JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

ANALYSIS OF PRIVACY DATA DETECTION AND CLASSIFICATION SYSTEM USING MULTI CLASSIFIERS

Aarthy.N[1] ,Devi.P.P[2]
aarthy.n@aalimec.ac.in[1],pp.devi@aalimec.ac.in[2]
Asst.Professor[1,2],Department of Computer Science and Engineering[1,2]
Aalim Muhammed Salegh College Of Engineering[1,2]

Abstract—Micro-blogging services like Twitter which allow users to post messages and follow activities are gaining in popularity. Part of the appeal is the ability to follow any user with a public profile, enabling them to communicate with each other and view each other’s posts directly. The content of the posted tweets is wide ranging, and sometimes includes private information like email addresses, physical addresses, birthdays and medical history. Such private data, if leaked through public posts, could be used by stalkers, foes, or unintended parties. Detecting the presence of private data in tweets is a first step towards analyzing the privacy risk associated with it. In purposed work is categorize the tweets into private and non-private, based on whether they reveal any private information or not. We train the model on novel features extracted from the labeled tweets and perform supervised classification. Our results show that detection of private data classification can be achieved with an accuracy of about 82.5% .Furthermore, we are able to differentiate between private tweets by classifying them into different categories with high accuracy.

Keywords:Classification, Privacy, Social Networks, Twitter

I.INTRODUCTION:

Today’s individual computational gadgets, from desktops to versatile PCs, advanced cells and purchaser hardware, run a wide assortment of utilizations that send client data over the system to different gatherings. This data is utilized as a part in numerous profitable ways e.g., current area data is utilized to interface with maps [3].In any case, breaks of individual data are additionally a potential reason for concern since they might attack security in ways that were not expected or coveted by clients, e.g., when outsiders assemble definite profiles of client conduct. So, the proprietors of data must know about the data uncovered by their applications or by their clients with the goal that they can evaluate whether it causes a risk to their protection and classification issues . Likewise, individuals who direct machines stand to profit by learning of what applications are uncovering which data to whom, with the goal that they can better survey and set protection and security arrangements. Be that as it may, it is extremely hard to know how individual data is uncovered by arranged applications. Clients should depend vigorously on portrayals gave by application engineers (or advertisers) since there is no genuine, autonomous approach to naturally confirm what is revealed by whom. Apparatuses to check system movement for individual data are ordinarily constrained to a little arrangement of very much characterized data, for example, MasterCard or government managed savings numbers . Along these lines, numerous data holes are found unintentionally. So to achieve protection and security issues of an association a framework must be utilized at the level of association itself for forestalling and identifying potential holes of touchy data by machine learning strategies

II.RELATED WORKS:

In an existing Works the realizations of privacy data detection require the plaintext sensitive data. However, this requirement is undesirable, as it may threaten the confidentiality of the sensitive information. If a detection system is compromised, then it may expose the plaintext sensitive data. Shingle with Rabin fingerprint was used previously for identifying similar spam messages in a collaborative setting as well as collaborative worm containment virus scan and fragment detection. Privacy requirement does not exist in above model, because if a detection system is compromised, then it may expose the plaintext sensitive data. Iterations for sensitive data go undetected. Our proposed model overcomes these issues. It enables the data owner to securely implement the detection using the keywords and links . Overall the system will give reliability to stack holders..

Data leakage is found to be a serious problem for any type of organization. In spite of the size of the organization data leaks can create a serious impact in their business and many proven examples has been found in the past, hence data leakage has to be prevented by any means. Most of the organizations have given more efforts in preventing their data from loss which might happen intentionally or accidentally. In finding the effective DLP both researchers and professionals are continually putting more efforts to make an effective DLP as DLPs are proved and recognised as the better solution for identifying, monitoring and protecting confidential data. The main aim of the survey was to find out the limitations of the existing DLPs, hence the security gaps could be identified and creating attention. Most of the methods involve manual interpretation a mandatory one as they perform well in classified data. Hence the future privacy detection system that has to be defined must not require manual influences for classifying the data instead artificial intelligence could be applied which might effectively classify the data and produce a better optimized privacy detection system. The future privacy detection system must be performing based on content based analysis instead of context based as content based could be more effective, as they retain a copy and compares the output of the classifiers. Finally the future system must be easy to classify the tweets among the best accuracy.

III.PROPOSED WORK:

Fig. 1 demonstrates the flow diagram of our privacy data detection model. We crawl tweets in real time with the help of Twitter Streaming API and store them in MongoDB. It is difficult to perform text analysis on unstructured text like tweets due to the presence of metadata (hashtags, user mentions, URL) which results into irregularities and ambiguities. So preprocessing of tweets is a critical step to reduce the amount of information that must be searched for detecting private content. After the pre-processing step, we represent each tweet in the form of feature vector combining different features extracted from tweets. We train classifiers on these set of features extracted from pre-labeled tweets. We compare the performance of three classifiers and choose Random Forest classifier since it gives better performance on the data-set. Once the tweet is identified as a private tweet based on the privacy categories d we pass it through a second level accuracy based on the final comparision. This classification task done on identifiying the content of private tweet.

A. Collection of twitter data: We collect data using Twitter Streaming API by restricting the language of tweets to English. We use the wrapper library called tweepy to interact with Twitter Streaming API using Python Tweets extracted from streaming API are in the JSON format. It requires a large relational database to store all the fields from JSON encoded tweets. Also, JSON parser is required to convert JSON text into object that can be stored into database. It gives error upon receiving unexpected or missing fields. So we use MongoDB to store our data, because it greatly simplifies this problem of tweet storage by eliminating the need of a JSON parser. It is a documentbased database that uses documents instead of tuples in tables to store data2 .

B. Pre-processing of Tweets:Tweets posted by users on Twitter do not follow proper English sentence structure or grammar. Since each tweet is restricted to 140 characters, most of the times it is found to include abbreviations, slang, hashtags and emoticons. Metadata such as hashtags, sometimes reveal private information and hence need to be considered. We extract only English tweets from streaming API for detecting privacy leak. For reducing features to a manageable size, we replace all the words starting with the symbols @,# with equivalent tokens as [MENTION] and [HASHTAG] respectively. For this purpose, we use python library called PyEnchant for spell checking. Tweets are a blend of abbreviations, slang and context specific terms like hashtags, therefore we further feed these processed tweets through the systems developed. The framework converts each tweet into proper English sentence and also recognizes words in long hashtags. As the last step of pre-processing, we use stop word removal method to eliminate redundant words inefficient for classification . NLTK library has an exhaustive list of stop words in standard English language. For example, the words like ‘is’, ‘the’, ‘of’, ‘this’ will be removed with the stop word removal technique and the remaining words will be segmented .

C. Feature Extraction :Feature extraction is a critical step in training the classifier. Our feature set includes three binary features and one numerical feature . We perform the classification of tweets in supervised learning approach, we train classifiers with the labeled tweets. This is achieved in two phases — training and testing.In the training phase, we format all the extracted features explained in the above section as a vector F ={f1,f2,. . . , fn} which is combined with the labels as one input instance. Training data represents an input feature vector and the label, (F, label), where the label is either private or non-private. Here the training data-set is denoted as the vector TS = {(F1,label1), (F2,label2),. . . , (Fn,labeln)}.

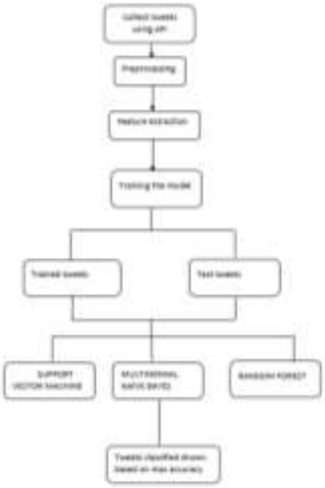


Fig 1:System Architecture

D. Classification:We train the classifier by feeding the complete training set to the machine learning algorithm. In the testing phase, we use this trained classifier to automatically label the testing data set T = {F1,F2,. . . ,Fn}.

We use three algorithms for testing — Naive Bayes (NB), Support Vector Machine (SVM) and Random Forest (RF). Naive-Bayes classifier produces results based on the Bayes Theorem. The underlying assumption for this classifier is that all the features are independent of each other. We choose this classifier since it is easy to train and suitable for textual classification . Support Vector Machine is a form of classifier which attempts to determine good linear separation between different classes. We choose this classifier because SVM classification gives good results on text data due to sparse nature of text .Random Forest classifier uses many decision tree models to give estimates of important variables in the classification task. To classify the test set, we submit the feature vector as an input to each of the tree. The class having the most votes across all of the trees is the label of that particular test feature vector. We choose RF classifier because it runs efficiently on large data sets and also gives good results for mixed feature set. B. Multiclass classification of private tweets has been done to classify as private or non-private, we feed all the private tweets through second classifier which detects the category of private data being revealed using 'Term Frequency-Inverse Document Frequency' (TF-IDF) model [16]. Term Frequency (TF) denotes the frequency of word appearing in a tweet. High value of TF indicates, high weight for the feature. Document frequency (DF) is the count of tweets in dataset that contain a specific word. Higher the value of DF, lower the importance of the feature. Inverse Document Frequency (IDF) for a feature is calculated as follows [16]: IDF = log(N/DF) where N is the number of tweets. Finally, TF-IDF score for a feature is computed as: TF-IDF = TF ∗ IDF After representing each tweet in form of TF-IDF feature, we use Multinomial Naive Bayes classifier for further classification into multiple classes. We choose this classifier because it is good for text classification where data are represented as TF-IDF vectors [16]. It considers the frequency of words appearing in a tweet and denoted as follows [16]: $P(c|t) \propto P(c)$ Y $1 \leq k \leq nd$ $P(wk|c)$ where $P(c|t)$ is the probability of a tweet belonging to class c, $P(c)$ is the prior probability of a tweet occurring in class c, and $P(wk|c)$ is the probability of word wk given class c .

Performance Evaluation:We evaluate the performance of classifiers by using following metrics:

True Positives (TP) - These are the correctly predicted positive values, which means that the value of actual class is yes and the value of predicted class is also yes.

True Negatives (TN) - These are the correctly predicted negative values which means that the value of actual class is no and value of predicted class is also no. False positives and false negatives, these values occur when your actual class contradicts with the predicted class.

False Positives (FP) – When actual class is no and predicted class is yes.

False Negatives (FN) – When actual class is yes but predicted class in no.

Accuracy - Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. If there is high accuracy then the model is considered to be the best and provide better expected result. This measure can be used only when there is symmetric datasets where values of false positive and false negatives are almost same. The proposed model has achieved an accuracy of 96% for the dataset under consideration.

Accuracy = TP+TN/TP+FP+FN+TN

Precision - Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. High precision relates to the low false positive rate. Here this detection system have got nearly 1.00 precision which is pretty good.

Precision measures the percentage of tweets flagged as private tweets that were correctly classified

Precision = TP/TP+FP

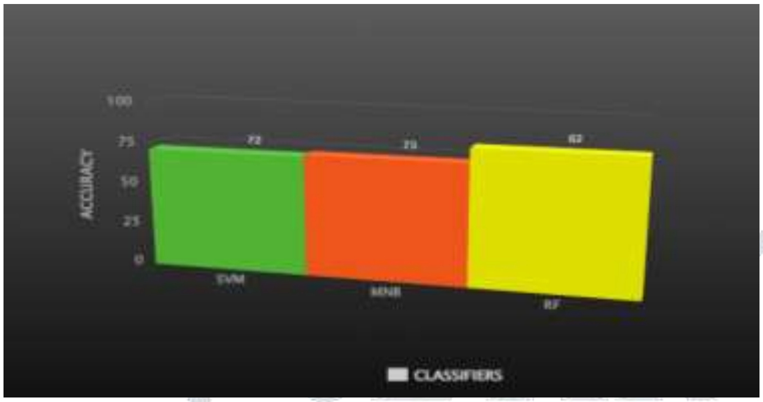


Fig 2:visualization of classifiers accuracy

Recall - Recall is the ratio of correctly predicted positive observations to all positive observations in actual class. The proposed system has got a recall of 1.00 which is good for this model.

Recall measures the percentage of actual private tweets that were correctly classified.

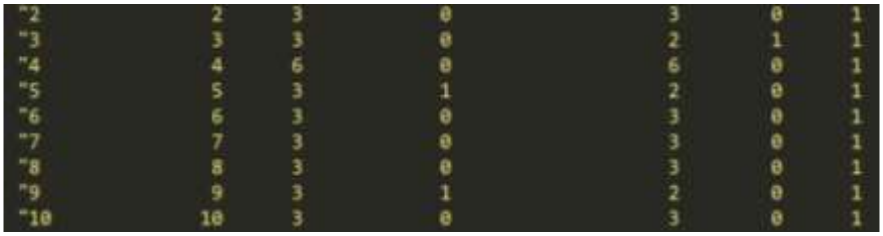
Recall = TP/TP+FN

F1 score - F1 Score is the weighted average score of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. F1 is usually more useful than accuracy, especially if you have an uneven class distribution. Accuracy works best if false positives and false negatives have similar cost. If the cost of false positives and false negatives are very different, it’s better to look at both Precision and Recall. In our case, F1 score is 1.00.

F1 Score = 2*(Recall * Precision) / (Recall + Precision)

Table 1:accuracy values

No	CLASSIFIERS	Classification of tweets
1.	SUPPORT VECTOR MACHINES	74%
2.	MULTINOMIAL NAÏVE BAYES	73%
3.	RANDOM FOREST	82.5%



```
"2             2    3         0                 3      0     1
"3             3    3         0                 2      1     1
"4             4    6         0                 6      0     1
"5             5    3         1                 2      0     1
"6             6    3         0                 3      0     1
"7             7    3         0                 3      0     1
"8             8    3         0                 3      0     1
"9             9    3         1                 2      0     1
"10           10    3         0                 3      0     1
```

Fig 3: Classification tweets

CONCLUSION:

In this paper we introduced a multiclasssifer technique for detection and classification of private tweets. Developed model detects private tweets using three classifier and also predicts the nature of data revealed through those tweets. Our results show that, we are able to detect private tweets with an accuracy of 82.5% and also predict the type of content revealed through the private tweet with highest accuracy produced by the classifiers. In future work consider tweets privacy and recommend to provide more security along with the classification and detection

REFERENCES:

[1] S. Schechter, A. B. Brush, and S. Egelman, “It’s no secret. measuring the security and reliability of authentication via ”secret” questions,” in 30th IEEE Symposium on Security and Privacy, 2009, pp. 375–390.

[2] Y. Wang, G. Norcie, S. Komanduri, A. Acquisti, P. G. Leon, and L. F. Cranor, “”I Regretted the Minute I Pressed Share”: A Qualitative Study of Regrets on Facebook,” in Proceedings of the Seventh Symposium on Usable Privacy and Security, ser. SOUPS ’11. ACM, 2011, pp. 10:1–10:16.

[3] X. Yan, J. Guo, Y. Lan, and X. Cheng, “A biterm topic model for short texts,” in Proceedings of the 22nd international conference on World Wide Web, 2013, pp. 1445–1456.

[4] S. Brindha, K. Prabha, and S. Sukumaran, “A survey on classification techniques for text mining,” in 3rd International Conference on Advanced Computing and Communication Systems (ICACCS), vol. 1. IEEE, 2016, pp. 1–5.

[5] R. Narmadha and G. Sreeja, “A survey on online tweet segmentation for linguistic features,” in International Conference on Computer Communication and Informatics (ICCCI). IEEE, 2016, pp. 1–6