Hindawi
Scientific Programming
Volume 2021, Article ID 1296076, 11 pages
https://doi.org/10.1155/2021/1296076

Research Article

Sentence Classification Using N-Grams in Urdu Language Text

Malik Daler Ali Awan[1], **Sikandar Ali**[2], **Ali Samad**[1], **Nadeem Iqbal**[3], **Malik Muhammad Saad Missen**[1] and **Niamat Ullah**[4]

[1]*Department of Information Technology, Faculty of Computing, The Islamia University of Bahawalpur, 63100 Bahawalpur, Pakistan*
[2]*Department of Information Technology, The University of Haripur, 22621 Haripur, Khyber Pakhtunkhwa, Pakistan*
[3]*Muhammad Nawaz Shareef University of Agriculture, Multan 61000, Pakistan*
[4]*Department of Computer Science, University of Buner, 19290 Sawarai Buner, Khyber Pakhtunkhwa, Pakistan*

Correspondence should be addressed to Sikandar Ali; sikandar@cup.edu.cn

Received 17 April 2021; Revised 27 May 2021; Accepted 7 November 2021; Published 22 November 2021

Academic Editor: Wei-Chuen Yau

The usage of local languages is being common in social media and news channels. The people share the worthy insights about various topics related to their lives in different languages. A bulk of text in various local languages exists on the Internet that contains invaluable information. The analysis of such type of stuff (local language's text) will certainly help improve a number of Natural Language Processing (NLP) tasks. The information extracted from local languages can be used to develop various applications to add new milestone in the field of NLP. In this paper, we presented an applied research task, "multiclass sentence classification for Urdu language text at sentence level existing on the social networks, i.e., Twitter, Facebook, and news channels by using N-grams features." Our dataset consists of more than 1,00000 instances of twelve (12) different types of topics. A famous machine learning classifier Random Forest is used to classify the sentences. It showed 80.15%, 76.88%, and 64.41% accuracy for unigram, bigram, and trigram features, respectively.

1. Introduction

The text is still dominant and prominent way of communication instead of only pictures, emoji, sounds, and animations. The innovative environment of communication, the real-time availability of the Internet, and the unrestricted communication mode of social networks attracted billions of people around the world. People share insights about various topics, opinions, views, ideas, and events happening around them on social networks in different languages. Social media and news channels: such communication platforms created space for local languages to share information. Google input tool (https://www.google.com/inputtools/) provides the language transliteration support to 88 different languages. The development of many local languages supporting tools is another factor that boosted the usage of local languages on social media and news channels. Obviously, people prefer to communicate in local languages instead of global languages because of easiness in conveying messages. It is also causing to generate heterogeneous data on Internet.

Sifting worthy insights from an immense amount of heterogeneous text of multiple local languages existing on social media is one of the interesting and challenging tasks of Natural Language Processing (NLP). Local language processing certainly provides the invaluable insights to develop NLP applications. These applications can respond in emergencies, outbreaks, and natural disasters, i.e., rain, flood, and earthquake [1]. The interesting feature like real-time interaction of social media has facilitated millions of people to share their intent, appreciation, or criticism [2], i.e., enjoying discount offer by selling brands or criticizing the quality of the product. Extracting and classifying such information are valuable to improve the quality of the product. The implementation of smart cities possesses a lot of challenges, such as decision making, event management, communication, and information retrieval. Extracting useful

insights from an immense amount of text dramatically enhances the worth and quality of smart cities [3]. Similarly, the classified information can be used to predict the effects of the event on the community and take security and rescue measures. Sentence classification information can be used to collect relevant information about the specific topic, top-trends, stories, text summarization, and question and answering system [4, 5]. Such information can be also used to predict upcoming events, situations, and happening. For example, sudden occurrence of earthquake can cause causalities, but classifying such news surely helps us response quickly and save the lives in disasters.

There are many local languages that are being used for communication, i.e., Arabic, Hindi, Persian, Turkish, Urdu, etc. on social media, i.e., Twitter and Facebook. In [6], more than 300 million Urdu language users were reported all around the world. In Pakistan and India, more than 65 million people can speak, understand, and write the Urdu language [7, 8]. The Urdu language is also a national language for different sates in India. It is the national language [9] of Pakistan, which is the 6th populous country in the world (https://www.worldometers.info/world-population/population-by-country/). Urdu is widely adopted as a second language all over Pakistan [9, 10]. In other South-Asian countries [11], i.e., Bangladesh, Iran, and Afghanistan also have a considerable number of Urdu language users.

Sentence classification in Urdu language text is a very interesting and challenging task. The lack of resources to classify sentences into different categories is the major challenge. The prominent characteristics of the Urdu language that made the event classification tasks complex and challenging are listed here:

(i) Cursive language
(ii) Morphologically rich
(iii) Different grammatical structure
(iv) Right to the left scriptwriting style
(v) No capitalization
(vi) Lack of resources

The lack of resources, i.e., part of speech tagger (PoS), words stemmer, datasets, and the word annotators are other factors that made Urdu text processing very complex. A considerable amount of Urdu text exists on social networks [12]. There exist only a few referential works on Urdu text. The factors, i.e., huge amount of data, resource poor, and very short referential work, motivated us to explore the Urdu language text. In this research article, we decided to classify sentences into different categories. The purpose of research work is to design a system to extract useful information from Urdu language text and develop various NLP applications.

1.1. Our Contribution

(i) In this research article, we tried to classify Urdu language text at sentence level in 12 different categories
(ii) N-grams features, i.e., unigram, bigram, and trigram, are selected to classify sentences
(iii) We developed multiclass annotated/labeled dataset
(iv) A dataset larger than others in size (instances) as reported in a state-of-the-art is used to classify sentences

1.2. Our Limitation

(i) The work is domain-specific (only for Urdu language), but other resource poor languages can be explored in the future
(ii) It can only classify reported types of events at sentence level

2. Challenges in Sentence Classification for Urdu Language Text

The resource poor languages possess a lot of challenges in the context of resource lacking, i.e., part of speech tagger, annotated datasets, sentence parsers, stemmer, and lexicons. The information extraction related to different events, business, and disasters varies from domain to domain. In the literature, for example, the event was defined in various aspects, such as a verb-, adjective-, and noun-based environmental situation [13, 14]. Extraction and classification of such information require grammatical-, semantic-, and contextual-based information. There are a number of tools for English-like languages that support tackling such challenging task, but the Urdu language is lacking such resources.

Multiclass classification is a type of classification that is the task of automatically assigning the most relevant one class from the given multiple classes (see Figure 1). It also has some serious challenges like detection of sentences that are overlapping in multiple classes [15, 16].

2.1. Limitation of Existing Text Processing Tools for Urdu Language Text. Google language translator (https://translate.google.com/?hl=en) supports more than 100 languages. The Urdu language comprises unique structure, complex writing script, and rich morphological features that isolate it from other languages. Urdu is a cursive language written in right to left order and considered as one of the resource-poor languages [8]. It is a mix-composition of different other languages, i.e., Arabic, Persian, Turkish, and Hindi [10]. In contrast to cursive languages, there exists some noteworthy work of information extraction and classification for, i.e., English, French, German, and many other noncursive languages [9, 11].

In the past, researchers were impassive in cursive languages because of poor resources. Therefore, a very low amount of research work exists in cursive language, i.e., Arabic, Persian Hindi, and Urdu [17]. Lack of resources in cursive language was the main barrier to make the research unexcited and vapid [8]. But now, the last few decades' cursive languages have attracted researchers. The main

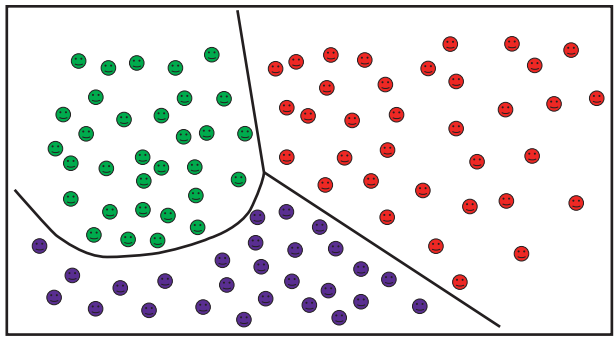

Figure 1: Multiple category classification

reason behind the attraction is that a large amount of cursive language data is being generated rapidly on regular basis. Now, some processing tools also have been developed, i.e., part of speech tagger, word stemmer, and annotator that play an important role by making research handier. But these tools are still limited, commercial, and close domain (http://www.cle.org.pk/).

Due to the reasons for poor resources, to achieve our goal, we decided to translate Urdu language text into English language text by using Google translator. The output of Google translator is given in Table 1 which depicted that translation is confined to literal translation. It completely misinterpreted the contextual and semantic insights. In example 1, the word (کانٹے کا مقابلہ- kaantay ka muqabla, contest) is translated as (quota), which gives the meaning of (share, حصہ), while the correct word is "contest." All the bold and italicized words from examples 1 to 6 showed the wrong translation and the limited capability of Google translator. Although, in the past decade, a number of researchers used the strategy of language translation using Google language translation tools and translated the resource poor languages to English language to cope with resources lacking problems, the case of Urdu language is different from other languages. It can be observed in Table 1 that Urdu language cannot be processed by the existing tools.

There is a need to explore and develop resources for the Urdu language to improve the NLP. Table 1 shows that all problems in the Urdu language cannot be resolved by English language processing tools.

2.2. Lack of Recourse. Natural language processing is tightly coupled with resources, i.e., processing resources, datasets, and semantic, syntactical, and contextual information. Textual features; i.e., Part of Speech (PoS) and semantics are important for text processing. Central Language of Engineering (CLE) provides limited access to PoS tagger because of the close domain and paid that diverged the researcher to explore Urdu text.

Contextual features, i.e., grammatical insight (tense) and sequence of words, play important role in text processing. Because of the morphological richness nature of Urdu, a word can be used for a different purpose and convey different meanings depending on the context of contents. Unfortunately, the Urdu language is still lacking such tools that are openly available for research. Other processing resources, i.e., stemmer, lemmatize, and annotators, are also close domain. Dataset is the core element of research. There is no specific dataset for multiclass sentence classification for Urdu language text. Some datasets for the Urdu language generally exist for name entity extraction with a small number of instances, which are given in Table 2 below.

3. Literature Review

Classification of events from the textual dataset is a very challenging and interesting task of Natural Language Processing (NLP). Textual contents on social media are explored in different ways to extract event information. Generally, the event has been defined as a verb, noun, and adjective [14]. A survey that discussed the various tasks and techniques related to the Urdu language text is available for the researchers as a benchmark text [27]. The event detection is a generic term that is further divided into event extraction and event classification. The lack of resources made research impassive in past to explore the cursive languages like Hindi, Arabic, Persian, and Urdu [28]. A system is designed [29] for Arabic text classification using multiple reduct algorithm. The proposed system showed 94% and 86% accuracies for K-NN and J48 classifiers.

Urdu textual contents explored [30] for classification using the majority voting algorithm. They categorized Urdu text into seven classes, i.e., Health, Business, Entertainment, Science, Culture, Sports, and Wired. They used 21769 news documents for classification and reported 94% precision and recall. The dataset was evaluated using these algorithms, Linear SGD, Bernoulli Naïve Bayes, Linear SVM, Naïve Bayes, random forest classifier, and Multinomial Naïve Bayes. They did not report the overall accuracy of the system for multiple classes. The information about feature selection is also omitted by the researchers, but comparatively, we disclosed the feature selection, engineering, and accuracy of classifiers for multiclasses. Our dataset consists of 102960 instances of sentences and twelve (12) classes that are comparatively very greater.

A framework [31] proposed a tweet classification system to rescue people looking for help in a disaster like a flood. The developed system was based on the Markov Model achieving 81% and 87% accuracy for classification and location detection, respectively. The features used in their system are as follows:

(i) Number of words in a tweet (w)
(ii) Verb in a tweet by (verb)
(iii) Number of verbs in a tweet by (v)
(iv) Position of the query by (Pos)
(v) Word before query word (before)
(vi) Word after query word (after)

A neural network-based system that is a combination of conventional neural network and recurrent neural network was designed to extract events from English, Tamil, and

Table 1: Wrong translation example of Google Translator.

Sr. No.	Urdu	English
1	بھارت اور انگلینڈ کے درمیان کانٹے کا مقابلہ جاری ہے	The quota is going on between England and India.
2	بجٹ بل کو لے کے حکومت اور اپوزیشن میں ٹھن گئی	Government and opposition were *tied* to take the budget bill.
3	عیدالاضحی پر مسلمان جانوروں کی قربانی کرتے ہیں	Muslims *sacrifice sacrifices* on the occasion.
4	یران پر امریکی پابندیوں کے بعد بھارت نے بھی ایرانا سے تیل کی درآمد بند کر دی	*After Iran's sanctions on Iran*, India also stopped importing oil from Iran.
5	شہید لیفٹیننٹ کرنل راشد کریم بیگ کی نماز جنازہ گلگت میں ادا کر دی گئی	The funeral prayers of martyr Lieutenants Colonel Rashid Karim Bag were *released* in Gilgit.
6	دھرتی ماں پر جان قربان کرنے والے جانباز قوم کے ہیرو ہیں	*On the forehead mother is the hero of the people who sacrificed life*.

Table 2: Urdu language dataset.

Sr. No.	Dataset
1	Enabling Minority Language Engineering (EMILLE) (only 200000 tokens) [18]
2	Becker-Riaz corpus (only 50000 tokens) [19]
3	Computing Research Laboratory (CRL) annotated corpus (only 55,000 tokens are publicly available data corpora) [20]
4	International Joint Conference on Natural Language Processing (IJCNLP) workshop corpus (only 58252 tokens)
5	Urdu Named Entity Recognition (UNER) [4]
6	Corpus of 705 sentences [21]
7	Corpus of BBC Urdu, Daily Jang [22]
8	corpus of 19.3 million words [23]
9	COUNTER, Naïve, NPUU [24, 25]
10	DSL Urdu news [26]

Hindi languages. It showed f-score 39.91%, 37.42%, and 39.71% [32].

To classify Urdu news headlines [23] by using maximum indexes of vectors, the stemmed and nonstemmed textual data was used for experiments. The system was specifically designed for text classification instead of sentence classification. Their proposed system achieved 78.0% for competitors and 86.6% accuracy for the proposed methodology. In comparison, we used sentences of Urdu language for classification and explored the textual features of sentences. A multiclass event classification task [18] was performed for Urdu language text that evaluated the performance of different classifiers. On the contrary, we evaluated the performance of Random Forest classifiers for different level of n-gram features.

Twitter [19] was used to detect natural disasters, i.e., bush fires, earthquakes, and cyclones, as well as humanitarian crises. To be aware of emergencies situation in natural disasters a framework work designed based on SVM and Naïve Bayes classifiers using word unigram, bigram, length, number of #Hash tag, and reply. These features were selected on sentence bases. SVM and Nave Bayes showed 87.5% and 86.2% accuracy, respectively, for tweet classification, i.e., seeking help, offering for help, and none. An intent mining system was developed [2] to facilitate citizens and cooperative authorities using a bag of the token model. The researchers exploited the hybrid feature representation for binary classification and multilabel classification. It showed a 6% to 7% improvement in the top-down feature set processing approach. Intelligence information retrieval plays a vital role in the management of smart cities [33]. This information helps enhance security and emergency management capabilities in smart cities. A very popular social website's twitter textual data used [20] to extract and classify events for the Arabic language. Implementation and testing of Support Vector Machine (SVM) and Polynomial Network (PN) algorithms showed promising results for tweet classification 89.2% and 92.7%. Stemmer with PN and SVM magnified the classification by 93.9% and 91.7%, respectively. Social events [30] were extracted assuming that, for prediction, either parties or one of them is aware of the event. The research aimed to find the relation between related events. Support Vector Machine (SVM) with kernel method was used on adopted annotated data of Automated Content Extraction (ACE). Structural information derived from the dependency tree and parsing tree is utilized to derive new structures that played important role in event identification and classification. In [24], Urdu text classification deep learning models evaluated using existing benchmark datasets [25]. The classification is performed for small, medium, and large size of preexisting dataset for product analysis [24]. A benchmark for the Urdu text classification [26] presented the comparison of machine learning classifiers using n-gram features on two closed source benchmark datasets CLE Urdu Digest 1000k, and CLE Urdu Digest 1Million and publicly available dataset.

A research study [34] was conducted to evaluate the students teaching environment using deep learning classifier RNN. The dataset consists of 15 4000 instructor reviews. Deep learning and conventional machine learning classifiers are evaluated on the dataset. Deep learning classifiers using word embedding ensemble with attention mechanism and

the system showed 98.29% accuracy. In the research article [21], linguistic and psychological features sets are used to analyze the sentiment on twitter. Five linguistic categories and their ensembles were used as input and four supervised classifiers evaluated. The analysis showed that ensemble models are better in performance than conventional classifiers [21]. The authors reported [34] that software products and organization performance were analyzed using k-mean and parallel k-mean clustering to improve the educational environment. The evaluation experiment was performed on 10,000 to 5,000 numerical instances. The results analysis showed that parallel clustering improved the time elapsed. A sentiment analysis performed to analyze the usage of product. The reviews given on product were used for sentiment classification. The weighted word embedding and deep neural networks are used in combination. The combined architecture of TF-IDF and CNN-LSTM showed better results as compared to conventional deep learning models [40]. The AdaBoost and Naïve Bayes showed the highest accuracy 88.1% with combination of consistency features. To classify unstructured data, hybrid supervised clustering based on ensemble scheme is used to compare the conventional and ensemble classifiers [34]. A feature selection model was introduced to classify text that is based on generic ranking aggregation [34].

4. Sentence Classification Methodology

Textual data classification possesses a lot of challenges, i.e., word similarity, poor grammatical structure, miss-use of terms, and multilingual words. We decided to adopt a supervised classification approach to classify Urdu sentences into different categories.

Sentence classification for Urdu Language text is performed by supervised machine learning approach. A complete overview of the multiclass sentence classification methodology is given here in Figure 2. The proposed framework is depicted in Figure 3.

4.1. Data Collection. Urdu textual data is collected from popular social networks, i.e., twitter, famous news channel's blogs, i.e., Geo News, Urdu Point, and BBC Urdu. Data collection consists of a title, body, published date, location, and URL. In the phase of data collection, a PHP-based web scraper is used to crawl data from the above-cited social websites. A complete post is retrieved from the websites and stored in MariaDB (database). As we have described earlier, our task is to classify events at sentence level instead of whole document classification. Our dataset consists of more than one million (102, 960) label sentences of different types of events. All the different types of events used in our research work and their maximum number of instances are shown in Figure 4.

There are twelve different types of events that we try to classify in our research work. These events are a factual representation of the state of the people. In Figure 2, the imbalances number of instances of each event is given. It can be visualized that politics, sports, and fraud and corruption

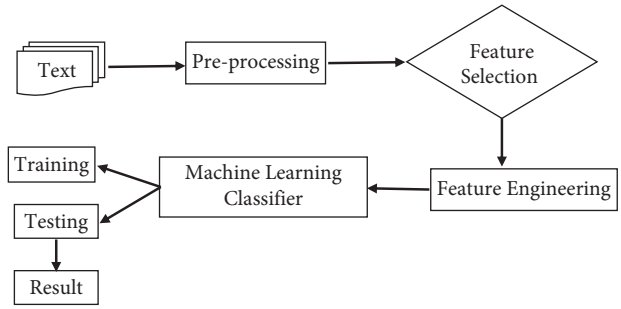

Figure 2: Generic abstract diagram of proposed system.

have a higher number of instances, while inflation, sexual assault, and terrorist attack have a lower number of instances. These imbalances number of instances made our classification more interesting and challenging.

In case of multiclass sentence classification, the corpus comprises many classes. There are different types of events used in our research work, i.e., sports, Inflation, Murder & Death, Terrorist Attack, Politics, Law and Order, Earthquake, Showbiz, Fraud and Corruption, Weather, Sexual Assault, and Business. All the sentences of the dataset are labeled by the above mentioned twelve (12) different types of events. Finally, a numeric (integer) value is assigned to each type of event label (see Table 3 for more details on the label and its relevant numeric value).

4.2. Preprocessing. The initial steps are performed on the corpus to prepare machine learning algorithms, because textual data cannot be directly processed by machine learning classifiers. It also contains many irrelevant words. So, we must apply some preprocessing steps; stemming is a powerful technique in preprocessing to find the root words and reduce the feature space. But, in our case, the nature of dataset is entirely different, because our dataset is a mix-up of novel/rare event and common events. Furthermore, generally, each type of event has varying vocabulary of text. Therefore, we assumed/considered that stemming would not affect the performance; that is why there is no need to use stemmer. The details of all the preprocessing steps followed in our research problem to prepare the dataset are given in Figure 5.

4.2.1. Post Splitting. The PHP crawler extracted the body of the post. It comprises many sentences as a paragraph. In the Urdu language script, sentences end with a sign called "-" Hyphen (Khatma-ختمہ). It is a standard punctuation mark in the Urdu language to represent the end of the sentence. As mentioned earlier, we are performing event classification at the sentence level. So, we split paragraphs of every post into sentences. Every line in the paragraphs ending at Hyphen is split as a single line.

4.2.2. Stop Words Elimination. Generally, those words that occur frequently in text corpus are considered as stop words. These words merely affect the performance of the classifier.

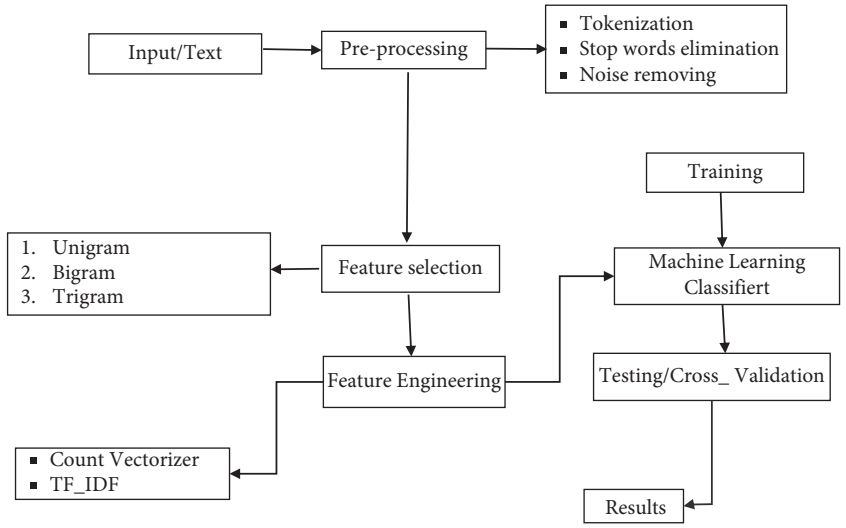

Figure 3: Flow diagram of proposed methodology for sentence classification from Urdu language text.

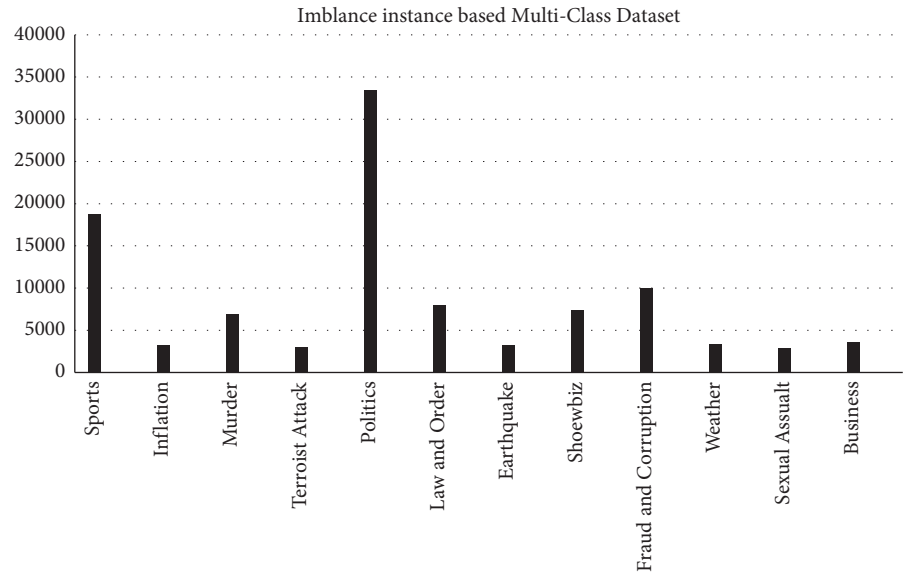

Figure 4: Maximum number of instances.

Table 3: Sentence with annotated label.

Sentence	Label
Sports	1
Inflation	2
Murder and Death	3
Terrorist Attack	4
Politics	5
Law and Order	6
Earthquake	7
Showbiz	8
Fraud and Corruption	9
Rain/Weather	10
Sexual Assault	11
Business	12

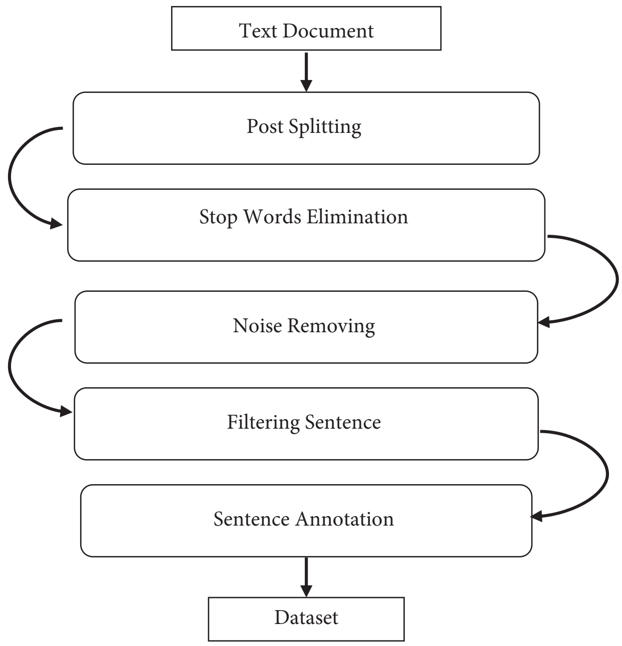

Figure 5: Dataset preprocessing steps.

The punctuation marks ("!", "@"," #", etc.) and frequent words of the Urdu languages (کا, کے, کی وغیرہ, etc.) are the common examples of stop words. All the stop words [33] that do not play an influential role in event classification for the Urdu language text are eliminated from the corpus. Stop words elimination reduces the memory and processing utilization and makes the processing efficient.

4.2.3. Noise Removal. Our data is collected by different sources (see Section 4.1). It contains a lot of noisy elements, i.e., multilanguage words, links, mathematical characters, special symbols, etc. In collected corpus, we found many multilingual sentences in the post. To make our corpus clean and ready for further processing, we removed those sentences, irrelevant links, and special characters from the corpus.

4.2.4. Filtering Sentences. The nature of our problem confined us to define the limit of words per sentence. Because of the multiple types of events, it is probably hard to find the sentence of the same length. We decided to keep the maximum number of sentences in our corpus. All those sentences, which are very short and very long, are removed from our corpus. We observed that a lot of sentences vary in length from 5 words to 250 words. We decided to use sentences that consist of 5 words to 150 words to lemmatize our research problem and consumption of processing resources.

4.2.5. Sentence Labeling. In supervised learning, providing output (Label) details in the corpus is a core element. Sentence labeling is an exhausting task that requires deep knowledge and an expert's skill of language. All the sentences were manually labeled by observing the title of the post and body of sentences by Urdu language experts (see Table 2). Three Urdu language experts were engaged in the task of sentence labeling. One of them is Ph.D. (Scholar), while the other two are M.Phil. To the best of our knowledge, it is the first largest labeled dataset for the multiclass event in the Urdu language. It consists of more than 1,00,000 label instances.

4.3. Feature Selection. The performance of prediction or classification models is cohesively related to the appropriate feature selection. Features selection in machine learning is very important to develop accurate models. Features are fundamental parameters that are given as input to learning classifiers in the phase of model development. The trained model further can be tested to classify, predict, or assign labels to new instances. A sentence contains very limited information that is insufficient to differentiate among multiple sentences. Classification of text at sentence level requires the contextual information. To capture the contextual information of sentences, we decided to use unigram, bigram, and trigram features for text classification at sentence level. The examples of the proposed features are given in Table 4. For example, the sentence "The brutal attack of Covid-19 killed billion of people" can be converted to *n*-gram features after preprocessing, as shown below:

4.3.1. Feature Engineering. Feature engineering is a way of generating specific features from a given set of features and converting selected features to machine-understandable format.

Our dataset that is text-based consists of more than 1,00,000 labeled instances, i.e., sports, inflation, death, terrorist attack, sexual assault, etc. For the 12 classes, we generated three features, i.e., unigram, bigram, and trigram. All the textual features are converted to numeric format using (Term Frequency_ Inverse Document Frequency) TF_IDF. The scikit-learn package is used to transform text data into numerical value [17].

4.3.2. Term Frequency Inverse Document Frequency. It is a statistical measure of word w to understand the importance of that word for specific document d in the corpus. The importance of w is proportionally related to frequency; i.e., the higher the frequency, the more important. The mathematical formulas of the TF_IDF are given below:

Term Frequency (TF) counts the number of terms how often it appears in the document. The formula of term frequency is given as follows:

$$\mathrm{TF} = \frac{\text{Number of time term } t \text{ appears in a document}}{\text{Total number of term } t \text{ in the document}}. \quad (1)$$

The inverse document frequency is used to identify that a term is rare or common in the corpus. The formula is given here:

Table 4: Selected features.

Sr. No.	Feature_Name	Example
1	Unigram	"Brutal," "attack", "Covid-19," "killed," "billion," "people"
2	Bigram	"Brutal attack," "attack Covid-19," "Covid-19 killed," "killed billion," "billion people"
3	Trigram	"Brutal attack Covid-19, "attack Covid-19 killed," "Covid-19 killed billion," "killed billion people"

$$\mathrm{IDF}_t = \mathrm{Log}_e \frac{\text{Total number of the documents}}{\text{Total number of the documents term } t \text{ appears}}. \quad (2)$$

The TF-IDF consists of the product of two components, i.e., term frequency and inverse document frequency.

$$\mathrm{TF\_IDF} = \mathrm{TF} * \mathrm{IDF}. \quad (3)$$

4.4. Training Dataset. To develop a generic model for event classification, we divided our dataset into three subsets, i.e., training dataset, testing, and validation dataset.

Random distribution of data is performed by using python library scikit. We distributed a 70% dataset randomly for training purposes. There are 72072 labeled instances for different types of events in our training dataset. A multiclass-instances-based training dataset is used for training deep learning models to develop a generic model.

4.5. Testing Dataset. To evaluate the performance of our proposed framework, we used a 30% dataset for testing/validations purposes. It consists of 30888 unknown instances that are never seen by trained models.

5. Machine Learning Classifier

Classifiers are the algorithms used to classify data instances. In machine learning for textual data, many classifiers exist, but, in our research work, we decided to use the Random Forest for classification, because it consists of multiple decision trees that are based on rules. Furthermore, it has never been used for text classification at the sentence level for the Urdu language text.

5.1. Random Forest. A multiclass classifier that is based on a large amount of imbalance dataset. It is a meta learner having multiple trees that form the forest.

Overall classification in random forest is determined by the vote of random trees. Vote of tress is used to assign the specific class to the input (see Figure 6). It follows a bootstrapping-like technique in the training phase. One-third of instances are preserved in out-of-bag. Features are chosen randomly for each tree. Finally, out-of-bag instances are used to test the model. Average misclassification of overall trees is known as estimated errors that can be used to measure the performance of classifier [6].

5.2. Performance Measuring Parameters. The most common performance measuring [29] parameters, i.e., precision, recall, and f1_measure, are used to evaluate the proposed framework.

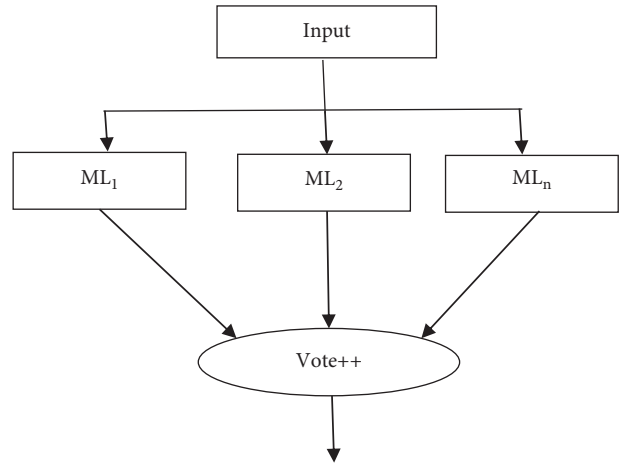

Figure 6: Random forest

5.2.1. Precision. Precision: it is the measurement of the exactness of the classifier. The precision calculating formula is given as follows:

$$\text{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}. \quad (4)$$

5.2.2. Recall. Recall: it measures the completeness of the classifier results. It is calculated by the following equation:

$$\text{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}. \quad (5)$$

5.2.3. F1_Measure. F1_Measure is the harmonic mean of precision and recall and can be calculated as

$$\mathrm{F1} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (6)$$

5.2.4. Accuracy. Accuracy: it is the most common measure for classifier performance and can be calculated as follows:

$$\text{Accuracy} = \frac{(\mathrm{TP} + \mathrm{TN})}{(\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN})}. \quad (7)$$

6. Experimental Results

To evaluate our dataset, the Python package scikit-learn is used to perform text classification at the sentence level. Comparison of the results obtained by using the proposed n-grams features is given below.

Table 5: Unigram

Label	Event	Precision	Recall	F1_Measure
1	Sports	0.94	0.93	0.93
2	Inflation	0.93	0.97	0.95
3	Murder and Death	0.71	0.62	0.67
4	Terrorist Attack	0.78	0.55	0.65
5	Politics	0.71	0.90	0.79
6	law and order	0.66	0.40	0.50
7	Earthquake	1.00	1.00	1.00
8	Showbiz	0.93	0.81	0.86
9	Fraud and corruption	0.75	0.58	0.65
10	Rain/weather	0.94	0.98	0.96
11	Sexual Assault/Intercourse	0.96	0.98	0.97
12	Business	0.84	0.63	0.72
Overall accuracy (%)		*80.15*		

Table 6: Bigram.

Label	Event	Precision	Recall	F1_Measure
1	Sports	0.92	0.89	0.91
2	Inflation	0.92	0.96	0.94
3	Murder and Death	0.70	0.55	0.62
4	Terrorist Attack	0.51	0.54	0.52
5	Politics	0.70	0.86	0.77
6	law and order	0.58	0.39	0.47
7	Earthquake	0.99	1.00	1.00
8	Showbiz	0.88	0.72	0.79
9	Fraud and corruption	0.69	0.55	0.61
10	Rain/weather	0.93	0.97	0.95
11	Sexual Assault/Intercourse	0.93	0.98	0.96
12	Business	0.78	0.64	0.71
Overall accuracy (%)		*76.88*		

Table 7: Trigram.

Label	Event	Precision	Recall	F1_Measure
1	Sports	0.44	0.96	0.60
2	Inflation	0.92	0.96	0.94
3	Murder and Death	0.68	0.39	0.49
4	Terrorist Attack	0.67	0.38	0.49
5	Politics	0.76	0.61	0.68
6	law and order	0.56	0.30	0.39
7	Earthquake	1.00	1.00	1.00
8	Showbiz	0.91	0.45	0.60
9	Fraud and corruption	0.70	0.46	0.56
10	Rain/weather	0.94	0.96	0.95
11	Sexual Assault/Intercourse	0.95	0.98	0.96
12	Business	076	0.47	0.58
Overall accuracy (%)		*64.41*		

In Table 5, we present the performance measuring parameters of different types of sentences. The Random Forest (RF) classifier showed 80.15% accuracy using unigram feature.

We also evaluated the performance of Random Forest classifier for bigram features to enhance the accuracy of the system. But bigram showed lower results as compared to unigram. The overall accuracy using bigram is 76.88% presented in Table 6.

We further explored the trigram features, but the accuracy of classifiers was decreasing. The trigrams features showed very low results as compared to unigram and bigram features (see Table 7 for more details). The machine learning classifier Random Forest showed 64.41% overall accuracy.

The comparison of accuracy of all features, i.e., unigram, bigram, and trigram, is given in Figure 7.

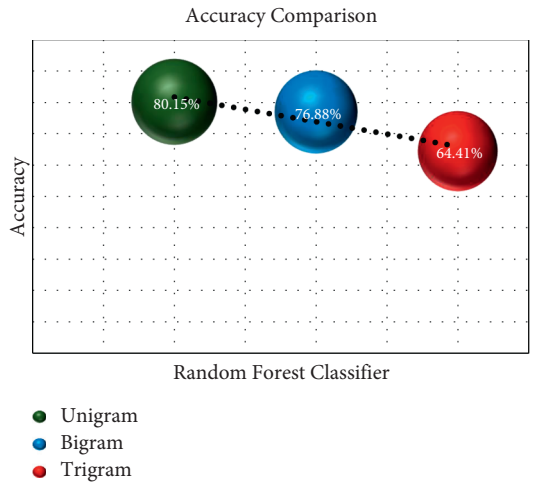

Figure 7: Radom forest accuracies using unigram, bigram, and trigram features

7. Conclusion and Future Work

In a comprehensive review of Urdu literature, we found a few numbers of referential works related to Urdu text processing. The main hurdle in Urdu exploration is the unavailability of the processing resources i.e., eventual dataset, close-domain Part of speech tagger, lexicons, and other supporting tools.

A massive amount of Urdu textual data exists on social networks and news websites. Multiclass classification for the Urdu language text at the sentence level is performed by selecting appropriate features. N-grams are the key features to achieve our expected results, because they can retain the sequence and contextual information of sentence. Count_ Vectorizer and TF-IDF feature generating methods are used to convert text into numeric real value for machine learning models. We did not use the word2vec model because of lacking pretrained models. Furthermore, customized pretrained models that are prepared using the corpus in hand are very inefficient in context of accuracy. The reason is that the amount of data is insufficient to build such model (Custom Word2Vec model).

Urdu event dataset was used to evaluate Random Forest using unigram, bigram, and trigram features. In our proposed framework, Random Forest showed Unigram, bigram, and trigram accuracy of 80.15% 76.88%, and 64.41%, respectively.

(i) Many open-source tools, i.e., PoS tagger, annotation tools, event datasets, and lexicons, can be created to extend the research areas in the Urdu language.

(ii) In the future, many other types of events of other domains like the medical event, social, local, and religious events can be classified using an advanced form of machine learning, i.e., deep learning.

(iii) In the future, grammatical, contextual, and lexical information can be used to categorize events. Temporal information related to sentence can be further utilized to classify it as real and retrospective.

(iv) Classification of Urdu language text can be performed at the document level and phrase level.

(v) Deep learning classifiers can be used for a many other types of sentences.

Data Availability

The data collected during the data collection phase are available from the corresponding authors upon request.

Conflicts of Interest

The authors declare that they have no potential conflicts of interest.

Acknowledgments

The authors thank Dr. Mujtaba Hasnain, Department of Information Technology, Faculty of Computing, The Islamia University of Bahawalpur, 63100 Bahawalpur, Pakistan.

References

[1] J. Yin, S. Karimi, A. Lampert, M. Cameron, B. Robinson, and R. Power, "Using social media to enhance emergency situation awareness," in *Proceedings of the Twenty-fourth international joint conference on artificial intelligence*, Buenos Aires, Argentina, 2015 June.

[2] H. Purohit, G. Dong, V. Shalin, K. Thirunarayan, and A. Sheth, "Intent classification of short-text on social media," in *Proceedings of the 2015 IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity)*, pp. 222–228, IEEE, Chengdu, China, 2015, December.

[3] M. Alkhatibl, M. El Barachi, and K. Shaalan, "Using Arabic social media feeds for incident and emergency management in smart cities," in *Proceedings of the 2018 3rd International Conference on Smart and Sustainable Technologies (SpliTech)*, pp. 1–6, IEEE, Split, Croatia, 2018 June.

[4] W. Khana, A. Daudb, J. A. Nasira, and T. Amjada, "Named entity dataset for Urdu named entity recognition task," *Organization*, vol. 48, p. 282, 2016.

[5] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy (effects of finite dimensions and interfaces on the basic properties of ferromagnets)," in *Spin Arrangements and Crystal Structure, Domains, and Micromagnetics*, G. T. Rado and H. Suhl, Eds., vol. 3, pp. 271–350, Academic, New York, NY, USA, 1963.

[6] F. Livingston, "Implementation of Breiman's random forest machine learning algorithm," *ECE591Q Machine Learning Journal Paper*, pp. 1–13, 2005.

[7] M. Naz and S. Hussain, "Binarization and its evaluation for Urdu Nastalique document images," in *Proceedings of the Multi Topic Conference (INMIC), 2013 16th International*, pp. 213–218, IEEE, Lahore, Pakistan, 2013 December.

[8] S. Mukund, R. Srihari, and E. Peterson, "An information-extraction system for Urdu---a resource-poor language," *ACM Transactions on Asian Language Information Processing*, vol. 9, no. 4, p. 15, 2010.

[9] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," *Lingvisticæ Investigationes. International Journal of Linguistics and Language Resources*, vol. 30, no. 1, pp. 3–26, 2007.

[10] S. M. Ghulam and T. Rahim Soomro, "Twitter and Urdu," in *Proceedings of the 2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, pp. 1–6, IEEE, Sukkur, Pakistan, 2018.

[11] K. Riaz, "Concept search in Urdu," in *Proceedings of the 2nd PhD workshop on Information and Knowledge Management*, pp. 33–40, Napa Valley, CA, USA, 2008, October.

[12] S. T. Ghulam and T. Rahim Soomro, "Twitter and Urdu," Department of Computer Science SZABIST Dubai Campus Dubai United Arab Emirates (UAE).

[13] S. Ramesh and S. Kumar, "Event extraction from natural language text," *International Journal of Engineering Ssciences & Research Technology*, 2016.

[14] J. P. Singh, Y. K. Dwivedi, N. P. Rana, A. Kumar, and K. K. Kapoor, "Event classification and location prediction from tweets during disasters," *Annals of Operations Research*, vol. 283, pp. 1–21, 2017.

[15] X. Kong, X. Shi, and P. S. Yu, "Multi-label collective classification," in *Proceedings of the 2011 SIAM International Conference on Data Mining*, pp. 618–629, Society for Industrial and Applied Mathematics, Mesa, AZ, USA, 2011, April.

[16] A. Sarker and G. Gonzalez, "Portable automatic text classification for adverse drug reaction detection via multi-corpus training," *Journal of Biomedical Informatics*, vol. 53, pp. 196–207, 2015.

[17] F. Pedregosa, G. Varoquaux, A. Gramfort et al., "Scikit-learn: machine learning in Python," *Journal of Machine Learning Research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[18] D. Ali, M. M. S. Missen, and M. Husnain, "Multiclass event classification from text," *Scientific Programming*, vol. 2021, Article ID 6660651, 15 pages, 2021.

[19] A. Agarwal and O. Rambow, "Automatic detection and classification of social events," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pp. 1024–1034, Association for Computational Linguistics, Cambridge, MA, USA, 2010 October.

[20] S. Hussain, *Resources for Urdu Language Processin*, Center for Research in Urdu Language Processing National University of Computer and Emerging Sciences B Block, Faisal Town, Pakistan.

[21] A. Onan, "Sentiment analysis on Twitter based on ensemble of psychological and linguistic feature sets," *Balkan Journal of Electrical and Computer Engineering*, vol. 6, no. 2, pp. 69–77, 2018.

[22] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes Twitter users: real-time event detection by social sensors," in *Proceedings of the 19th international conference on World wide web*, pp. 851–860, ACM, Raleigh North, CA, USA, 2010 April.

[23] A. R. Ali and M. Ijaz, "Urdu text classification," in *Proceedings of the 7th international conference on frontiers of information technology*, p. 21, 2009 December.

[24] M. P. Akhter, Z. Jiangbin, I. R. Naqvi, M. Abdelmajeed, and M. Fayyaz, "Exploring deep learning approaches for Urdu text classification in product manufacturing," *Enterprise Information Systems*, pp. 1–26, 2020.

[25] M. P. Akhter, Z. Jiangbin, I. R. Naqvi, M. Abdelmajeed, A. Mehmood, and M. T. Sadiq, "Document-level text classification using single-layer multisize filters convolutional neural network," *IEEE Access*, vol. 8, pp. 42689–42707, 2020.

[26] M. N. Asim, M. U. Ghani, M. A. Ibrahim, W. Mahmood, A. Dengel, and S. Ahmed, "Benchmarking performance of machine and deep learning-based methodologies for Urdu text document classification," *Neural Computing & Applications*, pp. 1–33, 2020.

[27] A. Daud, W. Khan, and D. Che, "Urdu language processing: a survey," *Artificial Intelligence Review*, vol. 47, no. 3, pp. 279–311, 2017.

[28] N. Alsaedi and P. Burnap, "Arabic Event Detection in Social Media," *Cardiff School of Computer Science and Informatics*, Cardiff University, Cardiff, CF, UK.

[29] Q. A. Al-Radaideh and M. A. Al-Abrat, "An Arabic text categorization approach using term weighting and multiple reducts," *Soft Comput*, vol. 23, no. 14, pp. 5849–5863, 2018.

[30] M. Usman, Z. Shafique, S. Ayub, and K. Malik, "Urdu text classification using majority voting," *International Journal of Advanced Computer Science and Applications*, vol. 7, no. 8, pp. 265–273, 2016.

[31] A. Kuila, S. chandra Bussa, and S. Sarkar, "A neural network based Event extraction system for Indian languages," *Fire*, vol. 2266, 2018.

[32] K. Ahmed, M. Ali, S. Khalid, and M. Kamran, "Framework for Urdu News Headline Classification," *Journal of Applied Computer Science & Mathematics*, vol. 10, 2016.

[33] M. Alkhatib, M. El Barachi, and K. Shaalan, *Using Arabic Social Media Feeds for Incident and Emergency Management in Smart Cities*, University of Wollongong in Dubai, Faculty of Engineering and Information Sciences, Knowledge Village, Dubai, UAE.

[34] R. Shang, B. Ara, I. Zada, S. Nazir, Z. Ullah, and S. U. Khan, "Analysis of simple K-mean and parallel K-mean clustering for software products and organizational performance using education sector dataset," *Scientific Programming*, vol. 2021, Article ID 9988318, 20 pages, 2021.