SMS Spam filtering using Machine Learning and Natural Language Processing

Rikinkumar Koringa, *Prof. Pankaj Chandre*
MIT ADT University, Pune, India

***Abstract*—As more and more people use cell phones all over the world, spam through text messages has become a big problem, affecting people's privacy and safety. Even though most folks think that texts are safe and reliable, the truth is that the amount of spam texts is going up every year. This shows how important it is to find ways to stop spam and make sure people have a good experience on their phones. Bad guys and spam senders find weak spots in cell phones and use text messages as a way to break in and do harmful stuff. A common trick is to send texts with links that shouldn't be clicked on. Clicking these links can let attackers get into someone's phone from far away. This is dangerous because it can lead to stolen personal info, losing money, stolen identities, and harmful software getting onto the phone. To deal with this urgent problem, smart systems have been made. They can tell the difference between normal texts and spam. These systems use smart learning, understanding of language, and spotting of patterns to look closely at what the message says, who sent it, and other clues. By finding spam signs, these systems can warn us about risky texts, helping us decide how to handle them. But, this is a constant chase. As defenders get better at catching spam, the spammers get trickier, finding new ways to sneak past defenses. So, the tools we use to spot spam need to keep getting smarter and learn from new spam tricks. On top of smart tools, teaching people about the dangers of text spam and how to avoid it is super important. Knowing not to trust every text you get, not clicking on strange links, and telling others about any weird texts can make these smart tools even better at stopping spam. This helps make our phones safer. In short, fighting text spam needs smart inventions, teaching people how to be safe, and everyone working together. By staying alert and ready to act, we can reduce the harm text spam does and keep our phone messaging safe for everybody.**

Keywords—Machine Learning, Multinomial Naive Bayes, Natural Language Processing, SMS Spam Detection, SMS Filtering.

I. INTRODUCTION

In a world where we're always online, we can talk to anyone, any time, no matter where they are. This is great, but it comes with its own big problem: unwanted messages, or spam, that we get through SMS. These spam messages cost companies a lot of money and annoy people. Because of this, making a strong system to spot spam SMS is very important. This helps figure out which messages are real (ham) and which are spam. SMS spam filters are super important for keeping our messages safe. They check what the message says to decide if it's spam or not. This is different from how we stop spam in emails, which looks at who sent the message and the message's title. To do this, SMS spam filters use special tools called NLP (natural language processing) and machine learning. They look for clues in the words to decide if a message is spam or real. Choosing the right bits of the message to look at is a key step in telling spam from real messages. This part is about picking out the important bits from the message and using them to tell if a message is spam or not. By doing this, the filter can better spot spam and real messages. It turns the messy, unorganized text into something the machine can learn from, so it can find and understand patterns. As more people use mobile phones, SMS spam is getting worse. It's a big problem for both people who get the messages and companies. For companies, spam means spending money they don't need to, working less efficiently, and possibly harming their reputation. For people, too much spam can make them trust companies less, and they might stop using their services. That's why having an effective system to find and stop SMS spam is more important than ever, to protect money and keep communication clear.The system we're talking about is made to tackle the big problem of SMS spam. It uses the latest tech and ways of doing things to get better at finding spam quickly and accurately. This system tries to take care of sorting messages into spam or not by itself, using machine learning, so it can pick out spam messages fast and rightly. Moreover, through robust feature extraction and selection techniques, the system aims to distill actionable insights from the message content, thereby improving the overall effectiveness of the classification algorithm. However, the success of the SMS spam detection system hinges on several critical factors, including the quality of the training data, the sophistication of the classification algorithm, and the efficiency of feature extraction and selection methods. Furthermore, ongoing research and development are essential to adapt the system to evolving spamming techniques and emerging threats, ensuring its continued efficacy and relevance in an ever-changing landscape of mobile communication.

II. AIMS AND OBJECTIVES

This study focuses on creating and putting to use a strong system that can tell the difference between spam text messages and real ones. It uses machine learning, especially the Multinomial Naive Bayes method. The main goal is to tackle the increasing issue of text message spam. This kind of spam can invade privacy, make communication less secure and efficient, and could be harmful. Thanks to machine learning, this study aims to build a system that's really good at spotting spam, helping to make chatting safer and more enjoyable for everyone. The goals of this study are wide-ranging and cover different parts of fighting SMS spam. First off, it wants to lay out all there is to know about SMS spam. This includes what SMS spam is, how it affects both people who use phones and businesses, why it happens, and common tricks used by spammers. By diving into these problems with SMS spam, the study shows why it's so important to find good solutions quickly.

Second, this study looks into and compares different machine learning methods for spotting spam in text messages, paying special attention to the Multinomial Naive Bayes method. By reviewing past studies and comparing them, the goal is to find out what works well and what doesn't when it comes to detecting spam in messages. Third, the project plans to build a strong system for finding spam in texts using the Multinomial Naive Bayes method. This involves gathering text messages, preparing the data, picking out important pieces of information, and training the system to learn from examples. The aim is to make the system better at telling spam from normal messages by improving its accuracy and other important measures. Fourth, the study will test how well the newly made spam-finding system works. It will do this by checking how accurate and reliable the system is with a separate set of test data and seeing how it stacks up against other ways of finding spam. With thorough tests, the goal is to show that this new method works well in real-life situations. Fifth, the research looks ahead to how spam-finding systems can be made better in the future. It suggests looking into more advanced learning methods and adding more data and features to make the systems stronger and able to handle more data. In short, this research paper aims to create and use a machine learning-based system to find spam in text messages, focusing on the Multinomial Naive Bayes method. The goals include understanding spam, checking out current ways of finding it, making a strong system for detecting it, testing how well it works, and suggesting ideas for future research. This work hopes to push forward the technology for finding spam in text messages and help keep user privacy and safety in mobile communication.

III. LITERATURE REVIEW

"SMS Spam Detection using Machine Learning Approach":

The research paper titled "SMS Spam Detection using Machine Learning Approach" authored by Abhishek Patel was published in the International Journal of Creative Research Thoughts (IJCRT). This study concentrates on employing machine learning algorithms such as Naive Bayes and Support Vector Machine for detecting SMS spam, highlighting the significance of feature selection and model optimization. The research attains notable success in achieving high accuracy rates in spam detection, especially with the Support Vector Machine. Nevertheless, limitations arise from dependence on specific datasets and the necessity for further exploration of feature engineering techniques to enhance the model's robustness and applicability.

"Spam Detection in SMS Using Machine Learning Through Text Mining":

The research paper titled "Spam Detection in SMS Using Machine Learning Through Text Mining" was published in the International Journal of Scientific & Technology Research, Volume 9, Issue 02, February 2020. This study centers on employing machine learning algorithms for detecting SMS spam.

Limitations of the study encompass the requirement for refining filtering techniques and the possibility of enhancing existing algorithms. The accomplishment of the research lies in the successful implementation of machine learning classifiers to accurately distinguish between spam and legitimate messages in SMS communication.

"Mobile SMS Spam Detection using ML Techniques":
The research paper titled "Mobile SMS Spam Detection using ML Techniques" was published in the Journal of Emerging Technologies and Innovative Research (JETIR) in December 2018, Volume 5, Issue 12. The paper presents a systematic literature review on SMS spam detection, analyzing 13 research papers in the field. It assesses the techniques, advantages, disadvantages, and challenges addressed by the proposed methods, highlighting the importance of publicly available datasets for effective spam filtering algorithms. However, the study lacks a thorough examination of challenges related to local content and the use of shortened words in SMS spam detection. Despite this limitation, the paper provides a comprehensive overview of existing SMS spam detection techniques, underscores the significance of dataset availability for developing robust spam filtering algorithms, and offers valuable insights for future research directions in mobile SMS spam detection.

"A review on social spam detection: Challenges, open issues, and future directions":
The review article titled "A review on social spam detection: Challenges, open issues, and future directions" published in the journal "Expert Systems With Applications," offers a comprehensive overview of the challenges and advancements in social spam detection. It emphasizes the importance of analyzing spammers' behavior to enhance security in online social network environments. While the review primarily focuses on machine learning and deep learning-based techniques, it could benefit from further exploration of real-world case studies and practical implementations to validate detection methods. Overall, the review provides valuable insights into the complexities of social spam detection, highlighting key challenges, open issues, and future research directions in combating spam on online social networks.

“Deep learning to filter SMS Spam”:
The research paper titled "Deep learning to filter SMS Spam" was published in the journal "Future Generation Computer Systems" by P.K. Roy, J.P. Singh, and S. Banerjee. The researchers introduced a deep learning framework utilizing Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) models to classify SMS spam messages with an impressive accuracy of 99.44%. Their study surpassed traditional machine learning classifiers, highlighting the potential of advanced neural network architectures in improving SMS spam filtering. Limitations include the focus solely on English text messages and the computational complexity associated with deep learning models. However, noteworthy achievements include the high accuracy rate achieved and the demonstration of the superiority of deep learning techniques in SMS spam classification.

“SMS spam filtering: Methods and data”:
The research titled "SMS spam filtering - Methods and data" was published in the journal Expert Systems with Applications. The paper addresses the challenges and potential remedies in SMS spam filtering, emphasizing the absence of consensus regarding the most effective techniques due to the field's early stage of development. While content-based solutions exhibit potential for accuracy and adaptability, the research encounters difficulties in comparing techniques across diverse datasets concerning language, size, and collection methods. Despite these challenges, the paper underscores the potential for progress through the integration of various technologies and solutions in the continuous battle against SMS spam.

“A weighted feature enhanced Hidden Markov Model for spam SMS filtering”:
The research paper titled "A weighted feature enhanced Hidden Markov Model for spam SMS filtering" was published in the journal Neurocomputing. This study introduces a novel method to enhance spam SMS filtering by integrating weighted features into a Hidden Markov Model. The proposed model aims to overcome the limitations of traditional rule-based systems and content-based

filtering techniques by improving accuracy and expediting training and detection processes. However, the study acknowledges the necessity for further investigation into weighting untrained words in the testing set and suggests exploring optimization techniques such as particle filter and deep reinforcement learning to enhance classification performance.

Paper Name	Advantage	Limitation
"SMS Spam Detection using Machine Learning Approach"	Is really good at spotting spam messages with high success . Highlights significance of feature selection and model optimization	Reliance on certain sets of data. We need to dig deeper into ways to build and improve our model. By exploring how we can change and create new features, we can make our model stronger and useful for more situations.
"Spam Detection in SMS Using Machine Learning Through Text Mining"	Machine learning classifiers are successfully implemented for precise spam detection Using text mining techniques to spot spam in text messages effectively.	Improving how we filter things is needed. There's a chance to make current algorithms better.
"Mobile SMS Spam Detection using ML Techniques"	Detailed overview of the state-of-the-art SMS spam detection methods, helping researchers to comprehend the topic. Emphasis on the importance of publicly available datasets for developing robust spam filtering algorithms	Lack of thorough examination of challenges related to local content and the use of shortened words in SMS spam detection
"A review on social spam detection: Challenges, open issues, and future directions"	Offers a comprehensive overview of challenges and advancements in social spam detection, providing valuable insights for researchers Emphasizes the importance of analyzing spammers' behavior to enhance security in online social network environments	Could benefit from further exploration of real-world case studies and practical implementations to validate detection methods
“Deep learning to filter SMS Spam”	Uses a smart blend of CNN and LSTM techniques to sort out spam texts, hitting an amazing success rate of 99.44%. Demonstrates the potential of advanced neural network architectures to surpass traditional machine learning classifiers, improving SMS spam filtering.	Focuses solely on English text messages, potentially limiting applicability to other languages or multilingual contexts. Involves computational complexity associated with deep learning models, potentially requiring significant computational resources.
“SMS spam filtering: Methods and data”	Content-based solutions show potential for high accuracy and adaptability in filtering SMS spam. Integration of various technologies and solutions presents a pathway for progress against SMS spam.	Lack of consensus on the most effective techniques due to the nascent stage of the field. Difficulty in comparing techniques due to diverse datasets (varying in language, size, and collection methods).
“A weighted feature	Integrates weighted features	Necessity for further

enhanced Hidden Markov Model for spam SMS filtering”	into a Hidden Markov Model to improve accuracy in spam SMS filtering. Aims to expedite training and detection processes, overcoming limitations of traditional systems.	research into weighting untrained words in the testing set. Suggests the need to explore optimization techniques (like particle filter and deep reinforcement learning) for enhanced classification performance.

IV. PROPOSED ARCHITECTURE

The goal of the study is to create a system that can tell the difference between spam and real ("ham") text messages, using the Python language, the study of how computers can understand human language (NLP), and computer algorithms that improve through experience. The work will be done on Google Colab, a place on the internet where people can work together and use cloud computing. First, we need to gather a lot of text messages, both spam and non-spam. This collection is key for teaching and testing our computer model. We'll make a special tool to clean and get the text ready. This includes making the text uniform, breaking it down into smaller parts, removing filler words, and finding patterns with regular expressions to make sure the text is good and consistent. After preparing the text, we will use techniques to pull out important features from the text. Using NLP methods like bag-of-words and TF-IDF, which helps us understand how often words appear and their importance, we transform the text into numbers that computer algorithms can work with. We'll also use the Natural Language Toolkit (NLTK) for more text analysis, including breaking the text into smaller parts and identifying the type of words. The main part of the study is to teach and test the spam detection model using a technique called multinomial naïve Bayes. This technique is good for sorting text and will learn what makes a message spam or not spam. We'll check how good the model is with measures like how accurate it is and how well it finds spam, using various tests to make sure it works well in different situations. We'll also use NLP and regular expressions throughout the study to better understand the text and pick out features that indicate spam. Regular expressions help us find specific patterns in spam messages, adding to the information we get from the text. Google Colab is our chosen tool for this project because it lets us work together in the cloud. It makes it easy to write code, try things out, and teach our model, plus we can easily share what we've found thanks to its connection to Google Drive. After we train the model, we'll check how well it works by using a separate set of test messages. We want to see if it can correctly identify spam and non-spam texts. We will also compare it to other methods to see how good our model is. Our plan focuses on a way of finding and figuring out spam texts with a special method called multinomial naïve Bayes. This method is really good for sorting texts and will learn from a bunch of texts we give it to spot patterns that tell us if a message is spam or not. Multinomial naïve Bayes is like a probability guesser based on a math rule, thinking each piece of a message (like words) doesn’t affect the others. When it looks at texts for spam, it guesses if a message is spam or not by looking at the words it has. It checks how often a word shows up in spam and not spam texts, then uses this info to guess. We'll check how good our method works with tests like accuracy, being precise, recall, and the F1 score. We’ll use cross-validation, which helps make sure our spam detector works well on new texts. By using this multinomial naïve Bayes method, our system gets a simple but powerful way to sort texts. It's good at dealing with a lot of data and is fast and efficient, perfect for quickly sorting through texts to find spam.

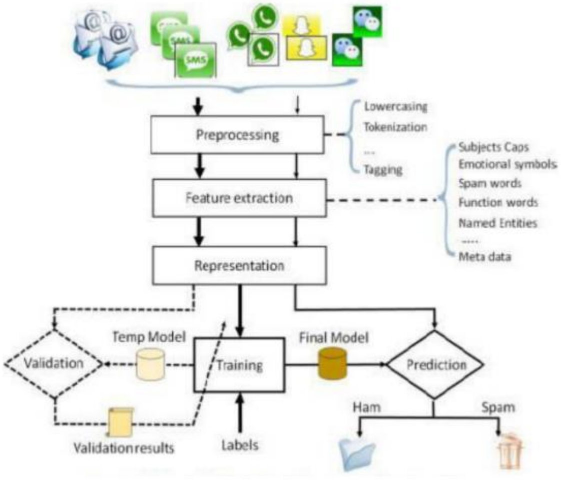


Fig 1. Flow chart of Spam Detection model

A. *Dataset*

The dataset is obtained from online resources like Kaggle which consists of a large number of sms messages to be trained and tested.

B. *Material/Tools Required*

- Python
- Natural Language Processing
- NLTK
- Regular Expression
- Scikit learn
- Google Colab

V. ALGORITHM

Multinomial Naive Bayes Algorithm The multinomial naive Bayes algorithm is a tool often used to figure out if a piece of text, like an email, is spam or not. It uses a thing called Bayes' theorem with the idea that each word in the text is separate and doesn't affect the other words. This is really good for looking at texts with lots of different words and seeing how often each word shows up. When it comes to figuring out if a text message is spam, this algorithm looks at each word in the message and decides how likely it is that the message is spam based on those words. It does this by learning from examples - it looks at lots of spam and non-spam messages it has seen before and uses this to make a good guess. Even though it's pretty simple, this algorithm is really good at sorting text into categories. It's able to deal with texts that have lots of different words without getting bogged down. And because it's based on probability, it can learn from new messages and get better over time at spotting spam. What's also great about the multinomial naive Bayes algorithm is that it works fast, making it perfect for real-time jobs like spotting spam texts as soon as they come in. This means we can catch spam quickly and keep our inboxes clean. All in all, the multinomial naive Bayes algorithm is a straightforward but powerful way to tell if a text message is spam or not. It's fast, gets better with more data, and is a top choice for keeping spam out of our messages.

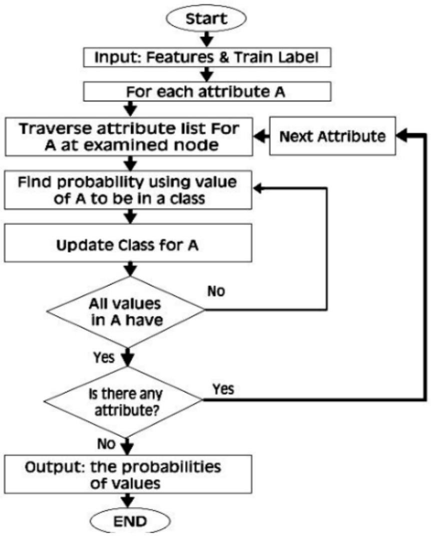

Start
Input: Features & Train Label
For each attribute A
Traverse attribute list For A at examined node
Next Attribute
Find probability using value of A to be in a class
Update Class for A
All values in A have
No
Yes
Is there any attribute?
Yes
No
Output: the probabilities of values
END

Fig 2. Flow Chart of multinomial naive bayes

A. *Equations*

P(A|B) = P(A) * P(B|A)/P(B)

Where we are calculating the probability of class A when predictor B is already provided.

P(B) = prior probability of B

P(A) = prior probability of class A

P(B|A) = occurrence of predictor B given class A probability

VI. IMPLEMENTATION

Getting Ready with Our Tools: First off, we gather all the tools we'll need for our job. This includes grabbing pandas to work with data, numpy to handle numbers, nltk for dealing with text, and scikit-learn for all our machine learning needs and to check how well our project is doing.

Setting Up NLTK: Next, we make sure all the important stuff from NLTK is ready to go. NLTK has a bunch of helpers like stopwords and tokenizers that we need to get our text ready for the project.

Loading in the SMS Messages: We grab our SMS messages from a CSV file. It's usually set up with one column for the text of the messages and another column to tell us if each message is spam or just a regular message (also called 'ham').

Making SMS Messages Ready: We have a step where we clean up the SMS messages to make them easier for our project to understand. We call this step preprocess_text(). It turns everything into lowercase to keep things consistent, splits the text into individual words, takes out common but not very meaningful words known as stopwords, and gets rid of punctuation.

Dividing Our Data: Once our messages are prepped and ready, we split them into two sets with the help of train_test_split() from scikit-learn. This way, we have one set for teaching our project and another to see how well it's learned.

Turning Text into Numbers: Before our messages can go through machine learning, they need to be turned into numbers. We do this by counting how many times each word appears using a technique called Bag-of-Words. With the CountVectorizer from scikit-learn, we change our cleaned-up text into a big chart that shows the count of each word across all messages.

Training a Spam Filter with Multinomial Naïve Bayes: We use the Multinomial Naïve Bayes method, a popular choice for sorting texts, to teach our model how to spot spam messages. This method is great for learning how to tell spam texts from regular ones because it's straightforward and works well. We create a spam filter with the help of a tool from scikit-learn called MultinomialNB(). We train it with examples of spam and non-spam messages.

Guessing Which Texts are Spam: After training, our spam filter is ready to guess which messages are spam. It looks at new messages and uses what it learned from the examples to decide if each one is spam or not.

Assessing the Performance of the Model:

Lastly, we compare our model's predictions to the genuine labels in the test set to see how well it performed. Text categorization models are evaluated using popular measures such as accuracy, precision, recall, and F1-score. We utilize scikit-learn routines to calculate these metrics, which we then display to assess the efficacy of our SMS spam filtering model.

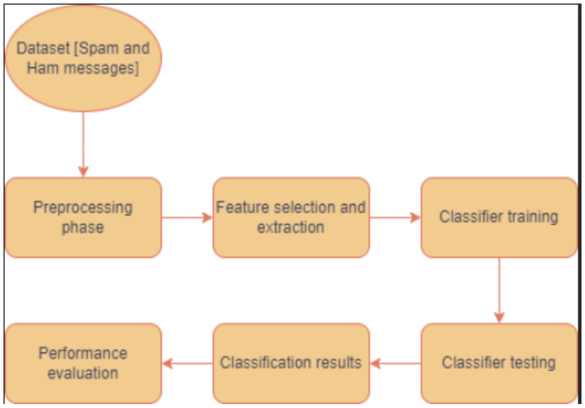
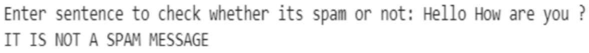


Fig 3. System Architecture

VII.TESTING AND RESULTS

We tested all the functionalities of the project and ensured that it would not give any uncertainty regarding the functionalities, working and output of our project models.

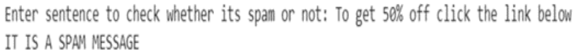
A. HAM Message:



```
Enter sentence to check whether its spam or not: Hello How are you ?
IT IS NOT A SPAM MESSAGE
```

Fig 4 . HAM Message Example

B. *SPAM Message*



```
Enter sentence to check whether its spam or not: To get 50% off click the link below
IT IS A SPAM MESSAGE
```

Fig 5 . SPAM Message Example

C. Precision score: 95.62%

D. Accuracy: 98.74%

E. Confusion Matrix: [[948   7]
[7 153]]

VIII.CONCLUSION

The growth of spam communications is a serious problem in the age of digital communication, endangering the effectiveness and security of communication channels. The importance of machine learning algorithms—in particular, supervised learning techniques—in the creation of efficient spam detection apps has been highlighted by this research. It has been shown that supervised machine learning can effectively categorise messages into their appropriate categories by utilising algorithms like Multinomial Naive Bayes (MNB). This makes it possible to identify and filter spam communications with a high degree of accuracy.

Over the course of this study, it has become clear that supervised machine learning algorithms like MNB perform exceptionally well at sorting messages and correctly categorising them. These algorithms make judgements about the type of incoming communications based on patterns and connections between features that they have learned from labelled training data. Robust and effective spam detection systems can be developed thanks to supervised learning techniques, which score the model and weigh its predictions.

This study's use of the Multinomial Naive Bayes method has shown to be especially successful in determining whether or not a message is spam. This algorithm has shown encouraging results in accurately differentiating between spam and valid messages. It is well-known for its simplicity and efficacy in text classification tasks. Effective spam filtering is made possible by MNB's ability to generate probabilistic predictions about the chance that a message is spam by examining the frequency distribution of words and other properties in the text data.

Even though this research has shown promising results, it is important to recognise that spam identification is still an area that requires development. The efficiency and effectiveness of spam detection algorithms must be continuously improved as communication technology advances and spammers employ ever-more-advanced strategies. In order to further increase the precision and dependability of spam detection systems, future research should concentrate on investigating sophisticated machine learning approaches that make use of deep learning algorithms and neural networks.

Furthermore, improvements in feature engineering, model optimisation, and data pretreatment methods can aid in the creation of spam detection algorithms that are more effective. Through the refinement of filtering techniques and the investigation of new feature extraction procedures, researchers can augment the scalability and resilience of spam detection systems, guaranteeing their efficacy in practical contexts.

In summary, our study has shown how crucial machine learning algorithms—especially supervised learning methods—are to the creation of spam detection software. High classification accuracy rates for spam communications have been attained by utilising techniques like Multinomial Naive Bayes. Nonetheless, in order to handle the dynamic character of spam and guarantee the effectiveness and dependability of spam detection systems going forward, constant advancements and innovations are required.

VIII.ACKNOWLEDGEMENT

A special thanks to our Guide Prof. Pankaj Chandare sir who helped me in stimulating suggestions and encouraging in supervising our project and special thanks to Prof Shafi Pathan sir who also stood by my side and guided me wherever necessary. I also extend our gratitude to our faculties and well-wishers who supported us morally. We will forever remain grateful to all of you.

REFERENCE

[1] Nuruzzaman, M. T., Lee, C., Abdullah, M. F. A. B., & Cho, D. (2012). Simple SMS spam filtering on independent mobile phone. Security and Communication Networks, 5, 1209-1220 was retrived from https://onlinelibrary.wiley.com/doi/epdf/10.1002/sec.577

[2] Chandre, P. R., Mahalle, P. N., & Shinde, G. R. (2018). Machine learning based novel approach for intrusion detection and prevention system: A tool based verification. In Proceedings of the 2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN). IEEE. https://ieeexplore.ieee.org/abstract/document/8668618

[3] Almeida, T. A., Hidalgo, J. M. G., & Yamakami, A. (2011). Contributions to the study of SMS spam filtering: new collection and results. ACM Digital Library. Retrieved from https://dl.acm.org/doi/10.1145/2034691.2034742

[4] Sravya, G. S., Pradeepini, G., & Vaddeswaram, Guntur. (2020). Mobile Sms Spam Filter Techniques Using Machine Learning Techniques. International Journal of Scientific & Technology Research, 9(03), ISSN 2277-8616 retrived from https://www.ijstr.org/final-print/mar2020/Mobile-Sms-Spam-Filter-Techniques-Using-Machine-Learning-Techniques.pdf

[5] Vishwakarma, A. K., Ansari, M. D., & Rai, G. (2020). SMS Spam Filtering Using Machine Learning Technique. In Proceedings of the International Conference on Computational and Communication Engineering 2020 (ICCCE 2020), Springer Nature Singapore retrieved from https://www.springerprofessional.de/en/sms-spam-filtering-using-machine-learning-technique/18471878

[6] YouTube. Simple SMS Spam Filter with Python - Step by Step Tutorial. Retrieved from https://youtu.be/VDg8fCW8LdM?si=nVqYH-Pi2qdLAvCg

[7] YouTube. Implementing a Spam classifier in python | Natural Language Processing. Retrieved from https://youtu.be/fA5TSFELkC0?si=gYZuNPzLodkas0Tr

[8] Abhishek Pate, Priya Jhariya, SudalaguntaBharath, Ankita wadhawan. "SMS Spam Detection using Machine Learning Approach." International Journal of Creative Research Thoughts (IJCRT), Volume 9, Issue 4 April 2021. Retrieved from https://ijcrt.org/papers/IJCRT2104653.pdf

[9] International Journal of Scientific & Technology Research. (February 2020). "Spam Detection in SMS Using Machine Learning Through Text Mining." Volume 9, Issue 02. Retrieved from https://www.ijstr.org/final-print/feb2020/Spam-Detection-In-Sms-Using-Machine-Learning-Through-Text-Mining.pdf

[10] Deshmukh, S. S., & Chandre, P. R. (2014). Survey on: Naive Bayesian and AOCR Based Image and Text Spam Mail Filtering System. International Journal of Emerging Technology and Advanced Engineering, 4(4), [Page Range]. Retrieved from https://www.ijetae.com/files/Volume4Issue4/IJETAE_0414_139.pdf

[11] Rao, S., Verma, A. K., & Bhatia, T. (2021). A review on social spam detection: Challenges, open issues, and future directions. Expert Systems With Applications, 186, 115742. Retrieved from https://www.sciencedirect.com/science/article/abs/pii/S0957417421011209

[12] Roy, P.K., Singh, J.P., & Banerjee, S. (2020). Deep learning to filter SMS Spam. Future Generation Computer Systems, 102, 524–533. Retrieved from https://www.sciencedirect.com/science/article/abs/pii/S0167739X19306879

[13] Delany, S. J., Buckley, M., & Greene, D. (2012). SMS spam filtering - Methods and data. Expert Systems with Applications, 39, 9899–9908. Retrieved from https://www.sciencedirect.com/science/article/abs/pii/S0957417412002977

[14] Xia, T., & Chen, X. (2021). A weighted feature enhanced Hidden Markov Model for spam SMS filtering. Neurocomputing, 444, 48–58. https://www.sciencedirect.com/science/article/abs/pii/S0925231221003313

[15] Abdulhamid, S. M., Abd Latiff, M. S., Chiroma, H., Osho, O., Abdul-Salaam, G., Abubakar, A. I., & Herawan, T. (2017). A Review on Mobile SMS Spam Filtering Techniques. IEEE Access, 5, 20120-20133 and it was retrieved from https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7851079

[16] Chandre, P. R., Mahalle, P. N., & Shinde, G. R. (2021). Intrusion prevention framework for WSN using deep CNN. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(6), 2033-2041. Retrieved from https://turcomat.org/index.php/turkbilmat/article/view/7145

[17] Samadhan Nagre. "Mobile SMS Spam Detection using ML Techniques." Journal of Emerging Technologies and Innovative Research (JETIR), Volume 5, Issue 12, December 2018. Retrieved from https://www.jetir.org/papers/JETIR1812A76.pdf

[18] Ashfaq, S., Chandre, P., Pathan, S., Mande, U., Nimbalkar, M., & Mahalle, P. (2023). Defending Against Vishing Attacks: A Comprehensive Review for Prevention and Mitigation Techniques. In Proceedings of the International Conference on Recent Developments in Cyber Security. Springer, Singapore. https://link.springer.com/chapter/10.1007/978-981-99-9811-1_3