Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 04, June-July 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.24.9.15

An Analysis on Email Classification on Hindi Language using Bayesian Classifier

Dr. Ishaan Tamhankar[1*], Ms. Ritu Bhatiya[2]

[1,2]Assistant Professor, V.T Poddar BCA College Gujarat, India.*

Email: [2]ritubhatiya6194@gmail.com
Corresponding Email: [1]prof.ishaantamhankar@gmail.com*

Received: 25 February 2022 **Accepted:** 15 May 2022 **Published:** 20 June 2022

Abstract: Spam messages are one of the most important problems on the Internet today, costing businesses money and causing frustration to individual users. Spam filtering can assist with the issue in a variety of ways. The classifier-related challenges have been the focus of several spam filtering studies. Machine learning for a spam classification is now a significant research topic. The application of various machine learning techniques for categorizing spam messages from e-mail is investigated and identified in this research. Finally, with spam categorization, a comparative study of the algorithms has been presented.

Keywords: Spam, Email Classification, Machine Learning, Naïve Bayes

1. INTRODUCTION

Spam, also known as unsolicited commercial or bulk e-mail, has recently become a major internet issue. Spam is a waste of time, space, and bandwidth for data transmission. Spam e-mail has been on the rise for several years. According to current figures, spam accounts for 40% of all emails, or 15.4 billion per day, costing internet users $355 million per year. At the moment, automatic e-mail filtering looks to be the most effective method of eliminating spam, and spammers and spam-filtering technology are battling it out. Knowledge engineering and machine learning are two techniques to e-mail filtering. To classify emails as spam or ham, a set of criteria must be specified in the knowledge engineering technique. Either the filter's user or another authority should develop a collection of such rules (for example, the software business that provides a specific rule-based spam-filtering tool). This technique is ineffective since the rules must be altered and maintained on a regular basis, which is annoying for most users and a waste of effort. Machine learning is more efficient than knowledge engineering since it does not require any rules to be specified. Instead, a collection of pre-classified e-mail messages is employed as a set of training examples. The categorization rules are then taught from these e-mail communications using a specific algorithm. There has been a lot of research into machine learning, and there are a lot of algorithms that can be used in e-mail filtering. Artificial Neural Networks and Naive Bayes are two examples.

Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 04, June-July 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.24.9.15

Related Work

Muhammad N. Marsono, M. Watheq El-Kharashi, and Fayez Gebali[2] are three researchers who have used machine learning approaches in e-mail categorization. They showed that the naive Bayes e-mail content categorization could be modified for layer-3 processing without having to reassemble the system. Suggestions on how to use spam control middleboxes to pre-detect e-mail packets in order to assist timely spam detection at receiving e-mail servers were provided. F. Gebali, M. N. Marsono, and M. W. El-Kharashi [1]. They demonstrated the hardware design of a nave Bayes inference engine for spam control using a two-class e-mail categorization system. Given a stream of probabilities as inputs, this can categorise more than 117 million features per second. This study could be expanded to include proactive spam management approaches on receiving email servers and spam throttling on network gateways. Y. Tang, S. Krasser, Y. He, W. Yang, and D. Alperovitch [3] devised a categorization system based on the SVM.This system extracts email sender behavior data based on global sending distribution, analyses them, and assigns a trust value to each IP address sending email message. Yoo, S., Yang, Y., Lin, F., and Moon [11] developed the personalised email prioritisation (PEP) method, which focuses on analysing personal social networks to capture user groups and obtain rich features that represent social roles from the perspective of a specific user, as well as a supervised classification framework for modelling personal priorities over email messages. Guzella, Mota-Santos, J.Q. Uch, and W.M. Caminhas[4] presented the innate and adaptive artificial immune system (IA-AIS), an immunological-inspired model that they applied to the challenge of identifying unwanted bulk e-mail communications (SPAM). It incorporates macrophage-like organisms, B and T cells, and models both the innate and adaptive immune systems. In some parameter combinations, a version of the algorithm was capable of detecting more than 99 percent of legal or SPAMS communications. It was compared against a better-optimized version of the naive Bayes classifier, which has exceptionally high accurate classification rates. It has been determined that IA-AIS has a better capacity to recognise SPAM communications than the implemented naive Bayes classifier, however its ability to identify legal messages is not as good.

2. METHODOLOGY

Naïve Bayes Classifier Working Model:
Hypothesis A opportunities in Visual Event P(A) P(A | B), has a Posterior option.
P (B | A) Opportunities: Evidence opportunities if the hypothesis of probability is true.
P(A) is given an earlier opportunity: hypothesis chances before proof is seen.
With Margin: Evidence Opportunity, P (B) is possible.

The Nave following model may be used to understand how the Bayes' Classifier works:

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}$$

Suppose we have a weather dataset and a target variable called "play." Therefore we have to determine whether we would play in accordance with the conditions of weather to use this data set on a certain day. We must take the following steps to tackle this problem:
Turn the data set into frequency tables. Using this Model we are going Classified Spam Emails

Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 04, June-July 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.24.9.15

in **Astrology, Bank, Education, Entertainment, Others, Shopping, Sports** in Various Categories.

Spam Email Data Set:

From	To	Subject
services@custcomm.icicibank.com	ishaan.tamhankar06@outlook.com	आपके विज़न बोर्ड 2020 में आपके वित्तीय लक्ष्य
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	धनु राशि में बृहस्पति का पारगमन 2020: आपके चंद्रमा पर प्रभाव
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	मेष और मीन 2020 में मंगल का प्रतिगमन: चंद्रमा के संकेतों पर प्रभाव
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	राहु केतु गोचर 2020: 12 चंद्र राशियों के लिए प्रभाव और भविष्यफल
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	पितृ पक्ष 2020 तिथियां, नियम और पूजा प्रक्रिया
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	ज्योतिष के अनुसार अपना लकी रंग और नंबर जानिए
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	ज्योतिष परामर्श से वित्तीय स्थिति में सुधार के उपाय
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	रिश्ता दुविधा? इसे ज्योतिष परामर्श से हल करें
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	संक्रांति | वर्ष 2020 का सबसे लंबा दिन
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	सभी चंद्र राशियों पर पेनुमब्रल चंद्र ग्रहण का प्रभाव
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	अक्षय तृतीया / अखा तीज 2020 तिथि, महत्व और अनुष्ठान
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	आपकी राशि पर चैत्र नवरात्रि 2020 के प्रभाव
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	विवाह में देरी के कारण और उपचार, देर से विवाह ज्योतिष
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	कल्कि अवतार: भगवान विष्णु का दसवां और अंतिम अवतार
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	ज्योतिष में योनी मिलान का महत्व
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	विवाह के लिए नक्षत्र या तारा मिलान
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	राशि चक्र संकेत के अनुसार पहली तारीख गाइड
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	ग्रिहाव प्रवीण मुहूर्त या तिथियां २०२० में - तीथि, समय और नक्षत्र
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	बिजनेस न्यूमेरोलॉजी - बिजनेस सक्सेस के लिए लकी नंबर
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	व्यापार में सफलता के लिए 5 सबसे शक्तिशाली मंत्र
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	शेयर बाजार की भविष्यवाणी के लिए योगासन | शेयर बाजार ज्योतिष
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	अपनी कुंडली के माध्यम से अपनी संपत्ति की संभावनाओं को जानें
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	मेरी शादीशुदा जिंदगी कैसी होगी? विवाह संबंधी समस्याएं और ज्योतिष
ganesha@gmail.ganeshspeaks.com	ishaan.tamhankar06@outlook.com	गोमेद पहनने के ज्योतिषीय लाभ | हेसोनाइट जेमस्टोन

Steps of Algorithm:
Step-1 Data Pre-Processing
Step-2 Fitting Data Set in to Naïve Bayes Algorithm
Step-3 Predicting the Test Result
Step-4 Test accuracy of and Creating Confusion Matrix
Step-5 Visualization of Result.

Data Pre-Processing Step:

```
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. 04, June-July 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.24.9.15

Fitting Naive Bayes classifier to the Training data:

In training data we are now going to equal the Naive Bayes divider. In this regard, we are introducing the sklearn.naive bayes library's MultinomialNB section. We will create a class divider object after introducing the class. Then, in the training data, we measure the separator. Underneath your code:

```
#Fitting Naive Bayes classifier to the training set
from sklearn.naive_bayes import MultinomialNB
nb = MultinomialNB()
nb.fit(features,y_train)
```

```
Out[24]: MultinomialNB(alpha=1.0, class_prior=None, fit_prior=True)
```

Output: if you execute the above code, the output is as follows
Predicting Test Results: We will create a y pred vector as in the logistic regression in order to predict the test of set results. Underneath your code:

```
import sys
from time import time
from sklearn.metrics import accuracy_score
# print ("Training time:", round(time()-t0, 3), "s")
t1=time()
y_pred=nb.predict(features)
print ("Prediction time:", round(time()-t1, 3), "s")
print ("Accuracy Score",accuracy_score(y_train,y_pred))
```

Output:

```
Prediction time: 0.001 s
Accuracy Score 0.9142857142857143
```

Creating the Confusion Matrix:

In order to see the precision of the split, we will build a confusion matrix for our Naive Bayes model now. Underneath your code:

```
from sklearn.metrics import multilabel_confusion_matrix
cm = multilabel_confusion_matrix(y_train, y_pred)
print(cm)
```


Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 04, June-July 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.24.9.15

We can therefore say that the performance in the model is improved by means of the K-NN algorithm in the above chart, 532 + 17 = 549 correct predictions, and 8 + 3 = 11 incorrect forecasts.

Visualizing the Training set result:

The training results for the model from Naive Bayes will now be visualised. With the exception of the graph name, the code is always the same as the KNN and SVM code. Underneath your code:

```
from sklearn.preprocessing import LabelEncoder
lb=LabelEncoder()
cl=lb.fit_transform(y_train)

plt.scatter(x_train, classifier.predict(cv.transform(x_train)), c=cl, cmap='winter')
plt.show()
plt.close()
```

Output :

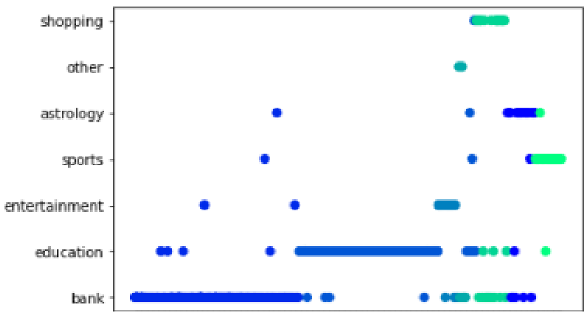

Figure 1. NB Visualizing Spam Email Data

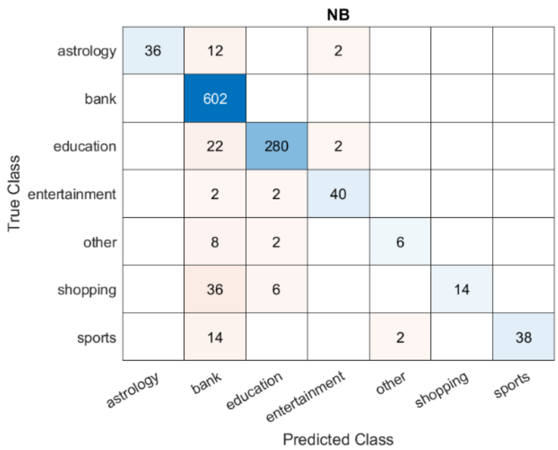

Figure 2. NB Confusion Matrix

Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 04, June-July 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.24.9.15

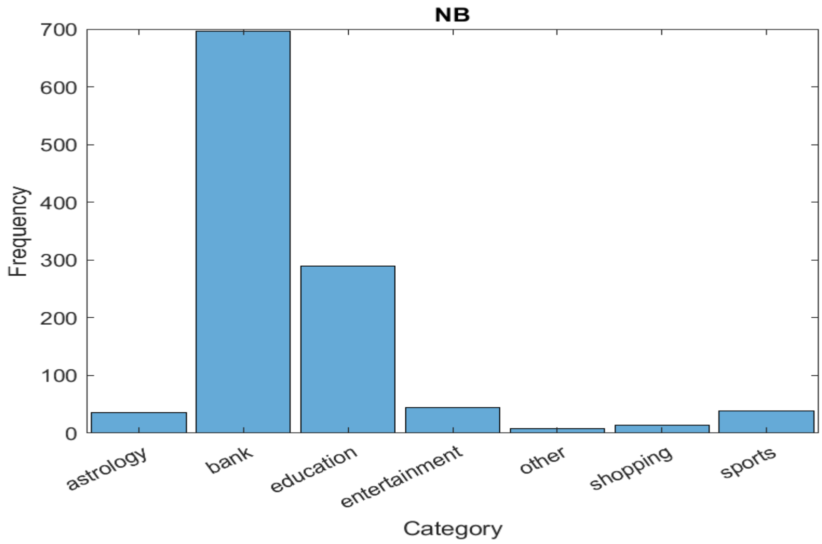

Figure 3. Classified Spam Email for Naïve Bayes

3. CONCLUSION

The data test results show that the primary goal has been met, as well as the categorization findings. This section uses the NB machine learning classification. Hence in this Implementation Model Achieved **91% Accuracy** for Classified Data Set. The Algorithm for the NB division is yours, as the distance scale must be set. Because distance understanding is limited, the effect of separation is totally determined on the distance used. As a result, specialists must determine whether the result is based on a set of data, two distinct algorithms, and two wholly different conclusions. The use of distinct grades is eliminated because it is often dynamic to recognize results.

4. REFERENCES

1. S.J. Delany, P. Cunningham, and L. Coyle, "An assessment of case-based reasoning for spam filtering", Artificial Intelligence Review Journal, Vol. 24, No. 3-4, 2018, pp. 359-378.
2. P. Cunningham, N. Nowlan, S.J. Delany, and M. Haahr, "A case-based approach in spam filtering that can track concept drift", In Proceedings: The ICCBR"03 Workshop on Long-lived CBR Systems, Trondheim, Norway, 2016
3. K. Wei, A naïve Bayes spam filter, Faculty of Computer Science, University of Berkely, 2012.
4. B. Kamens, Bayesian filtering: Beyond binary classification. Fog Creek Software, Inc., 2010.
5. M.I. Devi, R. Rajaram, and K. Selvakuberan, "Generating best features for web page classification", Webology, Vol. 5, No. 1, 2008, Article 52.
6. M. Hartley, D. Isa, V.P. Kallimani, and L.H. Lee, "A domain knowledge preserving in

Journal of Artificial Intelligence, Machine Learning and Neural Network
ISSN:2799-1172
Vol: 02, No. 04, June-July 2022
http://journal.hmjournals.com/index.php/JAIMLNN
DOI: https://doi.org/10.55529/jaimlnn.24.9.15

process engineering using self-organizing concept", In Proceedings: ICAIET 06. Sabah, Malaysia: Kota Kinabalu, 2006.
7. X. Su, A text categorization perspective for ontology mapping, Norway: Department of Computer and Information Science, Norweigian University of Science and Technology, 2002.
8. E.H. Han, G. Karypis, and V. Kumar, Text categorization using weight adjusted k-nearest neighbour classification, Department of Computer Science and Engineering, Army HPC Research Center, University of Minnesota, 1999.
9. A. McCallum, and K. Nigam, "A comparison of event models for naïve Bayes text classification", Journal of Machine Learning Research, Vol. 3, 2003, pp. 1265–1287.
10. S. Chakrabarti, S. Roy, and M.V. Soundalgekar, "Fast and accurate text classification via multiple linear discriminant projection", The VLDB Journal The International Journal on Very Large Data Bases, 2003, pp. 170–185.
11. I. Rish, An empirical study of the naive Bayes classifier, IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence. (available online: PDF, PostScript).
12. M. Mozina, J. Demsar, M. Kattan, and B. Zupan, Nomograms for Visualization of Naive Bayesian Classifier, In Proc. of PKDD-2004, pages 337-348. (available online: PDF), 2004.
13. O. R. Duda, P. E. Hart and D. G. Stork, Pattern classification (2nd edition), Section 9.6.5, p. 487-489, Wiley, ISBN 0471056693,2000.
14. J.R. Quinlan, C4.5: programs for machine learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, 1993. [12] S. Wermter, "Neural network agents for learning semantic text classification", Information Retrieval, Vol. 3, No. 2, 2004, pp. 87-103.
15. K. Nigam, J. Lafferty, and A. McCallum, "Using maximum entropy for text classification", In Proceedings: IJCAI-99 Workshop on Machine Learning for Information Filtering, pp. 61–67, 1999.
16. T. Joachims, "Text categorization with support vector machines: Learning with many relevant features", In Proceedings: Machine Learning: ECML-98, 10th European Conference on Machine Learning, pp. 137–142, 1998.
17. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I.H. Witten, "The WEKA data mining software: an update", SIGKDD Explorations, Vol. 11, No. 1, 2009, pp. 10-18
18. Richard Power. 1999 CSI/FBI computer crime and security survey. Computer Security Journal, Volume XV (2), 1999.

