Urdu News Content Classification Using Machine Learning Algorithms

Umair Arshad[1], Khawar Iqbal Malik[2], Hira Arooj[3]

[1, 2]Department of Computer Science, University of Lahore Sargodha Campus, Sargodha, Pakistan.

[3]Department of Mathematics and statistics, University of Lahore Sargodha Campus, Sargodha, Pakistan

Email: khawar.iqbal@cs.uol.edu.pk

ABSTRACT

As the world has become a global village, the flow of news in terms of volume and speed increases. It is necessary to engage computing machines for assisting people in dealing with this massive data. The availability of different types of news and such material on the Internet serves as a source of information for billions of users. Millions of people in our subcontinent speak and understand Urdu. There are several classification techniques that are available and are applied to classify English news like political, Education, Medical, etc. Plenty of research work has been done in multiple languages but Urdu is still to be worked on due to a lack of resources. This research evaluates the performance of twelve (12) different Machine learning classifiers for the Urdu News text Classification problem. The analysis was performed on a relatively big and recent collection of Urdu text that contains over 0.15 million (153,050) labeled instances of eight different classes. In addition, after applying pre-processing techniques, the TF-IDF weighting technique was adopted for feature selection and data extraction. After evaluating various machine learning methods, the SVM outperforms the other eleven algorithms with an accuracy of 91.37 %. We also compare its results with other classifiers like linear SVM, Logistic regression, SGD, Naïve bays, ridge regression, and a few others.

KEYWORDS: Urdu News classification, NLP, TF-IDF, Machine learning.

1. INTRODUCTION

Nowadays, information on the Internet is available in a range of simple-to-understand spoken languages, and people may access gathered information due to the vast number of electronic texts available on the Internet. Classification techniques may be considered among the most important research areas. In the Indo-Pak region, Urdu is Pakistan's official language. It is also spoken in several other countries including Bangladesh, India in Asia, the United Kingdom in Europe, and Canada in North America. Apart from these, it is also spoken in regions of other countries like UAE, America, etc. [1]. Urdu speakers are located all across the world, not just in Asia. Its users can be found on the Internet all across the world. In Pakistan, there are around 220 million Urdu speakers, with an estimated 500 million or more globally. Urdu's lexicon is largely derived from Arabic and Persian. It lacks linguistic resources although having a massive number of Urdu speakers. Even though Urdu is supplied by a plethora of dictionaries, there is still a scarcity of WordNet-like semantic concepts [2].

Text mining or text classification is a challenging task because it needs a lot of pre-processing techniques to transform unstructured input into structured data. Here we took the data of Urdu news and classify it so that we can put it over automated blogs or we can put a label on a document printed in Urdu. For example, if the document is posted over a blog and we can easily predict that the document belongs to a sports category or business etc. Similarly, printed documents like Urdu news articles could be labeled properly. Our problem statement comprises labeling the Urdu text news automatically so it can be posted on the property news section for online readers at different news blogs.

In English Natural Language Processing (NLP), there is a lot of pre-processing work done, but very few pre-processing tools are available in Urdu. However, for Urdu text pre-processing, some articles were published, we will discuss them later in the literary review section. We use two data sets in this study: the first is a self-created dataset that was generated from a range of online news sources, while the second dataset was acquired from publicly available sources. Health, Business & Economics, Entertainment, Science & Technology, Sports, Politics, and World are the eight pre-defined classifications used here.

In recent years, the classification of Naive Bayes and Logistic regression, SVM, and other machine learning classifiers have been examined over the Urdu news dataset, but here we used a new dataset and apply a few more machine learning classifiers as well. We used multiple machine learning algorithms and evaluated the most optimal one for our dataset. We used twelve distinct machine learning approaches to propose a Classification Model for Urdu News that achieves maximum accuracy and produces the best results on our dataset.

The suggested method's most essential feature is the classification of texts. Manually classifying documents necessitates reading all of the articles before they can be labelled and stored. Because no work has been done on recent and huge datasets in Urdu to classify Urdu news blogs automatically, the categorization job will require a significant number of specialists with extensive expertise and specialized knowledge. We will train several machine learning models on large and recent Urdu datasets, and then our algorithms will automatically categories material in Urdu news blogs.

This study is divided into many sections, the second of which is about the detailed of literature review of various scholars' work on text classification and sentimental analysis, whereas in the third section, we will discuss our methodology and its implementation stages. Section four is about a discussion of the outcomes of our research. Section five presents a summary, conclusion, and future work.

2. LITERATURE REVIEW

To identify Urdu news items, Imran Rasheed et al. propose using a hybrid feature selection technique (HFSA). Second, to extract key elements of Urdu documents, they frequently employed filter selection methods and Latent Semantic Indexing (LSI). On the Urdu "ROSHNI" dataset, the hybrid technique was evaluated over SVM. The findings indicate that SVM classification is more precise and effective [2]. While Syed Adnan et al used 141289 news words from eight different distinct classes (Sports, Entertainment, Armed Forces, Education, Accident, Local, International, and Weather). After analyzing numerous ML algorithms, the author determined that Ridge Classifier is the best predictor, with an accuracy of up to 87 percent [3].

Imran Rasheed et al [4] analyzed the functioning of three classifiers (DT, SVM, and KNN) using the WEKA (Waikato Environment Knowledge Analysis) program for the categorization of Urdu text. They evaluated approximately 16,678 documents, the majority of which were news pieces from the Urdu publication the Daily Roshni. The performance of SVM classifier was better as compared to others with more accurate and efficient results.

Wahab et al. [5] provided an in-depth analysis of different ML approaches for document classification. Ali et al. [6] Performed a comparison analysis of Urdu text categorization on 26,067 documents using various classifiers like NB and SVM. The results showed that SVM was more accurate compared to Naïve Bayes. They included six categories like Sports, News, Culture, Economy, Personal Communication, and Consumer Information.

Zia et al. [7] Used Naïve Bayes, KNN, and DT on two distinct Urdu datasets, EMILLE and Naive used five well-known feature selection algorithms, including CHI, GR, IG, oneR, and GR. The results demonstrate that the SVM and KNN classifiers outperform the IG technique, Bilal et al. [8] Investigated three categorization algorithms in Roman-Urdu to detect people's opinions about various objects. As far as accuracy, recall, and F-measure, NB was away better than KNN and Decision Tree. Irfan et al. discuss the various algorithm for roman Urdu text classification and present the best result by using CNN and the hybrid model approach [9]. Later the same author also experimented Logistic Regression classifier over roman Urdu data set with 93% accuracy [10].

The Roman Urdu news classification system, which categorizes news into five categories, is suggested by Rizwan Ali Naqvi and his colleagues. They compare the outcomes of several ML methods such as LG, MNB, LSTM, and CNN. The results revealed that the Multinomial Nave Bayes classifier has the best accuracy of 90.17 percent [11]. Syed Muhammad Hassan et al. explained the Roman-Urdu language news headline. There are 12319 news headlines in all, divided into seven categories. They compared findings from Perceptron, LR, MNB, LSVC, RC, PAC, NC, SGDC, and RF. SGD predicts the optimal result for identifying the desired class with a 93.50 percent accuracy rate [12].

Sentiment categorization of Urdu news on tweets is the goal of Raheela Bibi et al. They performed Pre-processing first and then created a Feature Vector. The vector contained the count of positive and negative words along with the negation value using POS Tags. Decision Tree is used on the Vector which resulted in a 90% accuracy rate [13]. Mukhtar et al. [14]- [15] suggested a supervised machine learning method for Urdu sentiment analysis. They applied KNN, DT, and SVM to data collected from 14 blogs. They found KNN to be more accurate and precise. They found it better in terms of F-Measure and Recall. They also proposed a lingual-based Urdu sentiment analyzer that was better than supervised machine learning techniques as far as accuracy, F-measure, and

precision in several domains. On an Urdu language corpus, Muhammad Usman et al. [16] presented five well-known categorization strategies and gave a class to the documents using a majority vote on 21769 news headlines organized into seven different categories (Business, Entertainment, Sports, Weird, Culture, and Health). They used tokenization, stop word removal, and a rule-based stemmer for Pre-processing, and 93400 features were taken from the data. Using majority voting, the author was able to attain up to 94 percent precision and recall.

In order to classify English text and documents, Xiaoyu Luo utilized SVM techniques [17]. They divide English language papers into two analytical sections to test the classifiers. On the small feature set Rocchio classifier delivers the best performance results, but SVM beats the other classifiers, according to experimental results on a collection of 1033 documents. According to the results of the experiment, when more than 4000 features are included, the classification rate exceeds 90%.

Shweta D. Mahajan and Dr. D.R. Ingle proposed a machine learning-based news classification methodology. They employed a news-related dataset that included a variety of data kinds such as entertainment, education, sports, politics, and so on. They used Naive Bayes plus certain word vectorizing techniques on this data to get the best results [18]. Similarly for the automatic Nepali news categorization problem, according to Tej Bahadur Shahi and Ashok Kumar [19] SVM, Naive Bayes, and Neural Networks were among the most widely used machine learning algorithms. The system was tested using a self-created Nepali News dataset with twenty separate classes and about 4964 items. TF-IDF-based features are extracted from the precompiled content to train an algorithm and test the models. According to the typical empirical findings, the SVM with RBF kernel leads the other three approaches with an accuracy of 74.65%. The linear SVM, which has a precision of 74.62 percent, is followed by the Multilayer Perceptron Neural Networks, which has a precision of 72.99 percent, and the Naive Bayes, which has a precision of 68.31 percent.

Bidi et al recommended a feature selection approach designed on Genetic Algorithms for text categorization (GA). Firstly, a thorough evaluation of several GA-based feature selection strategies, each of which includes a text representation method, is provided. Later, it gives a complete performance evaluation that improves its results utilizing known methodologies such as SVM, KNN, and NB [20]. Similarly, Khan et al. [21] propose a conditional random field-based (CRF) machine learning technique for Urdu word segmentation. Furthermore, this strategy facilitates the reduction of the compound and redundant terms. Many applications benefit from word segmentation, IR, POS, NER, sentiment analysis, and other techniques are used [22]. On the other hand, Puri and Singh [23], classified the Hindi text documents using a combination of SVM and fuzzy logic. In contrast, SVM is utilized for data classification, while fuzzy logic is used to remove ambiguity and It adapted the Latent Dirichlet technique. Anwar et al. [24] Suggested hypothetical research for rhetorical investigation in Urdu text.

3. METHODOLOGY

Our technique adopts a step-by-step approach, with the suggested model comprising numerous modules such as corpus collecting, pre-processing, feature extraction, classification algorithms, and performance assessment. The many modules employed in our system are depicted in Fig 1, and the characteristics of each module are detailed in the sections below.

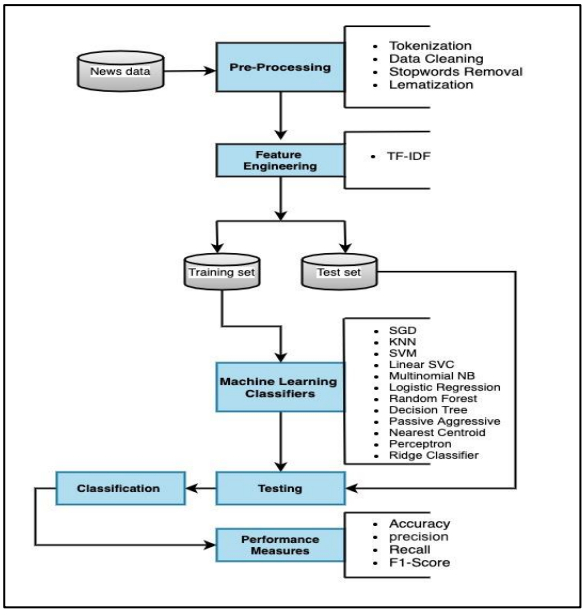

Fig 1: Framework of proposed model

3.1. Corpus collection

To provide acceptable accuracy in system classification, a standard collection is required. A standard dataset is needed for any Natural Language Processing (NLP) activity. For this objective, we use two data sets in this study. The first dataset is a self-created dataset that was

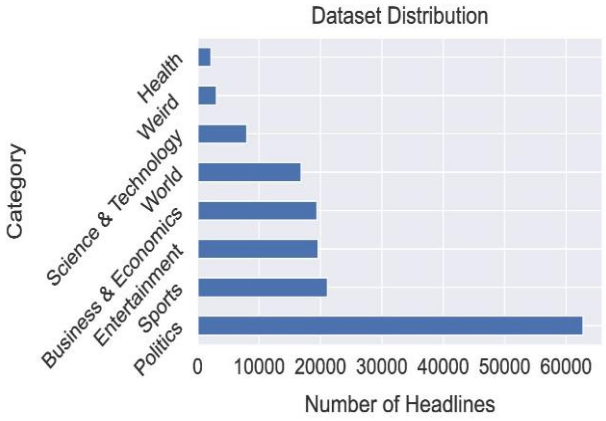

Figure 2: Data Distribution bar graph

manually generated from a range of Internet news sources, while the remainder of the data comes from publicly available sources. In Fig 2, bar graph depicting the dataset distribution according to the number of instances for different documents in each class.

There are eight classes and have over 0.15 million (153,050) text documents in the corpus. The total amount of words is 13,301,457, with 472,541 tokens. Table 1 summarizes the corpus characteristics for each class.

3.2. Preprocessing

Pre-processing is the process of normalizing and cleaning text for use in training and testing Machine Learning Models. It minimizes redundant data and removes the noise in data. This cleaned data helps in real-time news categorization and improves the overall accuracy of the classifier and makes speedier. The most common preprocessing techniques are listed below.

Table 1. Characteristics Of Corpus

Categories	Headlines	Words	Distinct words
Politics	62765	1271576	59336
Sports	21147	3784080	109212
Entertainment	19619	3601916	126676
Business & Economics	19419	2780660	79956
World	16853	330697	27603
Science &Technology	7993	1437394	52588
Weird	3074	56136	10760
Health	2180	38998	6410

3.2.1. Tokenization

Tokenization is the process of breaking down words into a sequence of characters. Words, numbers, Identifiers, and punctuation are all examples of tokens [25]. The tokenizer generates tokens from strings by reading delimiters such as /- “[] () :?<>!

To tokenize text, the NLTK (Natural Language Tool Kit) library was utilized in this project.

3.2.2. Special symbol removal

Unique symbols such as! :‘ ,o,>,,,,/,@,#,$, %, &, *,), (, _, -, +, =, [,], ‘,” etc., as well as numbers that have little significance in categorization, are deleted.

3.2.3. Stop word removal

There are several words in languages that are not helpful in any analysis but are helpful in completing the sentences like prepositions or

adjectives. These words are known as stop words. We created a list of 265 such words. Some Urdu

آ آئی آئیں آئے آتا آتی آتے کے گے سے
نے کس کوئی کون گیا یعنی یہ یہاں
یہی کا اگر کو کوئی تک تھا

Figure 3: List of Some Stop words

3.2.4. Lemmatization

A lemmatizer transforms a word's inflected surface forms into its lemma or root form, and it is closely related to the stemme [26]. To do lemmatization on our Urdu text, the Urduhack library was used, which allows us to lemmatize our Urdu text. Table 2 shows some lemmatized words.

Table 2. Some lemmatized words

Input Word	Root/Lemma
مہنگیں	مہنگا
دککتیں	دککت
پیر ھیاں	پیڑھی
سیڑھیاں	سیڑھی
نباتات	نبات

Table 3 shows an example of preprocessing procedures that we used in an article.

Table 3. Example of pre-processing

Input text	**پیٹرول کی قیمتوں میں اضافہ سٹے کی وجہ سے ہوا، ۔؟**
Tokenization	پیٹرول، کی، قیمتوں، اضافہ، سٹے، کی، وجہ، سے، ہوا، ۔؟
Data cleaning	پیٹرول، کی، قیمتوں، اضافہ، سٹے، کی، وجہ، سے، ہوا
Stop words removal	پیٹرول، قیمتوں، اضافہ، سٹے
Lemmatization	پیٹرول، قیمت، اضافہ، سٹہ

3.3. Feature Extraction

Machines are specialized in numerical data, but not so well with textual data. The term frequency-inverse data frequency (tf-idf) is one of the most frequently applied ways of analysing and expressing textual data. Sparck Jones [27] presented a method for generating a document's vector representation. Obtaining a vector representation of a text such that the distance between the vectors may be used to compare the similarity of the texts. As a result, we do a wide range of activities, including document classification, text summarization, and so forth. The general TF-IDF formula for phrase scoring is Equation 1.

$$tf - idf(t_{u,v}) \quad (1)$$

Where N is the corpus size, tf is the term frequency of term t in document d. Another term of df shows the document frequency in the number of the document where this term occurred.

3.4. Classification Algorithms

Datasets frequently contain vital information that is used to make timely decisions. Algorithms are unable to reach a conclusion without a good dataset. As a consequence, classification algorithms make the process easier by identifying pertinent models and emphasizing important data categories. K-Nearest Neighbour (KNN), Multinomial Naive Bayes (MNB), Linear Support Vector Classifier (SVC), Support Vector Machine (SVM), Decision Trees (DT), Logistic Regression (LR), Random Forest (RF), Passive-Aggressive Classifier (PAC) Nearest Centroid Classifier (NCC), Perceptron Classifier (PC), Stochastic Gradient Descent (SGD) and Ridge Classifier (RC) are text classification approaches. We utilized the sklearn library to classify models. In this study, all of the above methods were used for document categorization, and a few of the classifiers produced good results as discussed below:

SVM is a component of the supervised learning approach which is used for classification tasks. SVM utilizes the data as two sets of vectors. These vectors have an n-Dimensional Space. SVM creates a hyperplane in that space which maximizes the margin between two datasets [28]. High accuracy and resistance to overfitting are two of SVM's main features. SVM approach provides better results for text classification difficulties as compared to other classifiers due to the fact that it creates a solution fast.

LSV Technique is a type of machine learning strategy for categorizing samples into discrete classes by establishing linear boundaries (i.e., a linear hyper-plane) in a multi-dimensional space.
In the case of lower-dimensional classes, the purpose of a Linear SVC is to design a linear maximum margin among samples of various classes, thus it enhancing generalization. Slack variables are used to permit a certain number of samples to lie on the wrong side of the hyperplane, resulting in a "soft margin", because most actual datasets aren't perfectly linearly separable [29]. For a two-class linearly separable classification task, linear SVCs define a linear hyperplane such that for each sample x:

$$x^T\omega + b \geq 0 \text{ for } y = +1 \quad (2)$$

$$x^T\omega + b < 0 \text{ for } y = -1 \quad (3)$$

Another approach logistic regression works by multiplying an input value by a weight value [30].

This classifier determines which attributes in the input are most effective for differentiating between the distinct classes [31]. Logistic regression is a discriminative model, that indicates it calculates P(y|x) by distinguishing between various possible values of the classy based on input x. The following equation describes the process:

$$p(c|x) = \sum_{i=1}^{N} w_i . f_i \qquad (4)$$

The value of P (y|x) cannot be retrieved directly using the preceding approach since it will create a number spanning from - to, implying that no output between 0 and 1 will be produced. Use of exponent function to obtain a value of output in range of 0 or 1.

$$p(c|x) = \frac{1}{z} \exp \sum_{i} w_i . f_i \qquad (5)$$

Do the following to change the normalization factor Z and supply the number of features as N.

$$p(c|x) = \frac{\exp(\sum_{i=1}^{N} w_i . f_i)}{\sum_c \exp(\sum_{i=1}^{N} w_i . f_i)} \qquad (6)$$

The usage of binary-valued properties is frequent in language processing. The observation x and the candidate output class c, not just the observation x, share the same properties Fi (c, x) is used rather than fi or fi(x), with feature I from class c being assigned as the specified input of x [31]. The following is the final equation for estimating the probability of y belonging to class c given x:

$$p(c|x) = \frac{\exp\left(\sum_{i=1}^{N} w_i . f_i \ (c,x)\right)}{\sum_{c' \in C} exp\left(\sum_{i=1}^{N} w_i . f_{i_{(c',x)}}\right)} \qquad (7)$$

4. RESULTS AND DISCUSSION

4.1. Evaluation Measures

We investigated the classifier's efficiency using three key metrics: accuracy, recall, and the F-measure. The three metrics for the positive class are as follows:

Precision (P): It is computed by dividing the number of correct positive outcomes by the number of positive outcomes predicted by the classifier

$$p = \frac{\text{No. of correct positive predictions}}{\text{No. of positive predictions}} \qquad (8)$$

Recall (R): It is derived by dividing the total number of relevant samples by the number of genuine positive findings.

$$R = \frac{\text{No. of correct positive predictions}}{\text{No. of positive examples}} \qquad (9)$$

The F1 Score is the Harmonic Mean of accuracy and recall. The F1 Score has a value between 0 and 1. It tells you how precise and robust your classifier is (how many instances it successfully classifies).

$$F = \frac{2 \times P \times R}{P + R} \qquad (10)$$

4.2. Result Analysis

We used a machine learning approach to train our dataset. Our dataset is divided into two parts, we train on 80% data and test on 20%.

Fig 4 displays the model comparison graph and similarly Fig 5 shows the performance measures of our models with various Machine Learning Algorithms such as K-Nearest Neighbour, Support Vector Machine, Multinomial Naive Bayes, Linear Support Vector Classifier, Decision Trees, Logistic Regression, Random Forest, Passive-Aggressive Classifier, Nearest Centroid Classifier, Perceptron Classifier, Stochastic Gradient Descent, and Ridge Classifier. Except for the Random Forest model, which yields just 63.98 percent accuracy, the results range from 73.35 % to 91.37 %. We discovered that the SVM classifier and Linear SVC produced nearly identical results (91.37 percent vs. 91.35 percent). Similarly, the results of Logistic Regression and SGD are nearly identical at 90.25 % and 90.24 %, respectively. These Classifiers outperform the rest of the algorithms with distinct modifications in all categories based on the strategies they employ.

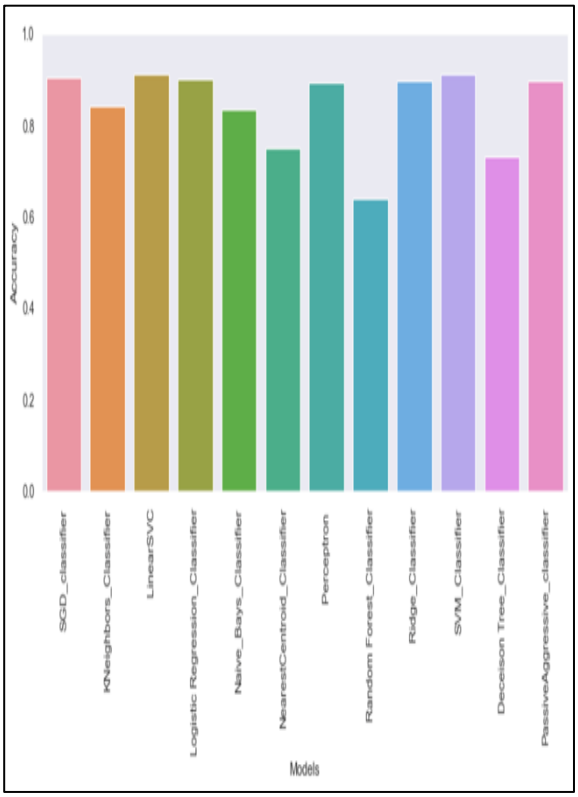

Fig 4: Basic models comparison

Table 4. Performance measures of models

	Models	Accuracy	Precision	Recall	F1-Score
1	SGD_classifier	0.903692	0.862983	0.792199	0.816395
2	KNeighbors_Classifier	0.843777	0.766087	0.744840	0.746582
3	LinearSVC	0.913525	0.858494	0.829578	0.842527
4	Logistic Regression_Classifier	0.902483	0.865665	0.787842	0.815650
5	Naive_Bays_Classifier	0.834107	0.814414	0.642431	0.652833
6	NearestCentroid_Classifier	0.751617	0.703275	0.758664	0.688834
7	Perceptron	0.893989	0.823593	0.803983	0.813107
8	Random Forest_Classifier	0.639791	0.545527	0.353327	0.378877
9	Ridge_Classifier	0.898040	0.855247	0.783481	0.808914
10	SVM_Classifier	0.913656	0.859239	0.829922	0.843033

In this case, SVM and Linear SVC classifiers have a greater influence than others since they provide more than 90% of the measured matrix than other classifiers that identify only a few categories above the 90% threshold. According to the supplied matrix, it is not necessary for all models to perform well; we discuss a few model matrices whose results are clearly understandable. Our methodology divides the dataset into training and testing, allowing us to investigate the primary reasons for misclassification on the test set.

The confusion matrix, which is based on projected and actual label differences, is a major source of error identification. Fig 6 SVM Confusion Matrix depicts an accurate prediction on the diagonal side, with the proper label of the Business & Economics category being 294, Entertainment 3691, Health 3697, Politics 1411, Science & Technology 4124, Sports 11898, Weird 280, and World 2572. However, Fig 7 shows the right predictions for eight categories, with Business & Economics at 294, Entertainment at 3691, Health at 3696, Politics at 1410, Science & Technology at 4125, Sports at 11897, Weird at 289, and World at 2571. Further Fig 8 shows an actual and expected matrix in which Business & Economics, Entertainment, Health, Politics, Science & Technology, Sports, Weird, and World are all represented (246, 3627, 3619, 1397, 4098, 11990, 195, and 2453).

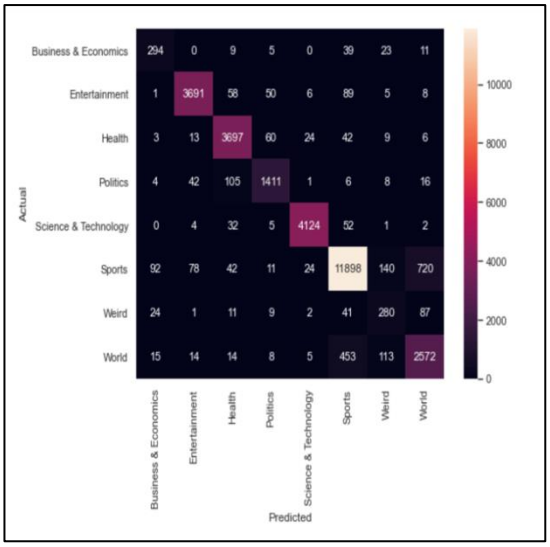

Figure 6: SVM Confusion Matrix

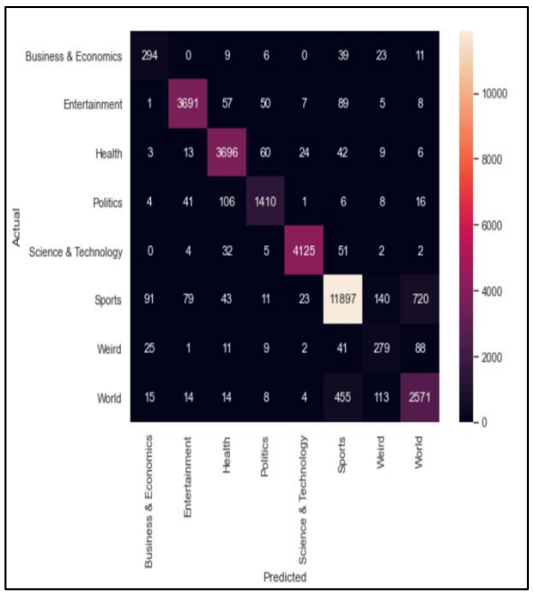

Figure 7: Linear SVC Confusion Matrix

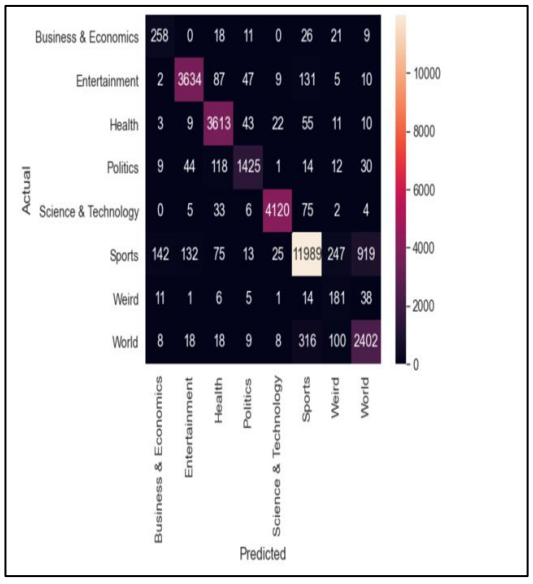

Figure 8: LR Confusion Matrix

5. CONCLUSION

The digital world's massive unstructured data is one of the venues where various machine learning approaches can be applied. Text categorization on Urdu news headlines was performed in this study utilizing twelve machine learning techniques. In addition, the TF-IDF model was used to analyze textual aspects. SVM and Linear SVC performed well, with the accuracy value not fluctuating too much. The trials' positive results suggest that news headlines can be relied on to forecast the type of news. This observation is significant since headlines are short sentences that require few computer resources to operate on.

Future recommendations for improving this work include extracting features from text documents and developing the fasttext and Word2Vec embedding approaches. Deep learning-based classification algorithms such as RNN (LSTM, BiLSTM) can also be used to solve the Urdu news categorization challenge.

References

[1] M. Iqbal, B. Tahir and M. A. Mehmood, "CURE: Collection for urdu information retrieval evaluation and ranking," in *2021 International Conference on Digital Futures and Transformative Technologies (ICoDT2*, IEEE, 2021, pp. 1-6.

[2] I. Rasheed, H. Banka and H. M. Khan, "A hybrid feature selection approach based on LSI for classification of Urdu text," in *Machine Learning Algorithms for Industrial Applications*, Springer, 2021, pp. 3-18.

[3] S. A. A. Zaidi and S. M. Hassan, "Urdu/Hindi News Headline, text classification by using different machine learning algorithms".

[4] I. Rasheed, V. Gupta, H. Banka and C. Kumar, "Urdu text classification: A comparative study using machine learning techniques," in *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*, IEEE, 2018, pp. 274-278.

[5] W. Khan, A. Daud, J. A. Nasir and T. Amjad, "A survey on the state-of-the-art machine learning models in the context of NLP," *Kuwait journal of Science,* vol. 43, no. 4, pp. 95-113, 2016.

[6] A. R. Ali and M. Ijaz, "Urdu text classification," in *Proceedings of the 7th international conference on frontiers of information technology*, 2009, pp. 1-7.

[7] T. Zia, M. P. Akhter and Q. Abbas, "Comparative study of feature selection approaches for Urdu text categorization," *Malaysian Journal of Computer Science,* vol. 28, no. 2, pp. 93-109, 2015.

[8] M. Bilal, H. Israr, M. Shahid and A. Khan, "Sentiment classification of Roman-Urdu opinions using Na{\"\i}ve Bayesian, Decision Tree and KNN classification techniques," *Journal of King Saud University-Computer and Information Sciences,* vol. 28, no. 3, pp. 330-344, 2016.

[9] I. Qutab, K. I. Malik and H. Arooj, "Sentiment Analysis for Roman Urdu Text over Social Media, a Comparative Study," *International Journal of Computer Science and Network,* vol. 9, no. 5, pp. 217-224, 2020.

[10] I. Qutab, K. I. Malik and H. Arooj, "Sentiment Classification Using Multinomial Logistic Regression on Roman Urdu Text," *International Journal of Innovations in Science & Technology,* vol. 4, no. 2, pp. 223-335, 2022.

[11] R. A. Naqvi, M. A. Khan, N. Malik, S. Saqib, T. Alyas and D. Hussain, "Roman Urdu news headline classification empowered with machine learning," *Computers, Materials and Continua,* vol. 65, no. 2, pp. 1221-1236, 2020.

[12] S. M. Hassan, F. Ali, S. Wasi, S. Javeed, I. Hussain and S. N. Ashraf, "Roman-urdu news headline classification with ir models using machine learning algorithms," *Indian Journal of Science and Technology,* vol. 12, no. 35, pp. 1-9, 19.

[13] R. a. Q. U. a. A. M. a. S. A. Bibi, "Sentiment analysis for Urdu news tweets using decision tree," in *2019 IEEE 17th International Conference on Software Engineering Research, Management and Applications (SERA)*, IEEE, 2019, pp. 66-70.

[14] N. Mukhtar, M. A. Khan and N. Chiragh, "Lexicon-based approach outperforms Supervised Machine Learning approach for Urdu Sentiment Analysis in multiple domains," *Telematics and Informatics,* vol. 35, no. 8, pp. 2173-2183, 2018.

[15] N. Mukhtar and M. A. Khan, "Urdu sentiment analysis using supervised machine learning approach," *International Journal of Pattern Recognition and Artificial Intelligence,* vol. 32, no. 02, p. 1851001, 2018.

[16] 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.

[17] X. Luo, "Efficient english text classification using selected machine learning techniques,"

Alexandria Engineering Journal, vol. 60, no. 3, pp. 3401-3409, 2021.

[18] S. D. Mahajan and D. Ingle, "News Classification Using Machine Learning," *International Journal of Innovative Science and Research Technology*, vol. 6, no. 5, pp. 873-877, 2021.

[19] T. B. a. P. A. K. Shahi, "Nepali news classification using Naive Bayes, support vector machines and neural networks," in *2018 International Conference on Communication Information and Computing Technology (ICCICT)*, IEEE, 2018, pp. 1-5.

[20] N. Bidi and Z. Elberrichi, "Feature selection for text classification using genetic algorithms," in *2016 8th International Conference on Modelling, Identification and Control (ICMIC)*, IEEE, 2016, pp. 806-810.

[21] S. N. Khan, K. Khan, A. Khan, A. Khan, A. U. Khan and B. Ullah, "Urdu word segmentation using machine learning approaches," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 6, pp. 193-200, 2018.

[22] K. I. Malik, Q. Abbas, H. Arooj, S. Niazi, Y. Saeed, I. Asghar and M. Ilyas, "Classical and Probabilistic Information Retrieval Techniques: An Audit," *Lahore Garrison University Research Journal of Computer Science and Information Technology*, vol. 5, no. 3, pp. 84-91, 2021.

[23] S. Puri and S. P. Singh, "Hindi text document classification system using SVM and fuzzy: A survey," *International Journal of Rough Sets and Data Analysis (IJRSDA)*, vol. 5, no. 4, pp. 1-31, 2018.

[24] W. Anwar, I. S. Bajwa, M. A. Choudhary and S. Ramzan, "An empirical study on forensic analysis of Urdu text using LDA-based authorship attribution," *IEEE Access*, vol. 7, pp. 3224-3234, 2018.

[25] M. a. A. P. Karthikeyan, "Probability based document clustering and image clustering using content-based image retrieval," *Applied Soft Computing*, vol. 13, no. 2, pp. 959-966, 2013.

[26] M. Humayoun, R. M. A. Nawab, M. Uzair, S. Aslam and O. Farzand, "Urdu summary corpus," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, 2016, pp. 796-800.

[27] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of documentation*, vol. 60, no. 5, pp. 493-502, 2004.

[28] Q. Gu and Z. Song, "Image classification using SVM, KNN and performance comparison with logistic regression," *CS44 Final Project Report*, 2009.

[29] S. Lakhotia and X. Bresson, "An experimental comparison of text classification techniques," in *2018 International Conference on Cyberworlds (CW)*, IEEE, 2018, pp. 58-65.

[30] S. Raschka, Python machine learning, Packt publishing ltd, 2015.

[31] S. Indra, L. Wikarsa and R. Turang, "Using logistic regression method to classify tweets into the selected topics," in *2016 international conference on advanced computer science and information systems (icacsis)*, IEEE, 2016, pp. 385-390.