Identifying Trolls and Determining Terror Awareness Level in Social Networks Using a Scalable Framework

Busra Mutlu
Department of Computer Engineering
KTO Karatay University
Konya, Turkey 42020
Email: busra.mutlu@karatay.edu.tr

Merve Mutlu
Department of Computer Engineering
KTO Karatay University
Konya, Turkey 42020
Email: mrv.muttlu@gmail.com

Kasım Oztoprak
Department of Computer Engineering
KTO Karatay University
Konya, Turkey 42020
Email: kasim.oztoprak@karatay.edu.tr

Erdogan Dogdu
Department of Computer Engineering
TOBB University of Economics and Technology
Ankara, Turkey 06560
Email: edogdu@etu.edu.tr

***Abstract*—Trolls in social media are 'malicious' users trying to propagate an opinion or distort the general perceptions. Identifying trolls in social media is a task of interest for many big data applications since data cannot be analyzed effectively without eliminating such users from the crowd. In this paper, we present a solution for troll detection and also the results of measuring terror awareness among social media users. We used Twitter platform only, and applied several machine learning techniques and big data methodologies. For machine learning we used k-Nearest Neighbour (kNN), Naive Bayes, and C4.5 decision tree algorithms. Hadoop/Mahout and Hadoop/Hive platforms were used for big data processing. Our tests show that C4.5 has a better performance on troll detection.**

***Keywords*—Troll detection, kNN, Naive Bayes, C4.5, terrorism awareness**

I. Introduction

Social media platforms allow people to share and discuss news, ideas, information and other user-generated content in the form of blogs, microblogs, forums, video, and audio. This user-generated content on social media is a source of big data for understanding the society or communities by way of analyzing with appropriate methods and tools. Twitter is one of the biggest social media and social network platforms. More than half a billion tweets (microblog posts) are posted everyday in average by millions of users on Twitter. Therefore, it is a great source for analyzing big social data.

Social media and network platforms are free to all, therefore we also find many malicious users, also called "trolls" on these platforms. Trolls try to create discord and distort the information flow on the network by false information or inflammatory messages [1]. Therefore, their presence and posts prevent the true understanding of the information on the network, and big data analytics on social media requires filtering out trolls in the first place. In this paper we present methods to detect trolls. Trolls can not be exactly determined since they have complex and unstable behaviors. By using several classification algorithms, we have tried to find out the best performing algorithms in order to classify users with troll-like behaviors.

Lately, terror events are almost everywhere in the world and therefore the topic of terror is discussed widely on social networks. A second aim of this work is to test and find the terror awareness levels of social network users.

In this study we tried to evaluate and measure the awareness of users against terrorist actions by gathering data, categorizing tweets by their contents. We also extended the study to investigate the opinions of the users tweeting about the topic. In contrast to our studies, other studies [1] [2] [5] about the subject focus on classifying users by implementing different classification algorithms or categorizing the context and features of tweets.

Finally, we measured the awareness level of users in Turkey and around the world in regard to terrorism. Turkey is one of the countries that have made an effort to raise public awareness about terrorism. We collected tweets using a keyword list related to terrorism and then analyzed the awareness level using Weka tool.

We employed three machine learning algorithms to detect trolls, these are C4.5 decision tree[1], Naive Bayes[2], and k-Nearest Neighbour (kNN)[3] methods. The results of our evaluations indicate that the proposed model detects the trolls or troll-like behaving users with 89% accuracy using C4.5 algorithms. In comparison, the success ratio with Naive Bayes method and kNN are 79% and 83% respectively.

The rest of the paper is organized as follows. In section 2, we give a brief summary of related literature in opinion mining for Twitter platform. Background definitions and a formal description of the proposed solution for the detection of trolls or troll-like users are given in section 3 and the determination of terror awareness level is presented in section 4. Experimental results and the comparison of the different machine learning methods are presented in section 5. We then conclude in section 6.

[1]https://en.wikipedia.org/wiki/C4.5_algorithm
[2]https://en.wikipedia.org/wiki/Naive_Bayes_classifier
[3]https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

978-1-4673-9005-7/16/$31.00 ©2016 IEEE

II. Related Works

It is very different to detect opinion manipulators (trolls) and troll-like behaving users. Todor et al. [1] proposed a method to solve this issue with different variations of a troll definition. A classifier has been trained to distinguish trolls from non-trolls. There were also several studies targeting detection of trolls accurately. To classify users, Kumar et al. [2] developed a way of troll detection which was faster than many past algorithms proposed in the literature.

Tweet content analysis is widely studied. Paul et al. categorized tweets based on whether they include a question or not using crowdsourcing methods [3]. Andre et al. [4] performed an analysis of microblog content from the readers' point of view, powered by a novel design for collecting large numbers of voluntary ratings. Claudia et al. [5] classified Twitter users in much broader categories and constructed a comprehensive list of features. They also demonstrated a study on extracting personality related attributes for Twitter users based on features extracted from their content in order to infer professions of users automatically.

Jain et al. [6] worked on categorizing Twitter users based on their interests. They conducted experiments Hadoop/Mahout platform, and studied the performance evaluation of K-means and Fuzzy k-means algorithms, and presented the comparative results of the algorithms. To detect and stop a cyberbullying situation, Garcia et al. [7] presented a methodology by collecting information about user profiles and their tweets, and then analyzing different features in the profiles of users. They developed a method using machine learning algorithms.

Many different machine learning algorithms are used for classifying and analyzing tweets in the literature. Naive Bayes, k-Nearest Neighbour (kNN) and C4.5 are commonly used algorithms as is the case in [8], [9] and [10]. However, the performances of the algorithms are different from each other depending on the dataset and the problem at hand. Manikandan et al. [8] found that Naive Bayes technique performs better than kNN and C4.5 Decision tree methods. Vijayarani et al. [9] analysed the performance of Bayesian and Lazy classifiers. From their experimental results, they observed that the Lazy classifier is more efficient than Bayesian classifier. Hence, in our study, to able to get the best result we compared commonly used supervised learning algorithms. In contrast to other studies, our goal is to analyze tweets and detect "troll" users first and then analyze the terrorism awareness among social media users.

III. Troll Detection

Troll, or malicious user, detection in social media is an important problem for social media analysis since these users distort the true message in the media and the elimination of these users and their content from the dataset before analyis is vital. Here we consider three different machine learning algorithms to classify Twitter users as troll or non-troll. We also present the dataset we collect for the evaluation.

A. Dataset

In order to collect test data, Twitter REST API is used. There is a usage limit of 180 queries per 15 minutes in the API. Using the upper limits on the number of queries per minute and the maximum number of tweets per query, we were able to collect 18.000 tweets per 15 minutes.

We collected 95.578 tweets belonging to 3.321 users on the topic of terrorism by using the Twitter REST API. We also collected meta data about these users including follower/following counts, tweet counts, profile picture, and retweet counts. The size of the collected data is approximately 50 MB, and had 95.578 tweets in English and Turkish belonging to 3.321 unique users. The most frequently used terms were chosen as keywords to collect tweets. Some of the keywords are "teror, terror, terrorism, *canlı bomba*('suicide bomber'), PKK, DAES". We used the WEKA[4] tool to apply kNN, Naive Bayes, and C4.5 decision tree algorithms on the dataset.

Table I shows the summary of collected datasets, part of which is divided into a training set that has the properties listed in Table II. 2.605 users out of 3.321 users were considered as training data and remaining 716 users out of 3.321 users were considered as test data.

TABLE I. Dataset collected

#tweets	95.578
#users	3.321

TABLE II. Training Dataset

#users	2.605
#troll users	1.402
#non-troll users	1.203

Many studies related to this topic had different approaches. Cambria et al. [11] used semantic and sentiment analysis to filter out trolls. Herring et al. and Buckels et al. studied the general trolling behavior related to users' personalities [12] [13]. Todor et al. [1] extracted a number of comments posted over a number of days on a Bulgarian media community forum on the Internet, number of days with at least one comment, and number of publications commented on for each user. The results of the studies indicated that trolls were unsteady entities. There were many kinds of them, such as "The Calling You Out" troll, the freeloader troll, the link jumping troll, the public shaming troll, the brand jacker troll, and the IRLtroll[5]. It was difficult to identify trolls precisely.

In the proposed model, the awareness of Twitter users regarding terrorism and the detection of trolls were investigated through the processing Twitter data. The data collection was performed by a mechanism in order to enable us to collect Twitter data for several days belonging to a group of users. The mechanism also collected data giving ideas on several features of the users to determine if they are trolls or not. These features are: the average number of tweets a user sends per day, the number of 'followers' a user, has the number of users a user was 'following' and the ratio of retweets in tweets sent by a user. These were the key data for troll detection. For the purpose of training the system, 2.605 twitter users were manually examined one by one. As a result of the

[4]http://www.cs.waikato.ac.nz/ml/weka

[5]http://www.twistimage.com/blog/archives/the-6-types-of-twitter-trolls/

examinations, the following rules were extracted in order to decide if a user is troll:

- Users sending more than 50 tweets a day,
- Users having follower/following rate 0.4 and below,
- Users exceeding 70% retweet rate (70 retweets per 100 tweets sent)
- Default egghead profile images refer to 0 and other profile images refer to 1 (this gives some hint on the users but not a core differentiator)

Hence, we extracted general features and tried to detect obvious trolls such as those listed in Table III. Our training dataset included data from 2.605 users formed from both troll and non-troll users. This dataset was training data and metric for the classifier.

TABLE III. SAMPLE FEATURE VALUES FOR TROLL USERS

Screen Name	Egghead	Followers/ Following	Tweet per Day	Percent of Retweet
user1	1	0.4	92.0	99.5
user2	1	0.2	106.0	100.0
user3	1	0.4	70.0	86.9
user4	1	0.3	78.0	98.0

During the detection process of trolls several machine-learning algorithms are employed. The details of how they engaged with the proposed model are explained in the following subsections.

B. Classification of trolls using k-Nearest Neighbour (kNN) algorithm

As the first algorithm, IBk was used in order to classify users as trolls or non-trolls. It is a k-Nearest Neighbour (kNN) classifier that focuses on distance learning. In this algorithm, K expression refers to a positive integer, is used to find k-nearest neighbour. The number of nearest neighbours can be specified explicitly in the object editor or determined automatically using leave-one-out cross-validation focus to an upper limit given by the specified value [9]. IBk algorithm includes Chebyshev, Manhattan, Minkowski and Euclidean function regarding distance metric. However, commonly used distance calculation for each variables is Euclidean distance. From more than one neighbour(classified users) can be weighted according to the test instance(new user). The number of training instances kept by the classifier can be restricted by setting the window size option. As new training instances are added, the oldest ones are detached to maintain the number of training instances at this size [9]. According to training instances, the test instance is added to the nearest neighbour class by using weight.
We used IBk as classification algorithm in order to determine if a user was troll or non-troll.

C. Classification using Naive Bayes algorithm

The Naive Bayes is a simple probabilistic classifier. As all classifying algorithms, which aims to classify new data accurately by using vectors of feature values of the training data. The Naive Bayes algorithm that uses the Bayes theorem, calculates a set of probabilities by counting the frequency and combinations of values in a given data set and assumes all attributes to be independent given the value of the class variable [14]. A vector is represented by a vector *(1)*

$$x = (x_1, x_2, \ldots, x_n) \qquad (1)$$

in which some n features (independent variables) in a class are independent of values of other predictors. Then, by using n features on a given class(c), all conditional probabilities are producted as seen in *(2)*.

$$P(c|x) = P(x_1|c)P(x_2|c)\ldots P(x_n|c)P(c) \qquad (2)$$

As can be seen in the equation *(2)* given above, Naive Bayes was briefly equal to the product of all conditional probabilities.

Next, expected values *(3) (4)* were calculated to distinguish trolls from non-trolls in our test dataset.

In other words, each class had its own conditional probability value. To understand which class that data belongs to, conditional probability values were multiplied [16]. In the equations *(3)* and *(4)* shown below;

$$E(Troll) = \frac{P(Tr)p(TpD|Tr)p(FF|Tr)p(PoR|Tr)}{normalization} \qquad (3)$$

$$E(NT) = \frac{P(NT)p(TpD|NT)p(FF|NT)p(PoR|NT)}{normalization} \qquad (4)$$

E is ExpectedValue, TpD is Tweets per Day, FF is Rate of Followers and Following and PoR is used for the value of Percentage of Retweets, NT is Non-Troll and Tr is Troll.

In order to figure out whether a user was a troll, first, training dataset was used to calculate the probability of trolls [P(Troll)]. Next a conditional probability was calculated for each feature of a troll (Tweets per Day, Followers-Following ratio, Percentage of Retweets) and was divided by the normalization value *(5)*.

Normalization Value= P(Troll)p(TpD| $Troll)p(FF \mid Troll)p(PoR \mid Troll) + P(NT)p(TpD \mid NT)p(FF \mid NT)p(PoR \mid NT)$ *(5)*

Likewise, probabilities were calculated for trolls and non-trolls respectively *(6)*.

$$P(Troll) = \frac{Number\ of\ Troll}{Number\ of\ Users} \qquad (6)$$

Before calculating the conditional probability for each feature, variance *(7)* was calculated as below

$$Var(X) = E[(X - \mu)^2] \qquad (7)$$

where E is the Expected Value, Var is the Variance and μ stands for the mean.

After this stage, conditional probabilities were calculated. There were many different distribution techniques used to find conditional probability, but Gaussian Distribution was used because of the accuracy obtained

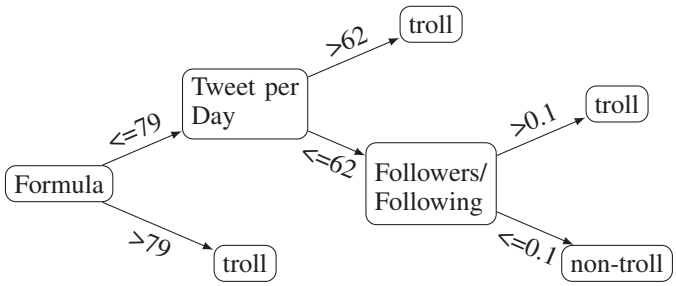
$$p(TpD|Troll) = \frac{1}{\sqrt{2\pi\sigma^2}} exp(\frac{-(x-\pi)^2}{2\sigma^2}) \qquad (8)$$

where TpD is Tweet per Day, exp is exponential and σ^2 is Variance, μ is mean value.

As in the equation *(8)* shown above, the conditional probability was calculated for each feature. Then the product of all probabilities was calculated. After expected values were found for trolls and non-trolls, these values were compared. Regarding Naive Bayes, the big value was a determinant for the users' class.

D. Classification using C4.5 algorithm

A decision tree generated by C4.5 algorithm is the implementation of the ID3 algorithm which calculates the entropy of each attribute, builds a decision tree with respect to entropy, and creates a binary tree by recursively evaluating the subsets by using the remaining attributes. The decision tree approach is the most useful algorithm in the classification problem. With this technique, a tree was constructed to model the classification process [14]. A decision tree was built by using WEKA (suite of machine learning tools). After the algorithm was trained by using the training data the following decision tree was obtained as output.



The steps of the algorithm used in the study can be summarized in the following entropy equation *(9)*:

$$I(M) = -\sum_{i=1}^{k}((freq\frac{(S_i.M)}{|M|}).log_2(freq\frac{(S_i.M)}{|M|})) \qquad (9)$$

where I is the Entropy, M is the example, S is the class and |M| is the number of example.

Regarding to values of a class, frequency was considered for any example. Then information value was calculated for each example.

- First, the algorithm gathers features of every attribute.
- After that, information was divided into partitions and each partition was processed recursively by using the following equation *(10)*

$$I_x(P) = \sum_{i=1}^{n}((\frac{|P_i|}{|P|}).I(P_i)) \qquad (10)$$

where P indicates the Partition.

- The following equation *(11)* calculates the information gain of each attribute

$$IG(X) = I(P) - I_x(P) \qquad (11)$$

where IG represents Information Gain, I represents Entropy and P stands for Partition.

- Each attribute is split them into subsets according to information gain and the maximum IG is at the top of the tree node.
- Recurred on subsets using remaining attributes.
- Finally, information gain for any attribute was equal to a change in entropy (entropy is typically changed when we use a node in a decision tree to partition the training instances into smaller subsets) [15].

IV. Terror Awareness Detection

A. Methods

1) Classifying Users Using the Mahout: Generally, classification algorithms can be used to automatically classify documents and images in many domains. The Apache Mahout[6] platform is a machine-learning library that is run on Hadoop in distributed manner [6]. We use Mahout in order to classify trolls with the Multinomial Naive-Bayes classifier. This Bayesian-algorithm works for text data by using a training data set, that is a set of tweets each of which is associated with a subject category ("Terror and War" and "Other" categories in this case). Training dataset is manually labeled for categorizing tweets by their subject contents as "Other" and "Terrorism and War". This set includes 400 users out of 3.321 users and their 400 tweets (one tweet per user). Training dataset was converted to the hadoop sequence file format. After uploading this file to HDFS, Mahout was run to transform the training set into vectors using TF-IDF weights with Multinomial Naive-Bayes classifier. The TF(Term Frequency) is in general defined as the number of times a given term t appears in a document d.

In practice, the term frequency is often normalized by dividing the raw term frequency by the document length [18].

$$TF(t) = \frac{(Number\ of\ times\ term\ t\ appears\ in\ a\ document)}{(Total\ number\ of\ terms\ in\ the\ document)}$$

The Inverse Document Frequency(IDF) is number of information the word provides, that is, whether the term is common or rare across all documents.

$$IDF(t) = log_e(\frac{Total\ number\ of\ documents}{Number\ of\ documents\ with\ term\ t\ in\ it})$$

TF-IDF approach also assumes that the importance of a word is inversely proportional to how often it occurs across all documents.

$$TF - IDF(t) = TF(t) * IDF(t)$$

After the TF-IDF weight calculation, the training set was used to train the classifier. The classifier was worked properly on the testing set. Finally, in line with our purpose, we classified users belonging to each category "Other" and "Terrorism and War".

[6]Apache Mahout, http://mahout.apache.org

2) Data Organization with HIVE: After using Mahout, we applied HIVE to the problem of unstructured data. HIVE[7] that has developed by Facebook, is data warehouse system, which enables reading, writing, and managing large datasets residing in distributed storage using SQL for Hadoop. Hive allows to design on unstructured data and supports queries expressed in a SQL-like declarative language called HiveQL(the Hive query language) [17]. It allows to create, alter, drop databases, tables, views, functions, indexes and to write queries with statements that are similar to T-SQL.

First, data obtained by using Twitter REST API, were loaded to Hadoop File System. In order to gain the ability to create arranged tables with different columns, Hive platform were used. By using HiveQL queries -like SELECT, INSERT, LOAD etc.- we tried to convert unstructured data to structured data. After saving on the table by using LOAD and INSERT commands, the Twitter users with location information(country or city) were filtered by using SELECT command and aggregation function statements. In some users' profile, there were no location information, because it is optional in Twitter. Therefore, we filtered as 300 users out of 3.321 users provided the country location information around the world and 200 users out of 3.321 users provided city location information in Turkey. The relative frequency of locations was calculated both for Turkey and other countries. The analyzing of filtered users indicated terror awareness level efficiently.

The aim of this study was to determine who were the manipulators of social media and how they generated false agendas. We tried to find out a mechanism to separate the trolls from the normal users in order to reach a consensus of the target group. Moreover, determining the location of the users both in Turkey and in the world with the highest awareness level was secondary purpose of our study.

V. EXPERIMENTS

We used WEKA tool for machine learning. We have specifically used J48 implementation for C4.5, IBk implementation for kNN, and NaiveBayes implemantion for Naive Bayes algorithms in WEKA tool.

A. Results for classification using three supervised machine learning methods

First, we applied kNN algorithm to 716 users out of 3.321 users for the test dataset by using training dataset in WEKA. We tested by selecting 'Supplied test set' option and uploading arff file. The confusion matrix is generated for T-NT (Troll-Non-Troll) classes having two possible outcome values "troll" or "non-troll".

TABLE IV. CONFUSION MATRIX FOR KNN ALGORITHM

	Non-troll	Troll
Non-Troll	303	70
Troll	50	293

Correctly Classified Instances 83%

The results of the experiments on kNN algorithm was summarized in Table IV. As depicted in Table IV, the number of true positives (TP) for class Non-troll is 303, while false positives (FP) is 70. And, for the class 'Troll', the number of true negatives (TN) is 293 and false positives (FP) is 50 respectively. The Diagonal elements of the matrix, TP+TN = 303+293 = 596 represent the correctly classified instances and other elements; 70+50 = 120 represent the incorrectly classified instances. Thus, the ratio of correctly classified instances, and therefore accuracy, is 83% (596 out of 716 users).

We applied kNN classification to the remaining 716 users found for Troll and Non-Troll classes on the test dataset. We found that 293 out of 716 users are classified as 'troll'.

Secondly, C4.5 algorithm was applied on the test dataset by using training dataset. We tested by selecting 'Supplied test set' option and uploading arff file in WEKA. The confusion matrix was generated for T-NT (Troll- Non-Troll) class having two possible values "troll" or "non-troll".

TABLE V. CONFUSION MATRIX FOR C4.5

	Non-Troll	Troll
Non-Troll	297	76
Troll	0	343

Correctly Classified Instances 89%

The results of the experiments on C4.5 were summarized in Table V. As depicted from Table V, the number of true positives (TP) for class 'Non-Troll' was 297, while false positives (FP) was 76. And for class 'Troll', the number of true negatives (TN) was 343 and false positives (FP) was 0 respectively. Diagonal elements of matrix, TP+TN = 297+ 343 =640 represent the correct instances classified and other elements; 76+0 =76 represents the incorrect instances. Thus, the ratio of correctly classified instance was 89% (640 out of 716 users).

According to decision tree above, a result 343 users out of 716 users were considered as 'troll'.

Thirdly, Naive Bayes algorithm was applied on the test dataset by using training dataset as the other algorithms used. We tested by selecting 'Supplied test set' option and uploading arff file in WEKA. The confusion matrix was generated for T-NT (Troll- Non-Troll) class having two possible values troll or non-troll.

TABLE VI. CONFUSION MATRIX FOR NAIVE BAYES

	Non-Troll	Troll
Non-Troll	280	93
Troll	53	290

Correctly Classified Instances 79%

The results of the experiments on Naive Bayes were summarized in Table VI. As depicted from Table VI, the number of true positives (TP) for class 'Non-Troll' was 280, while false positives (FP) was 93. And for class 'Troll', the number of true negatives (TN) was 290 and false positives (FP) was 53 respectively. Diagonal elements of matrix, TP+TN = 280+290=570 represent the correct instances classified and other elements; 93+53=146 represents the incorrect instances.

[7]Apache Hive TM, www.hive.apache.org

Thus, the ratio of correctly classified instance was 79% (570 out of 716 users).

As a result, 290 users out of 716 users were considered as 'troll'.

According to the results above, C4.5 has more accurate and desired conclusion with the success ratio of 89%. The comparison of three algorithms was summarized in Table VII below.

As depicted in the Table VII, 2.605 users out of 3.321 users were considered as training data and 716 users out of 3.321 users were considered as test data. Test data was identical for both C4.5, Naive Bayes and kNN, shown on Table VII. The accuracy percentages of three algorithms were calculated by WEKA tool. C4.5 had %89 accuracy and also the algorithm gave the most consistent results on classification of test data.

TABLE VII. CLASSIFICATION ALGORITHMS' ACCURACY

	kNN	Naive Bayes	C4.5
Training Data			
users	2.605	2.605	2.605
trolls	1.402	1.402	1.402
non-trolls	1.203	1.203	1.203
Test Data(Correctly Classified)			
users	716	716	716
trolls	293	290	343
non-trolls	303	280	297
Accuracy	83%	79%	89%

B. Results for Terror Awareness Detection

Our research showed that, the result for classifying users using Mahout: 354 users out of 400 users scored high in the 'Terrorism and War' category. This clearly indicates that, people's awareness of terrorism was 88.5% worldwide. The result for data organization with HIVE: 300 users out of 3.321 users provided the country location information around the world. And 200 users out of 3.321 users provided city location information in Turkey. After the calculation of relative frequency of locations both for Turkey and other countries, first five results were as depicted from Table VIII and IX. Our goal was to get value of awareness as well as interpreting it by using the other twitter data.

TABLE VIII. AWARENESS FREQUENCY OF THE FIRST FIVE CITY IN TURKEY

City	Relative Frequency
Istanbul	57.25
Ankara	7.5
Eskisehir	5.0
Izmir	5.0
Bursa	4.0

TABLE IX. AWARENESS FREQUENCY OF THE FIRST FIVE COUNTRY

Country	Relative Frequency
United States	19.6
Brasil	15.36
United Kingdom	6.3
Argentina	6.1
India	5.4

VI. CONCLUSION AND FUTURE WORK

In this paper, the detection and elimination of the effects of trolls generates complicated structure which needed extreme computation power and sophisticated architectural design in order to obtain expected results. Luckily, we achieved to detect 'obvious trolls'. During the experiments, we categorized users as troll or not about terrorism by using three supervised algorithms namely kNN, C4.5 Decision Tree and Naive Bayes. And we analyzed the algorithms by using the classification accuracy. From the results, it is observed that the C4.5 algorithm performs better than the other algorithms.

Next, we classified these trolls by the contents of their tweets since some of them aim to manipulate social media by generating false agenda. The classification process on the tweets was performed over Mahout using Naive Bayes algorithm. From the results, it is observed that 354 users out of 400 users scored high in the 'Terrorism and War' category. Finally, we revealed the awareness level of users in Turkey and world.

Social media is a platform, allowing people to share, discuss and modify user-generated content. For this reason, there may be many users called 'trolls' that creates trouble about a delicate issue like terrorism. In this study, we detected these kind of users and demonstrated how these users aware of terrorism in Turkey and around the world by using HIVE. From the results, it is observed that the United States has higher terror awareness level than the other countries on the world and Istanbul has higher terror awareness level than the other cities in Turkey.

In the future, the study can be extended by using sentiment or opinion analysis in order to determine every kind of trolls more precisely. And different classifying algorithms can be used to compare their performance and accuracy.

REFERENCES

[1] T. Mihaylov, G. D. Georgiev, P. Nakov, "Finding Opinion Manipulation Trolls in News Community Forums," in Proceedings of the Nineteenth Conference on Computational Natural Language Learning, CoNLL, July, 2015, Vol. 15, pp. 310-314.

[2] S. Kumar and F. Spezzano, "Accurately detecting trolls in Slashdot Zoo via decluttering," in Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on, August, 2014, pp.188-195.

[3] S.A. Paul, L. Hong,E.H. Chi, "What is a Question? Crowdsourcing Tweet Categorization," in Workshop on Crowdsourcing and Human Computation at the Conference on Human Factors in Computing Systems (CHI), 2011.

[4] P. Andre, M.S. Bernstein,K. Luther, "Who Gives A Tweet? Evaluating Microblog Content Value," in Proceedings of the ACM 2012 conference on Computer Supported Cooperative Work. ACM, February, 2012, pp.471-474.

[5] C. Wanger, S. Asur,J. Hailpern, "Religious Politicians and Creative Photographers: Automatic User Categorization in Twitter," in Social Computing (SocialCom), 2013 International Conference on, September, 2013, pp. 303-310.

[6] E. Jain and S.K. Jain, "Categorizing Twitter Users on the basis of their interests using Hadoop/Mahout Platform," in 9th International Conference on Industrial and Information Systems (ICIIS), December, 2014, pp.1-5.

[7] Galan-Garcia, Patxi, Jose Gaviria de la Puerta, Carlos Laorden Gomez, Igor Santos, and Pablo Garcia Bringas, "Supervised machine learning for the detection of troll profiles in twitter social network: Application to a real case of cyberbullying," in International Joint Conference SOCO'13-CISIS'13-ICEUTE'13. Springer International Publishing, 2014, pp. 419-428.

[8] P. Manikandan and D. Ramyachitra, "Naive Bayes Classification Technique for Analysis of Ecoli Imbalance Dataset," in International Journal of Computational Intelligence and Informatics, July âĂŞ Septembe, 2014, Vol. 4.

[9] Ms S. Vijayarani and Ms M. Muthulakshmi, "Comparative Analysis of Bayes and Lazy Classification Algorithms," in International Journal of Advanced Research in Computer and Communication Engineering, August,2013, Vol. 2.8: 3118-3124.

[10] Elijah Olusayo Omidiora,Ibrahim Adepoju Adeyanju and Olusayo Deborah Fenwa, "Comparison of Machine Learning Classifiers for Recognition of Online and Offline Handwritten Digits," in Computer Engineering and Intelligent Systems, 2013, Vol.4, No.13.

[11] E. Cambria, P. Chandra,A. Sharma and A. Hussain, "Do not feel the trolls," in Proceedings of the 3rd International Workshop on Social Data on the Web, ISWC, Shanghai, 2010.

[12] S. Herring, K. Job-Sluder,R. Scheckler and S. Barab, "Searching for safety on- line: Managing "trolling" in a feminist forum," in The Information Society, 2002, 18(5):371-384.

[13] E.E. Buckels, P.D. Trapnell and D.L. Paulhus,"Trolls just want to have fun," in Personality and individual Differences, 2014, 67:97-102.

[14] T.R. Patil, Mrs.S.S. Sherekar and S. Gadgebaba, "Performance Analysis of Naive Bayes and J48 Classification Algorithm for Data Classification," in International Journal Of Computer Science And Applications, 2013, Vol. 6, No.2.

[15] "Bilgisayar Kavramları", URL:http://bilgisayarkavramlari.sadievrenseker.com/2012/11/13/c4-5-agaci-c4-5-tree/8

[16] S.E. Seker about Is Zekasi ve Veri Madenciligi. Istanbul, July, 2013, ISBN: 605-12-76-71-7, pp.145-150.

[17] A. Thusoo, J.S. Sarma, N. Jain, Z. Shoa, P. Chakka, N. Zhang, S. Antony, H. Liu and R. Murthy, "Hive - a petabyte scale data warehouse using Hadoop," 2010 IEEE 26th International Conference on Data Engineering, March, 2010, pp.996-1005.

[18] S. Raschka, "Naive Bayes and Text Classification I-Introduction and Theory." arXiv preprint arXiv:1410.5329, 2014.