Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 03, April-May 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.23.17.23

A Combine Model for Email Classification in Hindi Language using Supervised Learning (NB, K-NN, DT, SVM)

Dr. Ishaan Tamhankar*

[*]Assistant Professor, Vimal Tormal Poddar BCA College Gujarat, India.

Corresponding Email: [*]prof.ishaantamhankar@gmail.com

Received: 15 January 2022 **Accepted:** 01 April 2022 **Published:** 02 May 2022

Abstract: ***Email communication is necessary in today's environment, yet unwanted emails create issue in such communication. The current study focuses on developing an Email classification model for the use of classifiers approaches. The goal of this research is to classification of emails based on features. For classification especially in Hindi language of the email dataset Different machine learning classifiers such as Naïve Bayes, Decision Tree, K-Nearest Neighbor and Support Vector Machine used in research work as well as we used combined model also for optimum accuracy.***

Keywords: ***Naïve Bayes (NB), K-Nearest Neighbor (K-NN), Decision Tree (DT), Support Vector Machine (SVM).***

1. INTRODUCTION

Email is becoming one of the most important and effective means of communication in personal and business life. Some users abuse their email by sending computer worms and spam, which are unwanted information sent to their email inbox. According to statistics, the average number of daily spam email messages sent in 2014 reached 54 billion. Spam mail overloads your email server and consumes network bandwidth and storage capacity. Therefore, email filtering is a very important process to solve these problems. The purpose of the filter is to identify and isolate spam emails [5].

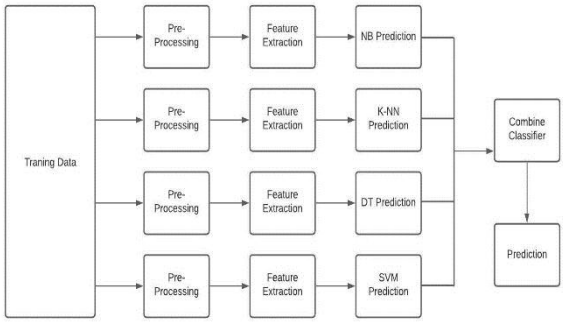
Many mail server engines are the usage of numerous authentication mechanisms to research Email content material and categorize the Emails into white and black lists so; they may be optimized with the aid of using users. Using white and black lists, the New Email supply is as compared with a database to recognize if it's far labelled as junk mail earlier than or not [9]. On every other side, an opportunity technique filters Emails with the aid of using extracting capabilities from the Email frame and the usage of a few type methods, together with Naive

Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 03, April-May 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.23.17.23

Bayes (NB), Random Forest (RF), Support Vector Machine (SVM), and Neural Networks (NN). Most of the associated works classify emails the usage of the time period that happens withinside the e mail. Some few works moreover bear in mind the semantic homes of the e-mail text.[3] Integrating semantic ideas and processes for e mail type is predicted to feature critical blessings of improving the computational performance, similarly to the accuracy of type [11].
In this perspective, an email spam-based classifier isn't simply expected to precisely order spam messages as spam yet additionally expected to characterize non-spam messages as non-spam or typical [14]. This is since both are viewed as conditions for assessing the nature of its characterization or expectation. In this Paper we have Focused on Classification approach specially for Hindi Language and classified in various categories like Bank, Entertainment, Education, Spiritual, Sports, Others.

2. METHODOLOGY

In this research paper I have used combined approach and created new algorithm for classification using naïve bayes, Support vector machine decision tree and K-Nearest Neighbor. I have implemented in Python as well as in MATLAB.
Combined Model shown in below figure:



Architecture of Email Classification using Supervised Learning Combined Approach
Combine model algorithm steps are as follows:

- Data Pre-processing step
- Predicting the Train Data
- Test accuracy of the result
- Visualizing the test set result.

2.1 Pre-Processing

Tokenization dependency libraries
This package tends to implement a Tokenizer and a stemmer for Hindi language. To import the package,
from HindiTokenizer import Tokenizer or
from sklearn.feature_extraction.text import CountVectorizer

Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 03, April-May 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.23.17.23

Steaming

For the Hindi language, there is no automation tool is available to create stemmed words list from dataset or corpus. We have used hand crafted Hindi list in order create a list of stemmed words

```
from nltk.stem.porter import PorterStemmer
from nltk.stem.lancaster import LancasterStemme
word=t.generate_stem_word()
```

Stop word Elimination

Till now, there is no unique stop words list is available for Hindi language. With the help of linguistic experts and by manual inspection, we have manually constructed a list of 531 stop words. This stop words list is only domain specific that includes sports, entertainment, health, business, spiritual and astrology.

```
from nltk.corpus import stopwords
t.remove_stopwords()        or
cv=CountVectorizer()
features= cv. fit_transform(x_t
```

in below diagram shows how Pre-processing works and shows pseudocode and shows first 560 train data and after 560 there are test data pre-processing steps

```
import pickle
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
df=pd.read_csv("emailsclassi.csv")
x = df["Subject"]
y = df["feature"]
# x_train,y_train = x[0:560],y[0:560]
# x_test,y_test = x[560:],y[560:]
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test= train_test_split(x, y, test_size= 0.25, random_state=0)
##Step3: Extract Features
cv = CountVectorizer()
features = cv.fit_transform(x_train)
```

Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 03, April-May 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.23.17.23

2.2. Predicting the Train Data

Here, we are Predicting the train data of every classifier algorithm like NB, K-NN, SVM and DT.

```
tree_algo=loaded_model.predict(vect)
knn_algo=classifier1.predict(vect)
svm_algo=clfrbf.predict(vect)
nb_algo=classifier.predict(vect)
```

2.3. Classify data using Prediction (NB, K-NN, DT, SVM) and Store

```
        names = []
names.extend([svm_algo[0],tree_algo[0],nb_algo[0],knn_algo[0]])
# print(names)
countnames = {}
for name in names:
    if name in countnames:
        countnames[name] += 1
    else:
        countnames[name] = 1
#   print(countnames)
#   print(svm_algo[0],tree_algo[0],nb_algo[0],knn_algo[0])
#   print(countnames)
# print(countnames)
key_list = list(countnames.keys())
val_list = list(countnames.values())
# print("key_lst => ",key_list)
# print("val_lst => ",val_list)
lst =countnames.values()
indexno =max(lst)
position = val_list.index(indexno)
#   print("Feature :",key_list[position],"\n","Values: ",d["Subject"])
if key_list[position]=="bank":
    datalst.append(key_list[position])
elif key_list[position]=="education":
    datalst.append(key_list[position])
elif key_list[position]=="entertainment":
    datalst.append(key_list[position])
elif key_list[position]=="shopping":
-     datalst.append(key_list[position])
elif key_list[position]=="astrology":
    datalst.append(key_list[position])
elif key_list[position]=="sports":
    datalst.append(key_list[position])
else:
    datalst.append(key_list[position])
```

Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 03, April-May 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.23.17.23

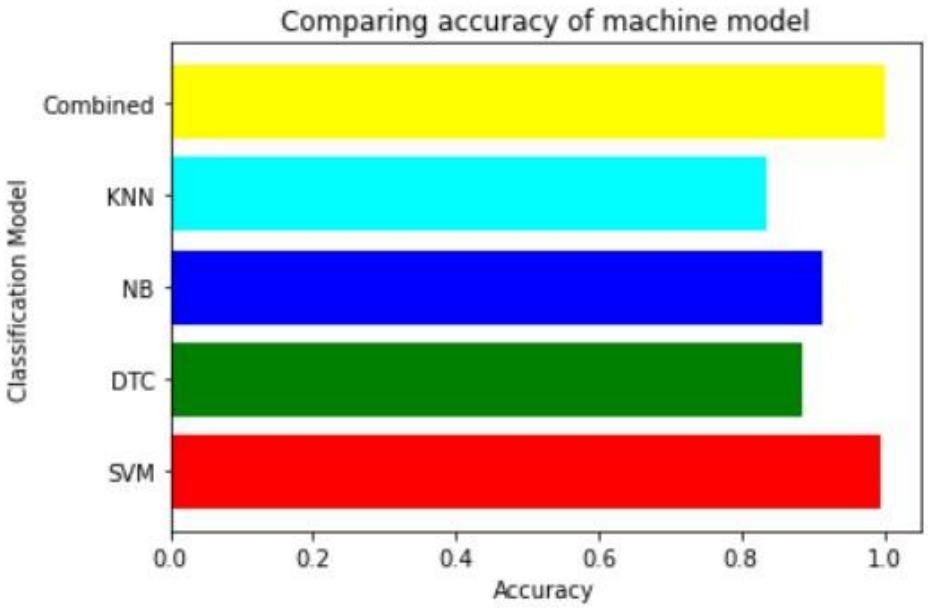
2.4. Checking the Accuracy

```
print ("Prediction time:", round(time()-t1, 3), "s")
print ("Accuracy Score",accuracy_score(y_train,datalst))
combined = accuracy_score(y_train,datalst)
```

Confusion Matrix

```
matrix = confusion_matrix(y_train,datalst)
print('Confusion matrix : \n',matrix)
Prediction time: 2.309 s
Accuracy Score 0.9998
Confusion matrix :
```

```
[[ 25   0   0   0   0   0   0]
 [  0 301   0   0   0   0   0]
 [  0   1 151   0   0   0   0]
 [  0   0   0  22   0   0   0]
 [  0   0   0   0   8   0   0]
 [  0   0   0   0   0  28   0]
 [  0   0   0   0   0   0  24]]
```



Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 03, April-May 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.23.17.23

3. CONCLUSION

In this paper compared NB, K-NN, DT and SVM and Combined New Model and data test show that the fundamental goal is achieved and the classification results are achieved. This section uses the Combined Algorithm (NB, K-NN, DT, SVM) learning classification. Hence in this Implementation Model Achieved **99.98% Accuracy** for Classified Data Set, therefore we can Strongly say Classification is working for Hindi Language and as far as accuracy concern it is up to the mark.

4. REFERENCES

1. M.-S. Chen, J. Han, and P.S. Yu, "Data Mining: An Overview from Database Perspective,"IEEE Trans. Knowledge and Data Eng., vol. 8, no. 6, pp. 866-883, Dec. 1996.
2. R. Agrawal and R. Srikant, "Privacy Preserving Data Mining," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2000.
3. D. Agrawal and C.C. Aggarwal, "On the Design and Quantification of Privacy Preserving Data Mining Algorithms," Proc. 20th ACM SIGMOD-SIGACT-SIGART Symp. Principles of Database Systems (PODS), 2001.
4. Y. Lindell and B. Pinkas, "Privacy Preserving Data Mining," J. Cryptology, vol. 15, pp. 177-206, 2002.
5. C.C. Aggarwal and P.S. Yu, "A Condensation Approach to Privacy Preserving Data Mining," Proc. Ninth Int'l Conf. Extending Database Technology (EDBT), 2004.
6. V.N. Vapnik, Statistical Learning Theory. John Wiley and Sons, 1998.
7. C.J.C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," Data Mining and Knowledge Discovery, vol. 2, no. 2, pp. 121-167, 1998.
8. K. Chen and L. Liu, "Privacy Preserving Data Classification with Rotation Perturbation," Proc. Fifth IEEE Int'l Conf. Data Mining (ICDM), 2005.
9. H. Yu, X. Jiang, and J. Vaidya, "Privacy-Preserving SVM Using Nonlinear Kernels on Horizontally Partitioned Data," Proc. ACM Symp. Applied Computing (SAC), 2006.
10. H. Yu, J. Vaidya, and X. Jiang, "Privacy-Preserving SVM Classification on Vertically Partitioned Data," Proc. 10th PacificAsia Conf. Knowledge Discovery and Data Mining (PAKDD), 2006.
11. J. Vaidya, H. Yu, and X. Jiang, "Privacy-Preserving SVM Classification," Knowledge and Information Systems, vol. 14, pp. 161-178, 2008.
12. S. Laur, H. Lipmaa, and T. Mielika¨inen, "Cryptographically Private Support Vector Machines," Proc. 12th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2006.
13. HIPAA, Standard for Privacy of Individually Identifiable Health Information,http://www.hhs.gov/ocr/privacy/index.html, 2001.
14. J.R. Quinlan, C4.5: Programs for Machine Learning. Morgan Kaufmann, 1993.
15. J. Han and M. Kamber, Data Mining: Concepts and Techniques. Morgan Kaufmann, 2006.
16. B. Mozafari and C. Zaniolo, "Publishing Naive Bayesian Classifiers: Privacy without Accuracy Loss," Proc. 35th Int'l Conf. Very Large Data Bases (VLDB), 2009.

Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 03, April-May 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.23.17.23

17. L. Sweeney, "Uniqueness of Simple Demographics in the US Population," LIDAP-WP4, Carnegie Mellon Univ., Laboratory for Int'l Data Privacy, 2000.
18. L. Sweeney, "Achieving k-Anonymity Privacy Protection Using Generalization and Suppression," Int'l J. Uncertainty, Fuzziness and Knowledge-Based Systems, vol. 10, no. 5,571-588, 2002.
19. A. Inan, M. Kantarcioglu, and E. Bertino, "Using Anonymized Data for Classification," Proc. 25th IEEE Int'l Conf. Data Eng. (ICDE), 2009.
20. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam, "l-Diversity: Privacy beyond k-Anonymity," Proc. 22nd IEEE Int'l Conf. Data Eng. (ICDE), 2006.
21. B. Pinkas, "Cryptographic Techniques for Privacy-preserving Data Mining," ACMSIGKDD Explorations Newsletter, vol. 4, no. 2, pp. 12-19, 2002.