International Journal of Scientific Research and Engineering Development-– Volume 4 Issue 4, July- Aug 2021

Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Hate Speech Detection Using KNN and SVM

Vijay*, Dr. Pushpneel Verma**

*(Department of Computer Science and Engineering, Bhagwant University, Ajmer
Email: vijay.movfrose@gmail.com)

** (Department of Computer Science and Engineering, Bhagwant University, Ajmer
Email: pushpneelverma@gmail.com)

----------**********----------

Abstract:

Extremist text posts, cyberbullying and propagating hate speech on social media have become a major problem nowadays. Toxic and hate provoking contents on social media can arouse communal hatred which can result in violence or riots. It has become necessary to curb the spread of hate speech texts on social media. Social media is a powerful instrument which can be used for various constructive purposes like promoting business. The power of social media should not be misused. It should not be used for destructive purposes. There is a need of automated system for detecting hate speech texts on social media. In this paper we used with two supervised machine learning algorithms for classifying text documents as containing hateful content and not containing hateful content. These two algorithms are KNN (K-Nearest Neighbor) algorithm and SVM (Support Vector Machine). In this research paper we have discussed the steps which we followed while performing experiments and then we presented the performance of these algorithms in form of confusion matrix.

Keywords — **Support Vector Machine, KNN (K-Nearest Neighbor). Text Classification, hate speech.**

----------**********----------

I. INTRODUCTION

Communication which disparages someone on the basis of characteristics like color, race, gender, nationality, religion and ethnicity etc. is called hate speech [1]. As the content on social media is growing steadily the hate speech content is also increasing [2]. Methods are required for automatic detection of hate speech. Increase in toxic and hateful content on social media has motivated the researchers to put efforts for identifying hateful content [3]. As people can easily express their opinions on social network there is increase in spread of online hate speech on social network platform [4]. Hate speech on social media can have many destructive offline effects. Extremists usually use social media to arouse hatred among people of different communities. Hate speech texts and other toxic content placed on social media by extremists can provoke communal disharmony and communal violence in some cases. Such anti-social activities on social media cannot be ignored.

In this paper we used two supervised machine learning algorithms for classifying text documents as hateful and not hateful. We used Support Vector Machine and K-Nearest Neighbor algorithm in this paper. Supervised machine learning algorithms require already labelled data for training purpose. In this case the problem of hate speech detection is the problem of text classification. First the text classifier is trained with already classified or labelled training data then this trained text classifier can be used for predicting the class labels of unlabelled text documents. Text classification is the problem of the field of Natural Language Processing (NLP) [8]. Text classifiers used in this paper were able to classify text documents on the basis of their content as hateful and not hateful.

In our experiments we compared the performance of Support Vector Machine (SVM) with linear kernel to the SVM with radial kernel. SVM is a supervised machine learning algorithm. It was proposed was Vapnik [5]. For text classification the performance of SVM is prominent [6].

The K – Nearest Neighbor (KNN) algorithm is simple and efficient for handling various types of tasks of text classification. Therefore, KNN is widely used for text classification [7]. In KNN algorithm the major problem is to find the appropriate value of K so that classification effectiveness is high.

II. SUPPORT VECTOR MACHINE (SVM)

SVM can be used for text classification. It is a supervised machine learning algorithm. SVMs are also known as kernel machines [9]. SVM classifier has high performance [6]. In SVM algorithm a data item with n features is represented as a point in n-dimensional space in such a way that the value of a feature is the value of its corresponding coordinate. Then a hyperplane is found in this n-dimensional space so that data points are distinctly classified. In our case each text document belongs to either one of two classes: hateful and not hateful. Many hyperplanes are possible which can separate the data points of these two classes but we need to select the hyperplane with maximum margin. Margin is the distance between the hyperplane and closest data points. For example, in Fig. 1 it is shown that there are data points of two different classes in 2- dimensional space. As it is shown in Fig. 1 that three hyperplanes A, B, and C which are classifying the data points distinctly but the hyperplane C is with maximum margin. Therefore, hyperplane C will be selected. Data points on a particular side of hyperplane belong to one class and data points on other side of hyperplane belong to a different class.

A group of mathematical functions called kernels are used by SVM algorithms. The kernel function transforms the input data into the desired form. There are different kinds of kernel functions like linear, nonlinear polynomial, sigmoid, and radial basis function (RBF). In this research paper we did experiments with linear kernel function and radial basis function (RBF).

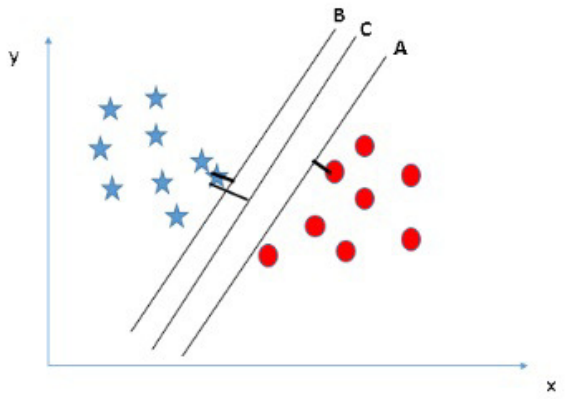

Fig. 1 Three hyperplanes separating data points of two different classes in 2-dimensional space.

III. KNN

For classification KNN is an effective method [11]. The K – Nearest Neighbor (KNN) algorithm can be used for text classification. In this algorithm the class of a text document is determined on the basis of the classes of the K documents which are nearest to it [10]. The KNN is a statistical classification algorithm. KNN is a lazy learning algorithm. In training phase no actual learning takes place in KNN algorithm and in testing phase all training data are required. When the class of an unclassified document is to be predicted the KNN algorithm finds K closest neighbors from training data, and by performing voting among these K closest neighbors the class is assigned to the unclassified document. In terms of memory and time the testing phase of KNN algorithm is costly [12]. In this research paper we used Euclidean distance metric for calculating the nearest distance in KNN algorithm.

Euclidean distance:

Let x and y are two records with n attributes.

$x = x_1, x_2, x_3, \ldots, x_n$ and $y = y_1, y_2\ y_3 \ldots, y_n$. The Euclidean distance between x and y is calculated by the formula given below:

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

IV. METHODOLOGY

The dataset which we used for experimentation was "2020-12-31-DynamicallyGeneratedHateDataset-entries-v0.1". We downloaded it from kaggle.com [13]. This dataset had eleven columns but we kept only two columns "text" and "label" for performing experiments. The column "text" contains text documents and the column "label" specifies the category of each of these text documents. The column "label" has only two values "hate" and "nothate". In column "label" the value "hate" specifies that text document is hateful and the value "nothate" specifies that the text document is not hateful. This dataset has 40623 text documents classified as "hate" and "nothate". First we shuffled the text documents well and then selected 10,000 text documents only for experimentation. After that we converted all the text documents into lower case and then removed all punctuations, numbers, stop words, and URLs. After that we stripped white spaces. Finally we converted the text documents into document term matrix.

A. Experiment with Support Vector Machine (SVM)

```
> conf.mat
Confusion Matrix and Statistics

          Reference
Prediction hate nothate
   hate     902     479
   nothate  221     398

               Accuracy : 0.65
                 95% CI : (0.6286, 0.6709)
    No Information Rate : 0.5615
    P-Value [Acc > NIR] : 4.709e-16

                  Kappa : 0.2656

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.8032
            Specificity : 0.4538
         Pos Pred Value : 0.6531
         Neg Pred Value : 0.6430
             Prevalence : 0.5615
         Detection Rate : 0.4510
   Detection Prevalence : 0.6905
      Balanced Accuracy : 0.6285

       'Positive' Class : hate

>
```

Fig. 2 Confusion matrix and statistics for testing by SVM with linear kernel function

When we did experiment with SVM, we used 8000 rows of document term matrix corresponding to 8000 text documents for training purpose and remaining 2000 rows of document term matrix corresponding to remaining 2000 text documents were used for testing purpose. Among 8000 text documents of training data 4376 were labelled as "hate" and remaining 3624 were labelled as "nothate". In test data among 2000 text documents 1123 were labelled as "hate" and remaining 877 were labelled as "nothate". SVM with linear kernel function gave us 65% accuracy. Fig. 2 shows the confusion matrix and statistics for testing performed by SVM with linear kernel function. When we did experiment with SVM with RBF (radial basis function) kernel we got 63.8% accuracy.

B. Experiment with K- Nearest Neighbor (KNN)

```
> prediction<-knn(dtm.train, dtm.test, df.train$class, k = 3)
> confusionMatrix(prediction, df.test$class)
Confusion Matrix and Statistics

          Reference
Prediction hate nothate
   hate     864     486
   nothate  259     391

               Accuracy : 0.6275
                 95% CI : (0.6059, 0.6487)
    No Information Rate : 0.5615
    P-Value [Acc > NIR] : 1.211e-09

                  Kappa : 0.2215

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.7694
            Specificity : 0.4458
         Pos Pred Value : 0.6400
         Neg Pred Value : 0.6015
             Prevalence : 0.5615
         Detection Rate : 0.4320
   Detection Prevalence : 0.6750
      Balanced Accuracy : 0.6076

       'Positive' Class : hate
```

Fig. 3 Confusion matrix and statistics for testing by KNN with K=3.

First we removed the terms from the document term matrix having sparsity greater than or equal to 99.6%. 8000 rows of document term matrix corresponding to 8000 text documents were used for training purpose and remaining 2000 rows of document term matrix corresponding to 2000 text documents were used for testing purpose. In training data among 8000 text documents 4376 were labelled as "hate" and remaining 3624 were labelled as "nothate". In test data among 2000 text documents 1123 were labelled as "hate" and

remaining 877 were labelled as "nothate". The KNN algorithm with K=1 gave us 60.8% accuracy, with K=2 we got 62.65% accuracy and with K=3 we got 62.75% accuracy. Fig. 3 shows the confusion matrix and statistics for testing by KNN algorithm with K=3.

V. CONCLUSION

We did experiments with Support Vector Machine with linear kernel and RBF kernel and K-Nearest Neighbor algorithm with K=1, K=2 and K=3. We got maximum accuracy of 65% from SVM with linear kernel function.

REFERENCES

[1] John T. Nockleby. 2000. Hate Speech. In Leonard W. Levy, Kenneth L. Karst, and Dennis J. Mahoney, editors, Encyclopedia of the American Constitution, pages 1277–1279. Macmillan, 2nd edition.

[2] Anna Schmidt, Michael Wiegand. A Survey on Hate Speech Detection using Natural Language Processing Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media , pages 1–10, Valencia, Spain, April 3-7, 2017. c 2017 Association for Computational Linguistics.

[3] Mozafari M., Farahbakhsh R., Crespi N. (2020) A BERT-Based Transfer Learning Approach for Hate Speech Detection in Online Social Media. In: Cherifi H., Gaito S., Mendes J., Moro E., Rocha L. (eds) Complex Networks and Their Applications VIII. COMPLEX NETWORKS 2019. Studies in Computational Intelligence, vol 881. Springer, Cham. https://doi.org/10.1007/978-3-030-36687-2_77

[4] Zewdie Mossie, Jenq-Haur Wang, Vulnerable community identification using hate speech detection on social media, Information Processing & Management, Volume 57, Issue 3, 2020, 102087, ISSN 0306-4573, https://doi.org/10.1016/j.ipm.2019.102087.

[5] V. Vapnik, "The nature of statistic learning theory. Springer, New York, 1995.

[6] Liu, Z., Lv, X., Liu, K., & Shi, S. (2010). Study on SVM Compared with the other Text Classification Methods. 2010 Second International Workshop on Education Technology and Computer Science. doi:10.1109/etcs.2010.248

[7] Chin Heng Wan, Lam Hong Lee, Rajprasad Rajkumar, Dino Isa, A hybrid text classification approach with low dependency on parameter by integrating K-nearest neighbor and support vector machine, Expert Systems with Applications, Volume 39, Issue 15, 2012, Pages 11880-11888, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2012.02.068.

[8] Chen, Z., Zhou, L. J., Li, X. D., Zhang, J. N., & Huo, W. J. (2020). *The Lao Text Classification Method Based on KNN. Procedia Computer Science, 166, 523–528.* doi:10.1016/j.procs.2020.02.053

[9] Kecman V. () Support Vector Machines – An Introduction. In: Wang L. (eds) Support Vector Machines: Theory and Applications. Studies in Fuzziness and Soft Computing, vol 177. Springer, Berlin, Heidelberg. https://doi.org/10.1007/10984697_1

[10] Vishwanath Bijalwan, Vinay Kumar, Pinki Kumari and Jordan Pascual. *KNN based Machine Learning Approach for Text and Document Mining.* International Journal of Database Theory and Application Vol.7, No.1 (2014), pp.61-70. http://dx.doi.org/10.14257/ijdta.2014.7.1.06

[11] Guo G., Wang H., Bell D., Bi Y., Greer K. (2003) KNN Model-Based Approach in Classification. In: Meersman R., Tari Z., Schmidt D.C. (eds) On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE. OTM 2003. Lecture Notes in Computer Science, vol 2888. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-39964-3_62

[12] Indu Saini, Dilbag Singh, Arun Khosla, QRS detection using K-Nearest Neighbor algorithm (KNN) and evaluation on standard ECG databases, Journal of Advanced Research, Volume 4, Issue 4, 2013, Pages 331-344, ISSN 2090-1232, https://doi.org/10.1016/j.jare.2012.05.007.

[13] https://www.kaggle.com/usharengaraju/dynamically-generated-hate-speech-dataset?select=2020-12-31-DynamicallyGeneratedHateDataset-entries-v0.1.csv

