IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-2, Feb 2019 E-ISSN: 2347-2693

Formation of Similar Users group by using Support Vector Machine with Facebook Posts

K. Mohankumar[1*], B. Srinivasan[2]

[1 2] PG and Research Department of Computer Science, Rajah Serfoji Government College, Thanjavur, Tamil Nadu, India.

[*]*Corresponding Author: tnjmohankumar@gmail.com*

DOI: https://doi.org/10.26438/ijcse/v7i2.158163 | Available online at: www.ijcseonline.org

Accepted: 21/Feb/2019, Published: 28/Feb/2019

Abstract- Users of Online Social Network (OSN) generate their own post by using texts, images, videos and resources like emojis, stickers etc., Among the different types of posts, the text content can easily be interpreted by other and exposes the full thoughts of a user towards a topic. This paper attempts to group similar users, who produced the same text posts towards a set of pre-defined topics. The similarity among users with their posts is calculated with the aid of linear Support Vector Machine (LinearSVM) classifier and the performance is evaluated.

Keywords- OSN, Similar Users, LinearSVM, Text Classification, text posts, tf-idf vectorizer

I. INTRODUCTION

Online Social Networks (OSN) are platforms, where users can share their information, make a new connection and explore the different events occurring in society [1]. The shared contents are reflecting the inner thought of users and producing a major impact around the communities in which they involved. Before OSN, users posted their thoughts on dedicated blogs and discussed the different topics on forums. A blog or forum contains a countable number of members. Users have to do a manual search process to find the intended blog or forum with a similar taste for discussions.

The advent of web 2.0 technology leads the genesis of OSN and allowing any user to generate a post regarding any topic. This kind of activity made the users of OSN to create a vast amount of unstructured data in every second, and hence the formation of implicit communities or groups. A community or group involves a set of users with similar thoughts, interests, opinions, and sentiments [2].

In the beginning stages of OSN, users posted textual data over the internet which includes news articles, and historical information and now allowed to share a vast variety of multimedia data [3]. Among the different types, posts created by using text data can easily be interpreted by others.

The using of text for posting, commenting and replying a post is fully exposing the internal traits of users towards a topic than other types of data. The text data is also referred to as categorical data and contains a huge amount of knowledge for assessing the behavior and characteristics of users. In recent years, the field of OSN become an excellent platform to express and share the opinion in the form of text towards the interesting topic. Through the sharing of posts by means of text the users of OSN also participate in the design of products with enterprises [4].

The text posts generated by the users in OSN include political, technology, products or movie reviews and daily conversing issues [5]. The posts generated by the users of OSN give a major chance to do newer dimensions of research works like detecting the feeling of users towards a topic, finding the reaction about a topic among the number of users and detecting the users of similar types.

Business companies are using the user's post to find those who are liking similar products and the intention of users towards a product which would greatly improve the business diplomacies. The posts generated by the OSN users also helps the business companies to do a new era of marketing known as digital media marketing [6].

Most research works are using the posts of OSN users to find the recommender systems, carry out sentimental analysis and to do opinion mining. All the research works lead to the detection and formation of communities or groups containing the users of similar tastes.

The textual posts produced by the users are processed by using a variety of techniques including regression, classification, and neural network training etc. The Classification method is the best for grouping text data in the field of Social Networks. Several Classification techniques, statistical measurements are used and produce results with high accuracies.

This paper tries to group Similar Users with the user's posts towards a set of pre-defined topics by using Linear Support Vector Machine (LinearSVM), a linear classifier used for classifying big volume of data. The work also used a statistical measure known as, term frequency-inverse document frequency (tf-idf) for generating equivalent vector values for the texts in user posts. The tf-idf vectorizer generates an equivalent matrix containing vectors for each corpus on the given user post. All the text data are linearly separable and the equivalent generated vectors are also exhibiting the same characteristics.

The paper is organized as follows. Section I presents the introduction of the topic. Section II contains the related works. Section III describes the methodology with the selected dataset, equations, diagrams and algorithmic steps used on this paper. Section IV discusses the results from the various observations with tables, figures and predicts the accuracy of the work. Section V is devoted to conclusions and future directions.

II. RELATED WORK

Research works have been carried out to classify the users of OSN with various methods. Identification of the same user among different social network was proposed by using the cross-links of followers from Twitter and Instagram [7]. A set of like-minded people are grouped with the help of two algorithms [8]. The first algorithm in this work retrieved interested centers form the user's text posts by using K-Means and LDA clustering techniques and the second algorithm gathered the groups with the maximum correlation between the users by using Principal Component Analysis(PCA) and the interested centers from the first algorithm. After completing the second algorithm, the collected data is fed to SVM classifier to classify new users.

Comparative study of different sentimental mining algorithms was carried out[9] for the effective text classification to be used in OSN. For better marketing, a model for doing feature-based opinion mining and sentimental analysis was proposed [10] by using fuzzy logic with features extracted from product-related textual posts. Research on classifying the top 20 most followed users of Instagram using tf-idf measurement was formulated [11]. The work also used the collected images captions as hashtags for classification.

In addition, to find the opinion and sentiments among the users, the grouping of public opinion related to a specific topic like employment problem in Indonesia is designed, using Convolutional Neural Networks(CNN) with multi-parallel CNN processes [12]. This indicates that opinion and sentiments are also used for grouping users in OSN.

There are different kinds of OSN emerging every day. The users belonging to such networks are also in different categories. Apart from the traits of users, posts created by the users of OSN are also reflecting the behavior related to their social activities. Related work on identifying socially dangerous people in a social network [13] is also carried out, using an Adaptive Neural Network (ANN) with the metadata left in pages during the registration process. The syntactic feature of text posts by the users is also considered major attention [14]. A novel approach for classifying with emotions was carried out [15]. The approach used different classification algorithms to predict the quality factor and clustered the results to extract the required feature from the training data.

Probabilistic approaches are also assuring the probability of grouping similar users. The well-known Navi Bayes classification is applied over the hashtags and hyperlinks among the users for effective grouping is carried out [16]. The work yielded better classification with known hashtags as classes.

All the related works used the sentiments, opinions, hashtags, hyperlinks, cross-links, and similarity of users by means of feelings, features related to a single topic. This paper groups similar users from the different number of samples over different topics (classes).

III. METHODOLOGY

The paper applies a linear classification of SVM over users posts on the various pre-defined topic to group users of similar traits. The SVM classifier operates under different modes like linear, non-linear and rbf. The best mode for this work is linear classification because all text data are linearly separable. The work uses tf-idf vectorizer to convert all text data into a vectorized matrix and the converted vector elements are fit with SVM classifier.

The present work also uses a dataset containing around 1 million users posts in the Times of India Facebook page. The dataset contains three columns, published_date, headline_category, and headline_text as shown in the following Table.1 with few samples

Table.1 Sample from the used dataset

published_date	headline_category	headline_text
20160520	city.vadodara	UPSC aspirants to get early coaching
20080413	other-racing.a1-gp	'A1GP race in India may take two years'
20080602	sports.football	Goa makes it to quarter-finals via penalty shoo...
20130618	city.coimbatore	Anna University defers release of engineering ...
20150922	city.noida	115 autos fined in enforcement drive in Noida

The published_date contains the date of publication of a post, headline_category contains the topic heading of the news in headline_text. The headline_text contains the actual posts of users as news. The headline_category and headline_text columns of the dataset are taken as label and feature values for classification. The work selected different samples from the dataset on each observation.

The proposed method is depicted in the following Figure.1

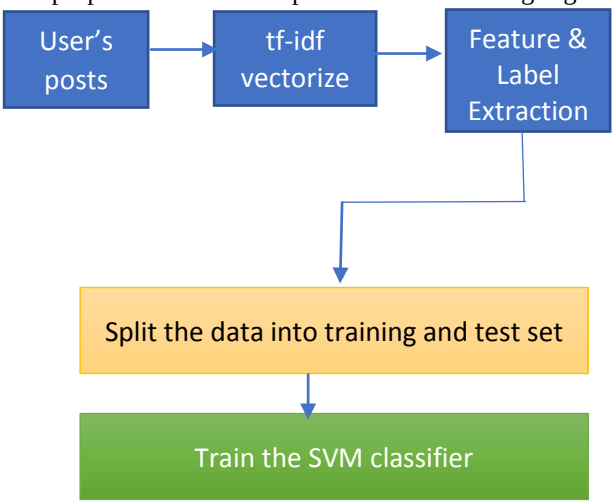

Figure.1 Methodology

The above Figure.1 shows that the methodology starts with getting users posts in headline_category and headline_text from the dataset. The dataset then is subject to tf-idf vectorize. The tf-idf vectorize stage stems and lemmatize the feature and label columns i.e. headline_text and headline_category, and finally, the process produces vector matrix. Following the production of equivalent vector elements for the taken texts, training and testing data are formed by splitting the vector elements. In the last stage, the training and test data are given to LinearSVM classifier for classification.

The tf-idf vectorize first calculates the term frequency by using the following equation(1)

$$TF(t, d) = \sum_{x \in d} f(x, t) \quad (1)$$

Where f(x, t) is a simple function defined as

$$f(x, t) = \begin{cases} 1, \text{ if } x=t \\ 0, \text{ otherwise} \end{cases} \quad (2)$$

The equation (1) calculates the term frequency by summing up the of occurrence of the word(t) in the post (d) with the aid of function f (x, t) defined in equation (2). The f (x, t) returns 1 if t occurs in post or 0 otherwise.

The inverse document frequency is calculated with the equation (3)

$$IDF(t) = \log (|D| /1+|\{d:t \in d\}|) \quad (3)$$

where $|\{d:t \in d\}|$ is the number of posts where the term t appears, when f(x,t) satisfies the condition TF(t,d)=1. The number 1 is added to avoid divide by zero error when f(x, t) returns 0. |D| is the total number of posts. The equation (3) calculates the natural logarithm for finding the inverse propagation of term t among the given number of posts D. The lesser value of IDF indicates the high occurrence of the term and a larger value indicates the low occurrence.

Finally, the tf-idf is calculated as follows

$$TF\text{-}IDF(t) = TF(t, d) \times IDF(t) \quad (4)$$

The TF-IDF value increases in proportion to the number of times a term (t) appears in a post and used as the offset frequency of the term in the generated corpus.

The algorithm FIND_SIMILAR_USERS represents all the works carried out in the present paper.

Table.2 Algorithm for text classification

Algorithm: FIND_SIMILAR_USERS

Step 1: Get samples from the dataset

Step 2: Select columns for classification.

Step 3: Preprocess the columns by removing unwanted and null values

Step 4: Apply tf idf vectorizer to
4.1 stem all words
4.2 lemmatize all words and
4.3 generate equivalent vectors

Step 5: Extract feature and class labels for Classification

Step 6: Split the feature and labels into training and test sets

Step 7: Train the LinearSVM model by using training and test data

Step 8: Predict the results

Step 9: Stop

The vectorizer in the above algorithm converts all the words in the given comments into its base format by using stemming and then converts all the verbs in the words into the base verb format. After that, the vectorizer produces equivalent vectors for each word from each post.

IV. RESULTS AND DISCUSSIONS

The performance of the LinearSVM classifier is evaluated by using the following three metrics over the observed results.

a) *Precision* gives the ratio of a relevant item over the related retrieved items.
b) *Recall* or Sensitivity gives the relevant items overall retrieved items and
c) *f1-score* is the weighted average of both precision and recall which gives how effectively a classifier classified the given samples.

Before classifying the samples, the biasing of posts towards the relevant topics was analyzed. Then, the LinearSVM is carried with the selected samples. The biasing of 50000 user posts towards 3 topics(classes). namely city. mumbai, India, and unknown are shown in the following Figure.2.

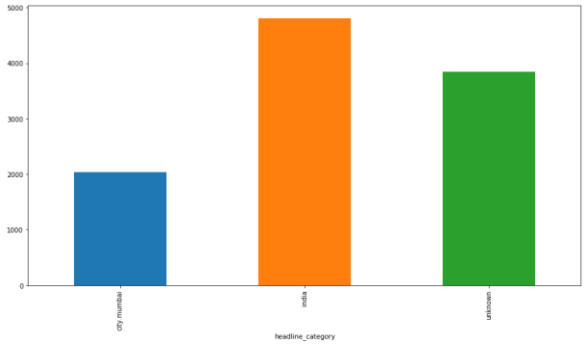

Figure.2 Biasing of 50000 samples towards 3 topics

The middle bar of Figure.2 is showing that topic India is having maximum posts and the right bar is showing the topic unknown, having the next level of maximum posts. The left bar of Figure.2 shows that the topic city.mumbai is having minimum posts. The LinearSVM is predicting the same with the following values.

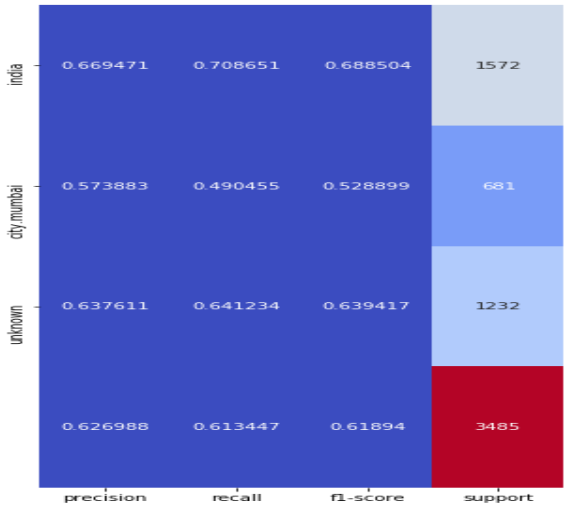

Figure.3 Precision, recall, and f1-score of 50000 samples with 3 topics

The predicted average of precision, recall, and f1-score for each topic is as follows

India = 67%
unknown= 64%
city.mumbai=57% and

From the above observations, the LinearSVM classifier predicts accurately the posts towards the topics as shown in Figure.2. The weighted average is represented in the last row of Figure.3 and predicting the accuracy of the classifier is 62%. The last column of Figure.3 is called support, which states the number of training samples of true response that contains within the related class boundary. The support values are also known as support vectors in the LinearSVM terminology and help the classifier to group the nearer point towards the topic(class) in which they belong. For example, there are 1572 support vectors, help the classifier to group the nearest test point towards India.

The confusion matrix for 50000 samples with 3 classes is depicted in Figure.4. The actual values are represented as rows and predicted values are represented as columns.

Figure.4 Confusion matrix for 50000 samples with 3 topics

The diagonal elements in the above Figure.4 give correct predictions (True-Positiveness) among three classes India, city.mumbai and unknown. The non-diagonal elements are giving false predictions among the selected samples. The values on diagonal elements also agreeing the biasing shown in Figure.2. The row-wise sum of confusion matrix in Figure.4 gives the actual support values in the last column of Figure.3.

The biasing of posts towards 10 topics as classes with 50000 posts as samples before the posts are subjected to LinearSVM classifier.

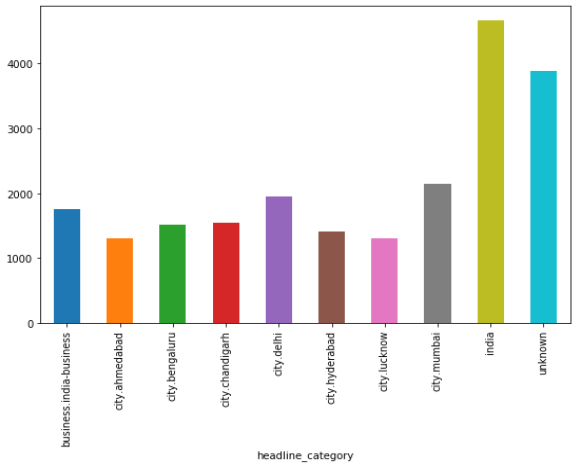

Figure.5 Biasing of 50000 samples towards 10 topics

From the above figure, it is observed that the taken 50000 samples are biased more specifically towards the specific topics than the classification shown in Figure.2. The above Figure.5 also explains that the samples begin to spread towards the correct topics when the topics are also grown in size. The corresponding predicted precision, recall, and f1-score along with support values for 50000 samples with 10 topics is shown in Figure.6

Figure.6 Precision, recall, f1-score and support values for 50000 samples with 10 topics.

The accuracy of the classification is observed as 42% and the value shows that the LinearSVM classifier classified the samples near accurately towards the relevant topics by agreeing the biasing shown in Figure.5

From the above two cases shown in Figure.3 and Figure.6, the LinearSVM harmoniously classified the sample posts towards the correct topics. The proportional classification nature of LinearSVM is verified with a different set of samples along with the accuracy values.

Table.3 Proportional classification of LinearSVM

S. No	No. of Sample posts	No. of topics(classes)	Accuracy (%)
1	500	2	46.4
2	500	3	38.5
3	5000	4	48.2
4	5000	5	44.6
5	5000	6	40.0
6	50000	7	48.5
7	50000	8	47.2
8	100000	9	47.0
9	100000	10	47.0
10	100000	20	39.2

The accuracy value on each row of Table.3 shows that the accuracy is inversely proportional to the number of classes considered for classification. When there is a larger number of classes in a classification, then the accuracy is having a lower value than the previous classification. The inverse proportion is not degrading the result but shows the accuracy of classification towards the correct topic.

The error rate is the next important factor, reflecting the efficiency of classification. The error rate of LinearSVM with 50000 samples over 10 topics is shown as a Learning Curve in Figure.7. The Learning Curve clearly explains the error rate of test data is gradually reducing along with the increasing amount of training data.

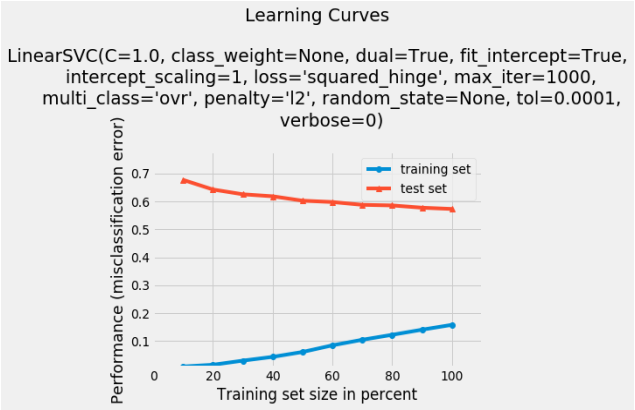

Figure.7 Learning curve showing the test accuracy.

V. CONCLUSION AND FUTURE SCOPE

The paper classified the users of similar type by using posts belonging to different topics. The LinearSVM classifier tunefully classified the samples towards the correct topics. The processing time of the classifier is comparably optimum than the other techniques. The main problem in the text classification is the number of topics(classes) used in the classification. The topics are unbalanced one when there are fewer in amount with a larger amount of posts. There is a chance for a post biased towards a wrong topic (unbalanced class), due to the amount of fewer topics. When the topics are grown in size, then there is a great chance for a post to bias towards the correct topic. The problem of unbalanced classes is to be taken into account during the text classification.

In practical, users of OSN can post anything on their walls, in a group or community without any pre-determined topic. So, a grouping of similar users without the known topics or with balanced classes of the known category is appreciated in the future.

REFERENCES

[1] Reshma. M and Raji.R.Pillai, *"Semantic Based Trust Recommendation system for Social Networks using Virtual Groups",* published in International Conference on Next Generation Intelligent Systems (ICNGIS),India, September 2016.

[2] Mariam Adedoyin-Olowe, Mohamed Medhat Gaber et al, *"A Survey of Data Mining Techniques for Social Network Analysis",* Journal of Data Mining and Digital Humanities, December 2013.

[3] Georgios Lappas, *"From Web Mining to Social Multimedia Mining",* published in International Conference on Advances in Social Networks Analysis and Mining, Taiwan, July 2011.

[4] Guoxin Li, Xue Yang et al,*" Customer-Generated Content in Company Social Media Platform: How Social Network Works?",* published in IEEE International Conference on Management of Innovation and Technology (ICMIT), Thailand, September 2016.

[5] Shankar Setty, Rajendra Jadit et al, *"Classification of Facebook News Feeds and Sentiment Analysis",* IEEE, September 2014.

[6] Harsh Namdev Bhor, Tushar Koul et al,*" Digital Media Marketing using Trend Analysis On Social Media ",* ICSCI, January 2018.

[7] Waseem Ahmad and Rashid Ali, "*A Framework for Seed User Identification across Multiple Online Social Networks",* IEEE, September 2017.

[8] Soufiene Jaffali, Salma Jamouss et al, *"Clustering and Classification of Like-Minded People from Their Tweets",* IEEE International Conference on Data Mining, 2014.

[9] Mohammed H. Abd El-Jawad et al, *"Sentiment Analysis of Social Media Networks Using Machine Learning",* published in 14th International Computer Engineering Conference (ICENCO), IEEE, December 2018.

[10] B. Vamshi Krishna, Ajeet Kumar Pandey et al, *"Feature Based Opinion Mining and Sentiment Analysis Using Fuzzy Logic",* Springer Link, Cognitive Science, and Artificial Intelligence, pp 79 – 89, December 2017.

[11] Bernardus Ari Kuncoro, Bambang Heru Iswanto et al, *"TF-IDF Method in Ranking Keywords of Instagram Users' Image Captions",* ICITSI,2015.

[12] Devi Munandar, Andria Arisal et al," Text *Classification for Sentiment Prediction of Social Media Dataset using Multichannel Convolution Neural Network",* published in International Conference on Computer, Control, Informatics and its Applications (IC3INA), November 2018.

[13] Irina Stefanova and Andrey Kiryantsev, *"Analysis of User Groups in Social networks to Detect Socially Dangerous People",* International Scientific Practical Conference Problems of Info Communications, Science and Technology (PIC S &T), October 2018.

[14] Macro Di Givvanni, Macro Brambilla et al, *"Content-based Classification of Political Inclinations of Twitter Users"* IEEE International Conference on Big Data, December 2018.

[15] S.Geetha, Visnukumar et al, *"Tweet Analysis Based On Distinct Opinion of Social Media Users'",* IEEE, December 2018.

[16] Vibhuti Gupta and Rattikorn Hewett, *"Unleashing the Power of Hashtags in Tweet Analytics with Distributed Framework on Apache Strom",* IEEE International Conference on Big Data Conference, December 2018.

AUTHORS PROFILE

Dr. K. Mohan Kumar received his Ph.D. in Computer Science from Bharathidasan University. Presently, he works as an Assistant Professor in the PG and Research Department of Computer Science at Rajah Serfoji Government College, Thanjavur, Tamil Nadu, India. He published more than 50 research papers in reputed journals. He has 23 years of teaching experience and 18 years of research experience. His main research areas are machine learning, IOT, network security, and big data analytics.

B. Srinivasan is a part-time research scholar, doing his Ph.D. in the PG and Research Department of Computer Science at Rajah Serfoji Government College, Thanjavur. He received his Master and M.Phil. degrees from Bharathidasan University and published more than 30 research articles in reputed journals. Presently, he works as an Assistant Professor in the PG and Research Department of Computer Science, Khadir Mohideen College, Adirampattinam, Tamil Nadu, India. He has 20 years of teaching experience and 13 years of research experience. His main research areas are social media mining, social network analytics, machine learning, data mining, and sentimental mining.

