62 million Urdu webpages, respectively. Overall, we find that 9.7% webpages of Humkinar-Web and 5.8% webpages of Humkinar-News have different levels of propaganda content. Further manual analysis reveals that ProSOUL shows superior performance in detecting propaganda content in case of web content from news and blog websites. Finally, we find that vocabulary difference present in different domains can adversely impact the classification of propaganda content.

In future, we plan to investigate different aspects of propaganda in detail to develop a generic propaganda detection system. Specifically, we want to extend the scope of our work to detect political propaganda, radicalization/hate speech, rumors, and disinformation. With respect to the platform, we intend to study and compare propaganda spread through different social media websites. We also plan to fine-tune the BERT model to enrich it with linguistic and topical features of Urdu content. We believe that such fine-tuning will help in better capturing of linguistic patterns to detect phrases and vocabulary used to spread a different kind of propaganda in Urdu text. Furthermore, we want to experiment with other machine learning architectures like OpenAI GPT2, Megatron-LM, and GPT-3 for better performance of Urdu text classification. Finally, we plan to build an online propaganda detection service using ProSOUL with the goal to enhance digital literacy.

## REFERENCES

[1] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. D., J. B., and K. Kochut, "Text summarization techniques: A brief survey," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 10, pp. 397–405, 2017.
[2] P. B. Brandtzaeg and A. Følstad, "Why people use chatbots," in *Proc. Int. Conf. Internet Sci.* Athens, Greece: Springer, 2017, pp. 377–392.
[3] M. Carlson, "The robotic reporter: Automated journalism and the redefinition of labor, compositional forms, and journalistic authority," *Digit. Journalism*, vol. 3, no. 3, pp. 416–431, May 2015.
[4] V. Lysenko and C. Brooks, "Russian information troops, disinformation, and democracy," *1st Monday*, vol. 23, no. 5, Apr. 2018. [Online]. Available: https://firstmonday.org/ojs/index.php/fm/article/view/8176
[5] D. Flick, "Combatting fake news: Alternatives to limiting social media misinformation and rehabilitating quality journalism," *SMU Sci. Technol. Law Rev.*, vol. 20, p. 375, Apr. 2017.

[6] G. L. Freed, S. J. Clark, A. T. Butchart, D. C. Singer, and M. M. Davis, "Parental vaccine safety concerns in 2009," *Pediatrics*, vol. 125, no. 4, pp. 654–659, Apr. 2010.

[7] D. J. Collison, "Corporate propaganda: Its implications for accounting and accountability," *Accounting, Auditing Accountability J.*, vol. 16, no. 5, pp. 853–886, Dec. 2003.

[8] S. D. Benegal and L. A. Scruggs, "Correcting misinformation about climate change: The impact of partisanship in an experimental setting," *Climatic Change*, vol. 148, nos. 1–2, pp. 61–80, May 2018.

[9] H. Cantril, "Propaganda analysis," *English J.*, vol. 27, no. 3, pp. 217–221, 1938.

[10] G. S. Jowett and V. O'donnell, *Propaganda and Persuasion*. Newbury Park, CA, USA: Sage, 2018.

[11] J. Ellul and K. Kellen, *Propaganda: The Formation Men's Attitudes*. New York, NY, USA: Vintage Books, 1973.

[12] W. J. Severin and J. W. Tankard, *Communication Theories: Origins, Methods, and Uses in the Mass Media*. New York, NY, USA: Longman, 1997.

[13] J. Mayer. (2018). *How Russia Helped Swing the Election for Trump*. Accessed: Jul. 8, 2020. [Online]. Available: https://www.newyorker.com/magazine/2018/10/01/how-russia-helped-to-swing-the-election-for-trump

[14] S. Illing. (2018). *Cambridge Analytica, the Shady Data Firm That Might be a Key Trump-Russia Link, Explained*. Accessed: Jul. 8, 2020. [Online]. Available: https://www.vox.com/policy-and-politics/2017/10/16/15657512/cambridge-analytica-facebook-alexander-nix-christopher-wylie

[15] C. Cadwalladr and E. Graham-Harrison. (2018). *Revealed: 50 Million Facebook Profiles Harvested for Cambridge Analytica in Major Data Breach*. Accessed: Jul. 8, 2020. [Online]. Available: https://www.theguardian.com/news/2018/mar/17/cambridge-analytica-facebook-influence-us-election

[16] R. DiResta, "The tactics and tropes of the Internet research agency," U.S. Senate Intell. Committee, Washington, DC, USA, Tech. Rep, Dec. 2019. [Online]. Available: https://digitalcommons.unl.edu/senatedocs/2/

[17] F. Carmichael and A. Hussain. (2019). *Pro-Indian 'Fake Websites Targeted Decision Makers in Europe'-BBC News*. Accessed: Jul. 28, 2020. [Online]. Available: https://www.bbc.com/news/world-asia-india-50749764

[18] W. Ahmed, J. Vidal-Alaball, J. Downing, and F. López Seguí, "COVID-19 and the 5G conspiracy theory: Social network analysis of Twitter data," *J. Med. Internet Res.*, vol. 22, no. 5, May 2020, Art. no. e19458.

[19] T. Warren. (2020). *British 5G Towers are Being Set on Fire Because of Coronavirus Conspiracy Theories*. Accessed: Jul. 30, 2020. [Online]. Available: https://www.theverge.com/2020/4/4/21207927/5g-towers-burning-uk-coronavirus-conspiracy-theory-link

[20] A. Aly, S. Macdonald, L. Jarvis, and T. M. Chen, "Introduction to the special issue: Terrorist online propaganda and radicalization," *Stud. Conflict Terrorism*, vol. 40, no. 1, pp. 1–9, 2017.

[21] J. Qin, Y. Zhou, and H. Chen, "A multi-region empirical study on the Internet presence of global extremist organizations," *Inf. Syst. Frontiers*, vol. 13, no. 1, pp. 75–88, Mar. 2011.

[22] R. A. Jawad. (2015). *Raw Boosts Funding in Balochistan*. Accessed: Jul. 8, 2020. [Online]. Available: https://nation.com.pk/05-Sep-2015/raw-boosts-funding-in-balochistan

[23] Z. Ebrahim. (2007). *World Health Day-Pakistan: Anti-Polio Drive Hits Resistance*. Accessed: Jul. 9, 2020. [Online]. Available: http://www.ipsnews.net/2007/04/world-health-day-pakistan-anti-polio-drive-hits-resistance/

[24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Assoc. Comput. Linguistics: Hum. Lang. Technol.*, vol. 1, 2019, pp. 1–14.

[25] P. Gupta, K. Saxena, U. Yaseen, T. Runkler, and H. Schütze, "Neural architectures for fine-grained propaganda detection in news," in *Proc. 2nd Workshop Natural Lang. Process. Internet Freedom: Censorship, Disinformation, Propaganda*, 2019, pp. 92–97.

[26] A. Barrón-Cedeño, I. Jaradat, G. Da San Martino, and P. Nakov, "Proppy: Organizing the news based on their propagandistic content," *Inf. Process. Manage.*, vol. 56, no. 5, pp. 1849–1864, Sep. 2019.

[27] *Media Bias/Fact Check–Search and Learn the Bias of News Media*. Accessed: Jul. 7, 2020. [Online]. Available: https://mediabiasfactcheck.com/

[28] (2019). *Ethnologue: Languages of the World*. Accessed: Jul. 8, 2020. [Online]. Available: https://www.ethnologue.com/language/urd

[29] J. W. Pennebaker, M. E. Francis, and R. J. Booth, "Linguistic inquiry and word count: LIWC 2001," *Mahway: Lawrence Erlbaum Associates*, vol. 71, p. 2001, Apr. 2001.

[30] B. D. Horne, W. Dron, S. Khedr, and S. Adali, "Assessing the news landscape: A multi-module toolkit for evaluating the credibility of news," in *Proc. Companion The Web Conf. Web Conf. (WWW)*, 2018, pp. 235–238.

[31] *Word Vectors for 157 Languages*. Accessed: Jul. 10, 2020. [Online]. Available: https://fasttext.cc/docs/en/crawl-vectors.html

[32] *Humkinar-Web*. Accessed: Jul. 10, 2020. [Online]. Available: https://www.humkinar.com.pk/

[33] *Humkinar-News*. Accessed: Jul. 10, 2020. [Online]. Available: https://humkinar.pk/News

[34] V. Pérez-Rosas, B. Kleinberg, A. Lefevre, and R. Mihalcea, "Automatic detection of fake news," in *Proc. 27th Int. Conf. Comput. Linguistics*, 2018, pp. 3391–3401.

[35] N. Ruchansky, S. Seo, and Y. Liu, "CSI: A hybrid deep model for fake news detection," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 797–806.

[36] Y. Long, "Fake news detection through multi-perspective speaker profiles," in *Proc. 8th Int. Joint Conf. Natural Lang. Process.*, 2017, pp. 252–256.

[37] M. Granik and V. Mesyura, "Fake news detection using naive bayes classifier," in *Proc. IEEE 1st Ukraine Conf. Electr. Comput. Eng. (UKRCON)*, May 2017, pp. 900–903.

[38] H. Rashkin, E. Choi, J. Y. Jang, S. Volkova, and Y. Choi, "Truth of varying shades: Analyzing language in fake news and political fact-checking," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 2931–2937.

[39] W. Hou and Y. Chen, "CAUnLP at NLP4IF 2019 shared task: Context-dependent BERT for sentence-level propaganda detection," in *Proc. 2nd Workshop Natural Lang. Process. Internet Freedom: Censorship, Disinformation, Propaganda*, 2019, pp. 83–86.

[40] Y. Dinkov, I. Koychev, and P. Nakov, "Detecting toxicity in news articles: Application to bulgarian," in *Proc. Natural Lang. Process. Deep Learn. World*, Oct. 2019, pp. 247–258.

[41] R. Baly, M. Mohtarami, J. Glass, L. Màrquez, A. Moschitti, and P. Nakov, "Integrating stance detection and fact checking in a unified corpus," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, vol. 2, 2018, pp. 1–7.

[42] A. Ferreira Cruz, G. Rocha, and H. Lopes Cardoso, "On sentence representations for propaganda detection: From handcrafted features to word embeddings," in *Proc. 2nd Workshop Natural Lang. Process. Internet Freedom: Censorship, Disinformation, Propaganda*, 2019, pp. 107–112.

[43] M. Amjad, G. Sidorov, and A. Zhila, "Data augmentation using machine translation for fake news detection in the Urdu language," in *Proc. 12th Lang. Resour. Eval. Conf.*, 2020, pp. 2537–2542.

[44] L. Van Wissen and P. Boot, "An electronic translation of the LIWC dictionary into Dutch," in *Proc. Electron. Lexicography 21st Century: ELEX 2017 Conf.*, 2017, pp. 703–715.

[45] A. L. Andrei, "Development and evaluation of tagalog Linguistic Inquiry and Word Count (LIWC) dictionaries for negative and positive emotion," MITRE Corp., Mclean, VA, USA, 2014.

[46] Al-Samawi and A. Muhammed, "Language errors in machine translation of encyclopedic texts from English into Arabic: The case of Google translate," *Arab World English J.*, vol. 3, pp. 182–211, May 2014.

[47] K. Leetaru and P. A. Schrodt, "GDELT: Global data on events, location, and tone," in *ISA Annual Convention*. San Francisco, CA, USA: The International Studies Association (ISA), 2013.

[48] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, and M. Norouzi, "Google's neural machine translation system: Bridging the gap between human and machine translation," 2016, *arXiv:1609.08144*. [Online]. Available: http://arxiv.org/abs/1609.08144

[49] J. W. Pennebaker and L. A. King, "Linguistic styles: Language use as an individual difference.," *J. Personality Social Psychol.*, vol. 77, no. 6, pp. 1296–1312, 1999.

[50] N. Holtzman, "Inferring grandiose narcissism from text: LIWC versus machine learning," *J. Lang. Social Psychol.*, Jul. 2020, doi: 10.1177/0261927X20936309.

[51] A. P. Valenti, M. Chita-Tegmark, L. Tickle-Degnen, A. W. Bock, and M. J. Scheutz, "Using topic modeling to infer the emotional state of people living with Parkinson's disease," *Assistive Technol.*, pp. 1–10, Jun. 2019, doi: 10.1080/10400435.2019.1623342.

[52] M. L. Newman, J. W. Pennebaker, D. S. Berry, and J. M. Richards, "Lying words: Predicting deception from linguistic styles," *Personality Social Psychol. Bull.*, vol. 29, pp. 665–675, Jun. 2003.
[53] A. Piolat, R. J. Booth, C. K. Chung, M. Davids, and J. W. Pennebaker, "La version française du dictionnaire pour le LIWC: Modalités de construction et exemples d'utilisation," *Psychologie Française*, vol. 56, no. 3, pp. 145–159, 2011.
[54] N. Ramírez-Esparza, J. W. Pennebaker, F. A. García, and R. Surià, "La psicología del uso de las palabras: Un programa de computadora que analiza textos en español," *Revista Mexicana de Psicología*, vol. 24, no. 1, pp. 85–99, 2007.
[55] C. Huang et al, "The development of the Chinese linguistic inquiry and word count dictionary," *Chin. J. Psychol.*, vol. 54, no. 2, pp. 185–201, 2012.
[56] F. Carvalho, G. Santos, and G. P. Guedes, "AffectPT-br: An affective lexicon based on LIWC 2015," in *Proc. 37th Int. Conf. Chilean Comput. Sci. Soc. (SCCC)*, Nov. 2018, pp. 1–5.
[57] (2019). *English-Words: A Text File Containing 479k English Words for All Your Dictionary*. Accessed: Jul. 6, 2020. [Online]. Available: https://github.com/dwyl/english-words
[58] (2020). *Complete Collection of Stop Words for the Urdu Language*. Accessed: Jul. 10, 2020. [Online]. Available: https://github.com/urduhack/urdu-stopwords
[59] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. New York, NY, USA: McGraw-Hill, 1986.
[60] P. R. Fitzsimmons, B. Michael, J. L. Hulley, and G. O. Scott, "A readability assessment of online Parkinson's disease information," *J. Roy. College Physicians Edinburgh*, vol. 40, no. 4, pp. 292–296, 2010.
[61] C. Andolina, "Syntactic maturity and vocabulary richness of learning disabled children at four age levels," *J. Learn. Disabilities*, vol. 13, no. 7, pp. 27–32, Aug. 1980, doi: 10.1177/002221948001300705.
[62] T. Mikolov, É. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," in *Proc. Int. Conf. Lang. Resour. Eval. (LREC)*, 2018, pp. 1–4.
[63] P. McCullagh and J. Nelder, *Generalized Linear Models* (Chapman & Hall/CRC Monographs on Statistics and Applied Probability Series). 2nd ed. London, U.K.: Chapman & Hall, 1989. [Online]. Available: http://books.google.com/books?id=h9kFH2_FfBkC
[64] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[65] G. Van Rossum and F. L. Drake, Jr., *Python reference manual*. Amsterdam, The Netherlands: Centrum voor Wiskunde en Informatica, 1995.
[66] E. Loper and S. Bird, "NLTK: The natural language toolkit," in *Proc. ACL Workshop Effective Tools Methodol. Teach. Natural Lang. Process. Comput. Linguistics*, Philadelphia, PA, USA, 2002, pp. 63–70.
[67] F. Pedregosa, G. Varoquaux, and A. Gramfort, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.
[68] R. Rehurek and P. Sojka, "Software framework for topic modelling with large corpora," in *Proc. LREC Workshop New Challenges NLP Frameworks*, 2010, pp. 45–50.
[69] M. Abadi, A. Agarwal, P. Barham, and E. Brevdo, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*. [Online]. Available: http://arxiv.org/abs/1603.04467
[70] N. Otani, H. Kiyomaru, D. Kawahara, and S. Kurohashi, "Cross-lingual knowledge projection using machine translation and target-side knowledge base completion," in *Proc. 27th Int. Conf. Comput. Linguistics*, 2018, pp. 1508–1520.
[71] M. Farrús Cabeceran, M. Ruiz Costa-Jussà, J. B. Mariño Acebal, and J. A. Rodríguez Fonollosa, "Linguistic-based evaluation criteria to identify statistical machine translation errors," in *Proc. 14th Annu. Conf. Eur. Assoc. Mach. Transl.*, 2010, pp. 167–173.
[72] J. Jiang, S. Pang, X. Zhao, L. Wang, A. Wen, H. Liu, and Q. Feng, "Cross-lingual data transformation and combination for text classification," 2019, *arXiv:1906.09543*. [Online]. Available: http://arxiv.org/abs/1906.09543
[73] X. Dong and G. de Melo, "A robust self-learning framework for cross-lingual text classification," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 6307–6311.
[74] H. M. Shafiq and M. A. Mehmood, "NCL-Crawl: A large scale language-specific Web crawling system," in *Proc. CLT*, 2020, pp. 1–7.
[75] C. Kohlschütter, "Boilerpipe–boilerplate removal and full text extraction from html pages," Google Code, 2010.
[76] N. Aslam, B. Tahir, H. M. Shafiq, and M. A. Mehmood, "Web-AM: An efficient boilerplate removal algorithm for Web articles," in *Proc. Int. Conf. Frontiers Inf. Technol. (FIT)*, Dec. 2019, pp. 287–2875.
[77] J. Ooms and D. Sites, "CLD2: Google's compact language detector 2," Tech. Rep., 2018, vol. 7, p. 2019. Accessed: Jul. 6, 2020. [Online]. Available: https://github.com/cld2owners/cld2

**SOUFIA KAUSAR** received the B.Sc. degree in computer science from the Lahore College for Women University, Lahore, Pakistan, in 2017, and the M.S. degree in computer science from the National University of Computing and Emerging Sciences (FAST-NU), Lahore, in 2019. She is currently working as a Research Officer with the Al-Khawarizmi Institute of Computer Science (KICS), University of Engineering and Technology (UET), Lahore. Her research interests include machine learning, deep learning, computer vision, and text mining.

**BILAL TAHIR** received the B.Sc. degree in electrical engineering from the National University of Computer and Emerging Sciences (FAST-NU), Lahore, Pakistan, in 2014, and the M.S. degree in computer engineering from the University of Engineering and Technology (UET), Lahore, in 2018. Since 2017, he has been working as a Research Officer with the Al-Khawarizmi Institute of Computer Science (KICS), UET at Lahore. His research interests include machine learning for images and text, natural language processing, deep learning, and information retrieval.

**MUHAMMAD AMIR MEHMOOD** received the Ph.D. degree in engineering from the Department of Electrical Engineering and Computer Science, Technische Universität Berlin/Deutsche Telekom Innovation Laboratories, Berlin, Germany, in 2012, under the supervision of Prof. Anja Feldmann. He is currently working as an Assistant Professor with the Al-Khawarizmi Institute of Computer Science, University of Engineering and Technology, Lahore, Pakistan. He has been the Head of the High-Performance Computing and Networking Laboratory (HPCNL), since 2013. His research interests include Internet measurements, big data, information retrieval, and deep learning.