1st.International Conference of Information Technology to enhance E-learning and other Application,
(IT-ELA 2020), Baghdad College of Economic Sciences University, Baghdad, Iraq

Rumor Detection on Twitter Using Features Extraction Method

Hussam Mohammed Jabir
Dept.of Computer Science
University of Babylon
Babylon, Iraq
hussamj1.pc@gmail.com

Mohammed Abdullah Naser
Dept.of Computer Science
University of Babylon
Babylon, Iraq
wsci.mohammed.abud@uobabylon.edu.iq

Safaa O. Al-mamory
College of Business Informatics
University of Information Technology and Communications
Baghdad, Iraq
salmamory@uoitc.edu.iq

***Abstract*— Rumors can widely and rapidly diffuse in social networks than in offline platforms. This work was an attempt to address this challenge by introducing an approach for rumors detection, which learns from the Twitter dataset. In other words, this paper is going to suggest new features were extracted from Twitter datasets based on user behavior, the propagation features and the temporal features that representing the social context to reduce variation in features. Also, the proposed work involves adopting an ensemble classifier as a more appropriate approach and for achieving a better performance compared to existing individual-based classifiers. Moreover, the proposed method processes unbalanced data sets to reduce their impact on detection algorithms and improve the classification accuracy rate. Experimental results on the PHEME dataset showed that the new features made the classification process more effective and accurate compared to many related works of rumor classification that use the same data. Experimental results achieved 78.54% accuracy as an average of the events used.**

Keywords— Rumor detection, Social media, Feature Extraction, Classifier ensemble, Classification.

I. INTRODUCTION

Social media is a place where extensive data is continuously generated. These days, on microblogs, novel breaking information appears first, before being posted in the usual known means [1]. Hence, microblogging websites are wealthy sources concerning data that have been successfully leveraged because of the evaluation of social phenomena, like notions, opinions, and sentiments between online conversations [2].

Online social sites have had great popularity in recent times, such as Twitter, Facebook, and Google+. Twitter is a microblogging situation counts along thousands and thousands of customers from all around the world [3]. Therefore, Twitter is a real-time method to spread information. In the last years, twitter has been used to spread the news concerning problems like earthquakes, epidemics, disasters, etc. Telecommunication channels had power failure under some of these situations, while the Internet and some platforms of social networks such as twitter worked more efficiently [4]. However, Twitter does not only give benefits. It has its negative impacts as well, like rumors. The increasing number of tweets raises the probability of rumors generated. These rumors might have serious negative effects and prevent the help required in emergency cases. A rumor is a false information transferred widely without a credited source [5].

Rumor detection can formulate as a classification project aiming to determine the credibility of the social media story and if it is true or false. There are two challenges in rumor detection: (1) the comprehension of contents and context associated with the events and (2) the designing of efficient detection algorithms [6]. Tweets have common features notified by previous work (e.g. created-at, source, is-retweet, favourite-count, followers-count, country-code, URL…etc.) [7]. Our approach performs new specific features designed by capturing information about the user's reaction to tweets and express confidence between them. Key features that extracted enable advanced algorithms to find out beneficial context information led to the development of a huge variety of algorithmic strategies [8]. These strategies rely on the extract or convert from the original data. In addition to the classification method, some forms of data compression that made it feasible in lots of cases to apply more advanced algorithms to deal with context features [9].

In the literature, there are many models and strategies for rumor detection based on work of the classifier, which relies on trained of the annotated dataset to recognize polarity between phrases of veracity, rumor or non-rumor direction [10]. Very few works focused on rumor detection using ensemble classifiers [11]. Nevertheless, given the effective advantages of the assignment of rumor mining on publicly available data, ensemble classifiers are an appropriate approach to use to increase the individual base classifiers' accuracy. In this research, the classifier ensemble approach in rumor identification is proposed [12]. It consists of three individual base classifiers Known as a support vector machine (SVM) [13], K-Nearest Neighbors (KNN) [14] and Naive Bayes (NB) [15] to decide whether an event is a rumor or not based on a set of tweets about an event [16]. Our results showed that adding new features and using the mentioned classification approaches gave a higher performance on benchmarked datasets, compared to recent state-of-the-art systems [17].

II. RELATED WORKS

This section focuses on providing a brief review of the work most closely to our study. We will outline the related works in three main areas: data collection and annotations, rumor analysis, and features for classification.

978-1-7281-8233-9/20/$31.00 ©2020 IEEE

A. Methods for Data Collection and Annotation

Ahmet Aker et al. [17], explained a classification approach to open position classification in Twitter to rank rumors. The approach took advantage of a new set of specific features of problems that could be determined automatically, which greatly enhanced the accuracy of the classifier.

Elena Kochkina et al. [18], Suggested a multi-tasked study approach, which permitted joint training of the main and additional tasks, enhancing rumor detection performance. The automatic accuracy of rumors was divided into smaller pipeline components, including detection, tracking, and stance classification, leading to the result of defining the validity of rumors.

Sooji Han et al. [19], worked on class imbalance and limited labelled data problem for rumor detection based on machine learning on social media. They presented an offline data augmentation method based on semantic correlation for rumor detection. A neural language that understands the context model and authenticity-focused Twitter corpus was used to learn the personification of rumor tweets to measure the semantic correlation.

B. Analyzing Rumors

Jing Ma et al. [20], Introduced a new way that learns continuous representations of microblog stories to identify rumors, relying on Repeated Neural Networks (RNN) to learn hidden representations that capture the diversity of contextual information for important posts over time.

Fang Jin et al. [21], worked on epidemiological models to describe the chain of information on Twitter, generated by both news and rumors. SEIZ Enhanced Epidemic model is used. It recognized skeptics to distinguish eight events worldwide and covered different types of events.

Dung T. Nguyen et al. [22], considered the k−Suspected problem which plans to identify the top k most suspected sources of false information. They suggested two effective approaches: classification and optimization-based algorithms.

C. Features for Classification

Fan Yang et al. [23], examined the problem of reliability of the information on Sina Weibo. They collected a wide range of microblogs that was confirmed as false rumors based on Sina Weibo's service. They then examined an extensive set of features and trained a classifier to spot rumors automatically.

Sejeong Kwon et al. [5], diagnosed characteristics of rumors by examining three diffusion attitudes: temporal, structural, and linguistic. For the temporal characteristics, they suggested a new periodic time sequence model, which considers daily and external shock cycles. They also established differences in key structure and linguistic in the diffusion of rumors and non-rumors.

Sardar Hamidian and Mona Diab [8] discussed two common matters. First, they detected rumors as a sort of false information propagation, and second, they classified rumors. That is Rumor Detection Classification (RDC). They analyzed the issue using a standard data set, created new features, and studied their effect on the test.

III. The Proposed System

In this section, a suggested system is presented to address the problem of rumor detection and classification within the context of microblog social media. The proposed work focused on Twitter data due to the availability of annotated data in this genre, in addition to the above-mentioned interesting characteristics of microblogging, and their specific relevance to rumor proliferation. In the proposed system, a set of new features was added to increase the accuracy of rumor detection. The strategy of features ranking and instances selection were used as inputs to the classification mechanism. The Ensemble method was used for a group of various classification algorithms to obtain the highest results. The description of representative parts of the proposed system as follows:

D. Data

The original corpus was gathered from Twitter social network platform (www.twitter.com) [24]. It includes a massive quantity of tweets with all tweets that reply to them (retweets). Five special newsworthy events were selected, all of which attracted huge activity in the media and were diffused with rumors: Ferguson unrest, Ottawa shooting, Sydney siege, Charlie Hebdo shooting, and German wings plane crash. Tweets were gathered by determining rumors of interest and afterwards filtering on keywords and related hashtags in Twitter Streaming API. Each tweet annotated manually to rumor and non-rumor is reported by the staff of journalists based on their experience. Zubiaga et al. (2016) described the rumor annotation execution, which was tackled by journalists. The tweet annotation sampled for five events brought a collection of 5,802 tweets with their annotation, of which 1,972 considered rumors and 3,830 considered non-rumors. These annotations were dispensed differently throughout the five events, as Table I illustrates. These data sets are publicly available called the PHEME dataset,[1] which contains the five events mentioned above, and they are classified as rumor and non-rumor, and this is our field of research [25].

TABLE I. Distribution of rumor and non-rumor annotations for the datasets

Event	No. of tweets	No. of rumor	No. of non-rumor
Charlie Hebdo	2,079	458 (22.0%)	1,621 (78.0%)
Ferguson	1,143	284 (24.8%)	859 (75.2%)
Germanwings Crash	469	238 (50.7%)	231 (49.3%)
Ottawa Shooting	890	470 (52.8%)	420 (47.2%)
Sydney Siege	1,221	522 (42.8%)	699 (57.2%)
Total	**5,802**	**1,972 (34.0%)**	**3,830 (66.0%)**

E. New Extracted Features

Diverse features were investigated on rumors detection in previous works, which classified into many categories; each category was performed in different approaches. This work focuses on social context features, which considers the relationships and connections amongst different users and reflects the process of rumors propagating, so it was extracted from the tweets temporal features, the tweets' propagation features, and the users' behavior.

Here, a set of features was utilized to perform the classification method. These contained several features that were used on the twitter platform. In our experiment, new features were proposed, which were not previously

suggested, and better ones were chosen that could enhance the accuracy of rumors determining results. Below, a detailed description of the new proposed features.[1]See https://figshare.com/articles/PHEME_dataset_of_ rumours_and_non-rumours/4010619 for PHEME dataset. h

- Rumor spread: Refers to the number of tweets that belong to different users, and it's considered as a response to the source tweets. The number of tweets that were suddenly burst, and it's spread widely. In our proposed system, tweets spread could be obtained from the number of reaction tweets, which have the same id of source tweets.
- Average rumor speed: Refers to tweets response rate per unit time. The average equation of the features is defined by "(1)"

$$\text{Average rumor speed} = \frac{\sum_{i=1}^{n}(R_i - C_i)}{n} \quad (1)$$

This average rumor speed could be operationalized by considering those tweets that were identified as rumors and there reaction was denoted by ***n***, the created time of source tweet denoted ***C*** and reaction time denoted by ***R***.

- Which day: Tweets have a created date, so the day of source tweets could be extracted and added as a feature to the dataset.
- Which month: consider the month of tweets' post, and it was determined from tweets to start date.
- Working days: Tweet launch date is verified with the calendar to determine if the day was a working day or a holiday, and return the binary results as a new feature.

F. Classification Methods

In the proposed work, attempts to gain better performance in rumors classification were made. Ensemble Classifier technique with appropriate features reachable is used for this objective. This section aims to evaluate the scope of which homogeneous ensembles could progress the effectiveness of the selection system, ensuring a satisfactory change-off between the stability of choice and predictive accuracy. The block diagram of the proposed rumor detection system is given in fig. 1.

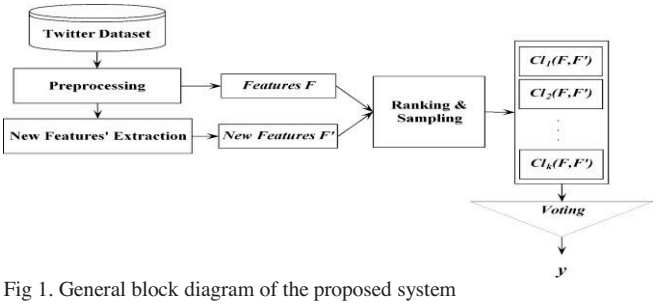

Fig 1. General block diagram of the proposed system

The set of supervised algorithms Cl used in voting is considered to be disjointed. The voting equation for the scheme proposed is shown in "(2)" where N_c is the highest number of votes, for instance, t.

$$y = argmax\left(N_c\left(Cl_1^t(F,F')\right), N_c\left(Cl_2^t(F,F')\right), \ldots, N_c\left(Cl_k^t(F,F')\right)\right) \quad (2)$$

The figure below shows the process of preparing the data used in the experiments. Moreover, the newly extracted features that previously mentioned adding to the original data. Besides, the desired value traits are selected from this data. Indeed, training data that was determined by N features, the features were arranged according to their importance and relevance, from the most important to the least important. A feature weighting could be sorted by the weights and converted to a feature ranking. Also, the sampling technique, which can be defined as several samples with replacements (boot-straps) is selected from the input dataset according to the distribution of uniform probability. These ranked features and selected samples would be the input for the ensemble schema. Three-machine learning algorithms in Ensemble Voting Classifiers were used, which are: 1) a K-Nearest Neighbors (KNN), 2) a Naive Bayes (NB) and 3) a support vector machine (SVM) to obtain the best output from the data set. Then the majority voting classifier would be utilized, which specified the majority class from the training datasets that were used in the tested data. In other words, each classifier used will be predicted by the class label for the test sample. The label that was most predicted will be selected as a voting classifier output.

IV. Experimental Results

All experiments prepared, designed, implemented, and evaluated based on different experimental settings and situations, are explained in this section.

G. Experimentation Frameworks

For this system, preferential experiment filters were performed, and after that, classification algorithms support diverse kinds of attributes. WEKA platform was used for training and testing the suggested models in our system. All results were obtained by using a Lenovo system core™i5-3320M, RAM 8G Windows 10 Pro 64-bit.

H. Domain Training Data

The main dataset was formed of five datasets. Each dataset has two classes, rumor and non-rumor. All data were gathered from the previously mentioned five real events. Each one of the five datasets was split into 66% for training and the rest for testing. Thirty-five attributes were removed because they were empty. Text attribute was removed as well, which contains the main tweet and relied on context features [25]. So the rest were 55 attributes. Table II presents the improvement in an average of mean and standard deviation for multi classifiers after removing text attribute.

TABLE II. Comparison of accuracy of an average of mean and standard deviation for multi classifiers for datasets with/without text features.

Original 55 features for PHEME data	Mean ± SD of The K-NN, Naive Bayes, SVM	
	With text attribute	Without text attribute
Charliehebdo	74.03± 12.21	74.32± 12.01
Ferguson	93.21± 4.67	93.52± 4.15

Germanwings-crash	60.10± 1.11	60.10± 1.11
Ottawa shooting	73.63± 1.91	73.52± 1.72
Sydney siege	69.85± 1.96	70.10± 2.40
Average	**74.16 ± 4.37**	**74.31 ± 4.27**

I. Feature Analysis

To study better, the impact of diverse categories of features on distinguishing the credibility of rumors related tweets; several classifiers were used to identify the most important features. This proposed method performed two sets of experiments. In the first experiment set, train the classifiers using a dataset with the previously suggested features to understand how well those features subset implement in Twitter rumor detection. In the second experiment set, the impact of combining each of the five newly proposed features was evaluated. Then, the experiment run with all these new five features with each other added to the set of features, to identify the effect of their contribution.

PHEME dataset results are shown in Table III, which shows achieving a higher accuracy by adding all-new features, which were applied by using a dataset contained context features. In the same table III, the results of using the original features are shown starting from the less feature to the top, and without adding new features. The arrangement of the importance of features is implemented through the ranking attribute filter.

TABLE III. COMPARISON OF ACCURACY OF MAJORITY VOTING FOR PHEME DATASET WITH/WITHOUT THE NEW FEATURES.

No.	Dataset	Original features	Original + new features
1	Charliehebdo	79.3605	81.1047
2	Ferguson	96.2963	99.3827
3	Germanwings-crash	60.5839	62.7737
4	Ottawa shooting	76.7123	76.3699
5	Sydney siege	70.603	73.1156

In each dataset, results were checked after using multi classifiers. The mean of these classifiers of each topic was calculated, and the average of the mean of the five topics was taken and compared with the results of the same operation. After adding the new features mentioned above The dataset, showed an increase in the results in the final average for all topics. The addition of the new features led to an overall increase in performance, compared to the average of mean accuracy of the same classifiers on the original dataset only, as shown in fig.2. When we compare the results of accuracy by using the average of standard deviation accuracy, the difference is presented in fig.3.

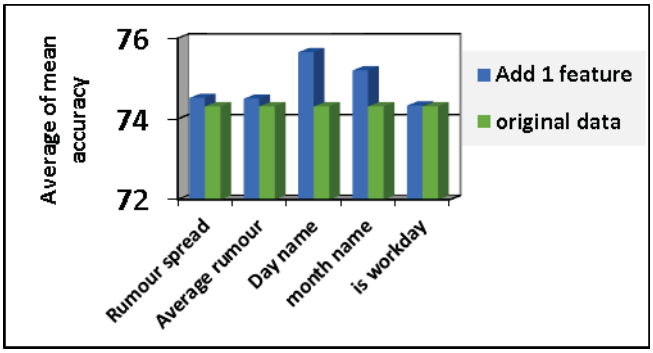

Fig 2. The Average of mean accuracy for multi classifiers employed for one feature at a time compared with original data.

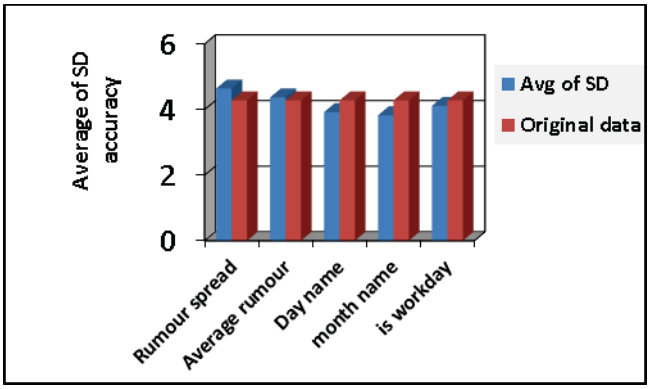

Fig 3. Accuracy scores by features using the average of standard division (gray, original data) vs adding one feature by using multi classifiers.

J. Classification Performance

The proposed work used a classifier ensemble strategy to combine the three individual leaner algorithms in a majority voting method to obtain the classification result. Some classifiers, to create an ensemble approach, were addressed in various ways and different techniques to construct models, and to achieve diversity. If by an independent decision, errors were made by base classifiers, the certainly outperform of majority voting, the overall performance of the individual classifier as the best one. The formula of ensemble classifier consists of Naive Bayes, K-NN, and Support Vector Machines designed to distinguish the polarity of the topics class.

In order to solve binary class dataset imbalance, where there was a big difference in representative data class size in many actual events, the sampling method proposed to deal with this situation was by resampling the dataset with replacement and then determining specified samples [26]. Furthermore, features ranked according to their importance, the total of the used features were 59 features. Therefore, the percentage of ranked features ended with 60 %. The percentage of resampling and ranking was increased to obtain the best results, and the increase in metrics started after 70% of resampling for all five events. Weka platform was used to apply features ranking by using Gain Ratio Attribute Evaluator, and Ranker search method and instances resampling were performed with replacement and checking of the results.

Tables IV, V are the results of the two events and are considered as examples of the classification method. These tables display the increase in using features ranking and

instances sampling with ensemble approach and majority voting system on original data plus new features.

The amount of samples is determined, then the features increase by 10% until the total number of features is reached. After that, the samples are increased by 10%, and then the same process for the features is repeated. Thus all the existing data are tested to determine the ideal result.

TABLE IV. COMPARATIVE ANALYSIS OF THE INCREASE IN METRICS ACCURACY, PRECISION, RECALL, AND F-MEASURE IN USING RESAMPLING AND RANKING FOR ENSEMBLE CLASSIFICATION APPROACH FOR CHARLIEHEBDO EVENT.

Charlie Hebdo		Metrics	Percentage of attribute selection					
			10 %	*20 %*	*30 %*	*40 %*	*50 %*	*60 %*
Percentage of sampling	10 %	Accuracy	86.9565	86.9565	84.058	91.3043	**92.7536**	**92.7536**
		Precision	0.852	0.852	0.863	0.907	**0.923**	**0.923**
		Recall	0.870	0.870	0.841	0.913	**0.928**	**0.928**
		F-measure	0.858	0.858	0.850	0.908	**0.921**	**0.921**
	20 %	Accuracy	75.1825	76.6423	78.8321	78.8321	78.8321	79.562
		Precision	0.714	0.734	0.772	0.772	0.772	0.782
		Recall	0.752	0.766	0.788	0.788	0.788	0.796
		F-measure	0.717	0.722	0.751	0.751	0.751	0.762
	30 %	Accuracy	77.1845	81.5534	82.5243	81.5534	80.0971	81.068
		Precision	0.767	0.806	0.818	0.822	0.809	0.812
		Recall	0.772	0.816	0.825	0.816	0.801	0.811
		F-measure	0.769	0.809	0.821	0.818	0.805	0.811
	40 %	Accuracy	81.4545	82.5455	82.1818	81.8182	80.7273	81.8182
		Precision	0.801	0.811	0.809	0.804	0.793	0.806
		Recall	0.815	0.825	0.822	0.818	0.807	0.818
		F-measure	0.806	0.815	0.813	0.809	0.798	0.810
	50 %	Accuracy	82.5581	84.0116	84.3023	85.1744	84.8837	84.8837
		Precision	0.835	0.833	0.843	0.850	0.846	0.846
		Recall	0.826	0.840	0.843	0.852	0.849	0.849
		F-measure	0.829	0.835	0.843	0.851	0.847	0.847
	60 %	Accuracy	86.4407	85.4722	83.293	83.0508	83.0508	83.0508
		Precision	0.864	0.861	0.848	0.848	0.848	0.848
		Recall	0.864	0.855	0.833	0.831	0.831	0.831
		F-measure	0.864	0.857	0.839	0.837	0.837	0.837
	70 %	Accuracy	84.6154	86.9023	85.447	86.9023	86.4865	85.2391
		Precision	0.856	0.875	0.872	0.890	0.887	0.879
		Recall	0.846	0.869	0.854	0.869	0.865	0.852
		F-measure	0.850	0.871	0.860	0.875	0.871	0.860
	80 %	Accuracy	89.8182	89.8182	89.6364	88.3636	88.1818	88.1818
		Precision	0.897	0.895	0.895	0.884	0.883	0.883
		Recall	0.898	0.898	0.896	0.884	0.882	0.882
		F-measure	0.897	0.896	0.896	0.884	0.882	0.882
	90 %	Accuracy	88.5299	87.7221	86.7528	86.5913	86.5913	86.7528
		Precision	0.887	0.878	0.871	0.868	0.868	0.869
		Recall	0.885	0.877	0.868	0.866	0.866	0.868
		F-measure	0.886	0.878	0.869	0.867	0.867	0.868
	100 %	Accuracy	89.5349	**90.5523**	88.9535	90.2616	87.5	87.5
		Precision	0.895	**0.904**	0.893	0.903	0.885	0.885
		Recall	0.895	**0.906**	0.890	0.903	0.875	0.875
		F-measure	0.895	**0.905**	0.891	0.903	0.879	0.879

As shown in Table IV, the original data contained numerous instances with high imbalance class in 1621 (78.0%) as non-rumor and 458 (22.0%) as a rumor. After comparison with the results of the original data, the increase in the metrics occurred after the number of samples exceeded half of the total samples. According to the results shown in such a quantity of data, the highest results occurred at 10% of the total samples, when all features included in the classification process.

Table V shows the results of operations the previously described operations that were performed on Germanwings Crash event that included samples with relatively little imbalance in trained data. It contained 231 (49.3%) for non-rumor and 238 (50.7%). The highest results in the measures used were after the increase in samples exceeded 70% of the total samples. This ratio is the highest in reviewed metrics results among the five events, and therefore it was adopted on all data cases for all topics.

TABLE V. COMPARATIVE ANALYSIS OF THE INCREASE IN METRICS ACCURACY, PRECISION, RECALL, AND F-MEASURE IN USING RESAMPLING AND RANKING FOR ENSEMBLE CLASSIFICATION APPROACH FOR GERMANWINGS CRASH EVENT.

German wings		Metrics	Percentage of attribute selection					
			10 %	*20 %*	*30 %*	*40 %*	*50 %*	*60 %*
Percentage of sampling	10 %	Accuracy	28.5714	35.7143	42.8571	50	50	35.7143
		Precision	0.143	0.371	0.457	0.458	0.510	0.371
		Recall	0.286	0.357	0.429	0.533	0.500	0.357
		F-measure	0.190	0.347	0.405	0.500	0.503	0.347
	20 %	Accuracy	37.037	33.3333	37.037	37.037	44.4444	44.4444
		Precision	0.301	0.270	0.369	0.341	0.444	0.444
		Recall	0.370	0.333	0.370	0.370	0.444	0.444
		F-measure	0.302	0.278	0.365	0.332	0.429	0.429
	30 %	Accuracy	46.3415	41.4634	65.8537	63.4146	65.8537	63.4146
		Precision	0.448	0.429	0.685	0.667	0.656	0.632
		Recall	0.463	0.415	0.659	0.634	0.659	0.634
		F-measure	0.451	0.413	0.657	0.631	0.656	0.633
	40 %	Accuracy	53.7037	57.4074	68.5185	66.6667	72.2222	68.5185
		Precision	0.544	0.580	0.701	0.672	0.722	0.686
		Recall	0.537	0.574	0.685	0.667	0.722	0.685
		F-measure	0.531	0.572	0.681	0.666	0.722	0.685
	50 %	Accuracy	63.2353	67.6471	67.6471	67.6471	61.7647	66.1765
		Precision	0.703	0.689	0.682	0.717	0.648	0.679
		Recall	0.632	0.676	0.676	0.676	0.618	0.662
		F-measure	0.636	0.681	0.679	0.682	0.624	0.667
	60 %	Accuracy	51.2195	67.0732	69.5122	68.2927	70.7317	69.5122
		Precision	0.517	0.669	0.695	0.685	0.709	0.698
		Recall	0.512	0.671	0.695	0.683	0.707	0.695
		F-measure	0.513	0.669	0.695	0.683	0.708	0.696
	70 %	Accuracy	70.8333	72.9167	73.9583	75	77.0833	76.0417
		Precision	0.711	0.743	0.747	0.753	0.771	0.760
		Recall	0.708	0.729	0.740	0.750	0.771	0.760
		F-measure	0.702	0.719	0.733	0.746	0.769	0.759
	80 %	Accuracy	70.6422	77.0642	79.8165	77.9817	74.3119	75.2294
		Precision	0.727	0.773	0.798	0.783	0.748	0.759
		Recall	0.706	0.771	0.798	0.780	0.743	0.752
		F-measure	0.705	0.768	0.798	0.780	0.744	0.753
	90 %	Accuracy	76.4228	76.4228	78.8618	78.8618	78.0488	79.6748
		Precision	0.765	0.776	0.789	0.789	0.780	0.797
		Recall	0.764	0.764	0.789	0.789	0.780	0.797
		F-measure	0.764	0.764	0.789	0.789	0.780	0.796
	100 %	Accuracy	81.7518	83.2117	86.1314	86.1314	86.1314	**87.5912**
		Precision	0.819	0.841	0.862	0.862	0.862	**0.876**
		Recall	0.818	0.832	0.861	0.861	0.861	**0.876**
		F-measure	0.817	0.831	0.861	0.861	0.861	**0.876**

In the second topic represented in Ferguson event that consists of 859 (75.2%) as non-rumor and 284 (24.8%) as a rumor, all results were ideal for each iteration of the proposed technique. The perfect performance in metrics accuracy, precision, recall, and F-measure was due to the high results that were achieved before using the ensemble classification approach.

In Ottawa shooting event the original dataset is divided as 470 (52.8%) for rumors and 420 (47.2%) for non-rumors. The superiority in results of the used measures started after the 60% rate of instances resampling.

In Sydneysiege event, the percentage of imbalance in the class label increased in the original data, as it consists of (42.8%) for rumors and (57.2%) for non-rumors, with a total of instances up to 1221. After conducted the proposed method, it was found that the increase in the results began after 50 % of the resamples from the total of the original samples.

Here, rumor detection experiments performed using PHEME datasets and the Ensemble Classifier technique for both old and new features. Majority baselines utilized in this system. The proposed approach was compared with related models reported on the same PHEME dataset that was used in our proposed method, as shown in Table VI. At first, it was compared with the best result of adding features and used three classifiers (macro mean) (Aker et al., 2017) [15]. Also, the high results of different classifiers using either or both of the content-based and social features (CRF) by Zubiaga et al. (2016) [18]. Finally, these results were compared with the offline data augmentation method based on semantic relatedness for rumor detection presented by (Han et al., 2019) [17].

TABLE VI. THE AVERAGE PERFORMANCE OF STATE OF THE ART THAT APPLIED ON **THE PHEME** DATASET AND COMPARISON WITH OUR WORK IN THE SAME MEASUREMENTS.

No.	Measurement	Our result	macro mean[17]	CRF[25]	Pheme5 [19]
1	Accuracy	78.54932	77.42	____	0.707
2	Precision	0.7892	___	0.683	0.580
3	Recall	0.7856	___	0.556	0.497
4	F-measure	0.7858	___	0.607	0.535

V. CONCLUSIONS

In this research work, common features suggested by relevant studies combined with new features, to raise overall accuracy average. The outcomes show that this approach leads to better results on PHEME datasets compared to known state- of-the-art systems. Furthermore, features ranking and instances sampling in the context dataset for optimal elicitation size and most suitability and reaching better mining effects as possible. Ensemble Voting Classifier technique was proposed for identifying both rumor and non-rumor that employ classification models based on a K-Nearest Neighbors (KNN), a Naive Bayes (NB), and a support vector machine (SVM). The effectiveness of the Ensemble approach experimentally explained that include the given chosen set, selected from the original dataset records. As shown by using extensive experiments, this ensemble schema can lead to a significant obtain in performance.

REFERENCES

[1] L. Fan, Z. Lu, W. Wu, B. Thuraisingham, H. Ma, and Y. Bi, "Least cost rumor blocking in social networks," Proc. - Int. Conf. Distrib. Comput. Syst., pp. 540–549, 2013.

[2] E. K. Al-Yasiri and A. Al-Azawei, "Improving Arabic sentiment analysis on social media: A comparative study on applying different pre-processing techniques," Compusoft, vol. 8, no. 6, pp. 3150–3157, 2019.

[3] S. Lathiya and M. Chaudhari, "Rumour Detection from Social Media : A Review," Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol., vol. 3, no. 7, pp. 384–389, 2018.

[4] O. Enayet and S. R. El-Beltagy, "NileTMRG at SemEval-2017 Task 8: Determining Rumour and Veracity Support for Rumours on Twitter.," pp. 470–474, 2018.

[5] S. Kwon, M. Cha, K. Jung, W. Chen, and Y. Wang, "Prominent features of rumor propagation in online social media," Proc. - IEEE Int. Conf. Data Mining, ICDM, pp. 1103–1108, 2013.

[6] J. Cao, J. Guo, X. Li, Z. Jin, H. Guo, and J. Li, "Automatic Rumor Detection on Microblogs : A Survey," vol. 1, no. c, pp. 1–14, 2018.

[7] A. Mahabub, "A robust technique of fake news detection using Ensemble Voting Classifier and comparison with other classifiers," SN Appl. Sci., vol. 2, no. 4, p. 525, 2020.

[8] S. Hamidian and M. Diab, "Rumor Detection and Classification for Twitter Data," SOTICS 2015 Fifth Int. Conf. Soc. Media Technol. Commun. Informatics, no. c, pp. 71–77, 2015.

[9] D. Figo, P. C. Diniz, D. R. Ferreira, and J. M. P. Cardoso, "Preprocessing techniques for context recognition from accelerometer data," Pers. Ubiquitous Comput., vol. 14, no. 7, pp. 645–662, 2010.

[10] Q. Li, Q. Zhang, L. Si, and Y. Liu, "Rumor Detection on Social Media: Datasets, Methods and Opportunities," pp. 66–75, 2019.

[11] A. M. Al-Abadi, "Mapping flood susceptibility in an arid region of southern Iraq using ensemble machine learning classifiers: a comparative study," Arab. J. Geosci., vol. 11, no. 9, 2018.

[12] I. Perikos and I. Hatzilygeroudis, "Aspect based sentiment analysis in social media with classifier ensembles," Proc. - 16th IEEE/ACIS Int. Conf. Comput. Inf. Sci. ICIS, 2017, pp. 273–278, 2017.

[13] Y. K. Zamil, S. A. Ali, and M. A. Naser, "Spam image email filtering using K-NN and SVM," Int. J. Electr. Comput. Eng., vol. 9, no. 1, p. 245, 2019.

[14] E. H. Dhah, M. A. Naser, and S. A. Ali, "Spam Email Image Classification Based on Text and Image Features," 1st Int. Sci. Conf. Comput. Appl. Sci. CAS, 2019, pp. 148–153, 2019.

[15] M. M. Hoobi, "Keystroke Dynamics Authentication based on Naïve Bayes Classifier" vol. 56, no. 2, pp. 1176–1184, 2015.

[16] A. Roy, K. Basak, A. Ekbal, and P. Bhattacharyya, "Deep Ensemble Framework for Fake News Detection and Classification," 2017.

[17] A. Aker, L. Derczynski, and K. Bontcheva, "Simple open stance classification for rumour analysis," Int. Conf. Recent Adv. Nat. Lang. Process. RANLP, vol. 2017-Septe, pp. 31–39, 2017.

[18] E. Kochkina, M. Liakata, A. Zubiaga, U. Kingdom, and U. Kingdom, "All-in-one : Multi-task Learning for Rumour Verification," 2017.

[19] S. Han, J. Gao, and F. Ciravegna, "DATA AUGMENTATION FOR RUMOR DETECTION USING CONTEXT - SENSITIVE NEURAL LANGUAGE MODEL WITH LARGE - SCALE CREDIBILITY CORPUS," pp. 1–6, 2019.

[20] J. Ma et al., "Detecting rumors from microblogs with recurrent neural networks," IJCAI Int. Jt. Conf. Artif. Intell., vol. 2016-Janua, no. May, pp. 3818–3824, 2016.

[21] F. Jin, E. Dougherty, P. Saraf, P. Mi, Y. Cao, and N. Ramakrishnan, "Epidemiological modelling of news and rumors on Twitter," Proc. 7th Work. Soc. Netw. Min. Anal. SNA-KDD 2013Jin, F., Dougherty, E., Saraf, P., Mi, P., Cao, Y., Ramakrishnan, N. (2013). 7 Epidemiol. Model. News rumors Twitter. Proc. 7th Work. , 2013.

[22] D. T. Nguyen, N. P. Nguyen, and M. T. Thai, "Sources of misinformation in Online Social Networks: Who to suspect? " Proc. - IEEE Mil. Commun. Conf. MILCOM, 2012.

[23] F. Yang, X. Yu, Y. Liu, and M. Yang, "Automatic detection of rumor on Sina Weibo," Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min., vol. 2, 2012.

[24] P. Lendvai, I. Augenstein, K. Bontcheva, and T. Declerck, "Monolingual social media datasets for detecting contradiction and entailment," Proc. 10th Int. Conf. Lang. Resour. Eval. Lr. 2016, pp. 4602–4605, 2016.

[25] A. Zubiaga, M. Liakata, and R. Procter, "Learning Reporting Dynamics during Breaking News for Rumour Detection in Social Media arXiv : 1610. 07363v1 [cs . CL] 24 Oct 2016," 2016.

[26] E. Burnaev, P. Erofeev, and A. Papanov, "Influence of Resampling on Accuracy of Imbalanced Classification," 2017.