Spam Mail Detection using Bayesian Networks

B Sai Abhishek
Artificial Intelligence,
Amrita School of Engineering,
Banglore,India
BL.EN.U4AIE21015@bl.students.amrita.edu

Balam Ruchith balaji
Artificial Intelligence,
Amrita School of Engineering,
Banglore,India
BL.EN.U4AIE21017@bl.students.amrita.edu

Chillakuru Hari
Artificial Intelligence,
Amrita School of Engineering,
Banglore,India
BL.EN.U4AIE21038@bl.students.amrita.edu

***Abstract*— In the ever-evolving landscape of electronic communication, email remains a primary means of information exchange. However, the proliferation of spam emails poses a significant threat to users' privacy, productivity, and overall online experience. This paper presents a sophisticated approach to spam mail detection utilizing Bayesian networks, aiming to enhance email security by accurately identifying and filtering out unwanted or malicious content.The proposed method leverages the probabilistic nature of Bayesian networks to model the relationships between various features commonly found in email datasets. Features include sender information, email content, header details, and other relevant attributes. By incorporating the probabilistic dependencies between these features, the Bayesian network adapts dynamically to changing patterns of spam, demonstrating a high level of adaptability and resilience against emerging threats.**

***Keywords*— Spam mail detection, Bayesian networks, Email security, Probabilistic modeling, Feature extraction, Conditional probabilities.**

I. INTRODUCTION

In an era dominated by digital communication, email has become an integral part of our daily lives, facilitating seamless information exchange and connectivity. However, this widespread reliance on email has also given rise to the persistent challenge of spam, an unwelcome intrusion that inundates our inboxes with unsolicited and often malicious content. To combat this menace and ensure the integrity of our digital communication channels, the quest for efficient and accurate spam mail detection mechanisms has become paramount.

One promising approach in the realm of spam mail detection is the utilization of Bayesian networks. Bayesian networks, rooted in probabilistic reasoning and statistical modeling, offer a sophisticated and adaptable framework for analyzing the intricate relationships between various features within an email and predicting the likelihood of it being spam. By leveraging the inherent structure of Bayesian networks, which represents dependencies among variables through a graphical model, we can harness the power of probabilistic inference to make informed decisions about the nature of incoming emails.

This paper delves into the conceptual underpinnings of Bayesian networks and explores their application in the realm of spam mail detection. We will discuss how these networks can effectively capture the probabilistic relationships between different email attributes, such as sender address, content, and metadata, allowing for a holistic assessment of the email's legitimacy. Furthermore, we will explore the advantages of Bayesian networks over traditional rule-based or machine learning approaches, emphasizing their ability to handle uncertainty and adapt to evolving spam tactics.

As we embark on this exploration of Bayesian networks in spam mail detection, we aim to shed light on their potential to enhance the precision and recall of spam filtering systems. By understanding the intricacies of Bayesian networks and their application in this context, we can pave the way for more robust and intelligent defenses against the ever-evolving landscape of spam emails, ultimately ensuring a safer and more secure digital communication environment.

II. RELATED WORKS

Ahmed, N., Amin, R., Aldabbas, H., Koundal, D., Alouffi, B. and Shah, T **[1]** proposed methodology not only handles high imbalances in data but also improves the performance of the classification model. It appropriately manages data imbalance without leading to oversampling or undersampling of data and appropriately assigns weightage to minority classes. The Dual-Layer architecture achieved a minimum F1-score of 97.6%, which is greater than the maximum F1-scores of 97%, 95.1%, and 95.3% obtained for spam email classification using traditional machine learning and deep learning techniques, with or without resampling. The maximum F1-score obtained for classifying spam emails is 99% achieved by the Dual Layer ANN classifier. Compared to earlier research studies, the proposed dual-layer architecture yielded improved performance metrics for all individual evaluation factors. This architecture is flexible and may be reproduced and employed for a variety of other classification applications where severe class imbalance is observed. Further research can be conducted in different areas, such as applying the proposed method to other complex text classification tasks and analyzing the effect of different feature extraction, selection, and preprocessing techniques on the proposed method.

Doshi, J., Parmar, K., Sanghavi, R. and Shekokar, N **[2]** has discussed about in the last two decades, spam detection and filtration gained the attention of a sizeable research community. (e reason for a lot of research in this area is its costly and massive effect in many situations like consumer behavior and fake reviews. (e survey covers various machine learning techniques and models that the various researchers have proposed to detect and filter spam in emails and IoT platforms. (e study categorized them as supervised, unsupervised, reinforcement learning, etc. (e study compares these approaches and provides a summary of learned lessons from each category. (is study concludes that most of the proposed email and IoT spam detection methods are based on supervised machine learning techniques. A labeled dataset for the supervised model training is a crucial and time-consuming task. Supervised learning algorithms SVM and Na¨ıve Bayes outperform other models in spam detection. (e study provides comprehensive insights of these algorithms and some future research directions for email spam detection and filtering.

Guo, Y., Mustafaoglu, Z. and Koundal, D **[3]**, an efficient spam detection model was proposed based on a BERT model and supervised learning classifier to detect spam emails. Email texts were represented via the features obtained from the BERT outputs, and classifier algorithms in machine learning were employed to classify the feature vectors into ham or spam categories. The experimental results demonstrate that the logistic regression algorithm achieved the best classification performance in two publicly available datasets. To sum up, there is a promotion to use the BERT model and classifier in spam detection. This study can be extended to various applications, e.g., spam messages detection in a mobile system and fake news detection in social media platforms. This study demonstrates the high ability of the BERT model to interpret text and provides salient features for future processing. Further research in combing more comprehensive layers inside the BERT is encouraged to further validate the proposed framework.

Jáñez-Martino, F., Alaiz-Rodríguez, R., González-Castro, V., Fidalgo, E. and Alegre, E **[4]** presents a review on spam email detection, focusing on the analysis of spammer strategies and the changing nature of the data in this feld. The spammer, or the adversarial fgure in this environment, follows sophisticated strategies to bypass the flters. In our study on spam datasets over the last few decades, we identifed the use of poisoning text, obfuscated words, hidden text salting and image-based spam as the most popular spammer tricks. More recent strategies include multi-language emails to poison the text, attachments like PDF or ".docx" fles to introduce a spam message, and URLs to connect to potentially harmful sites. We also reviewed the studies involved in detecting these tricks and minimising their efect. We explored the most recent works on fltering spam emails emphasising on the phases of feature extraction and feature selection to mitigate overfting by reducing the dimensionality of the input space. Despite the rise of deep learning in other application felds, traditional machine learning algorithms are still the most popular approaches in the literature for the spam email fltering. Their high performance as well as their simplicity when compared with deep learning models may be the reasons.

Shitanshu Jain, S.C.Jain, Santosh Vishwakarma create classification models for every KNN and Nave Bayes classification methods, as well as a performance table that compares the performance of different methodologies classifiers The proposed approach is one of the best classification methods employing similarity measures, according to the comparison of different measures with KNN classifier methods discussed previously. In compared to other classifiers, K-NN has attained the best overall performance. The idea of this research is to present a variety of various Term Weighting Schemes for Text-Classification. It has been discovered that a new approach of building classifiers may be developed, providing KNN Classification techniques to be applied to classification problems and assisting in the solving of a number of significant issues with existing classification systems. When compared to traditional text categorization methods, the proposed method has the highest accuracy.

Zhou, X., Kan, Z., Meng, H. and Li, Y.,[6] presents a comprehensive noise reduction and correction algorithm for trenching depth data in precision agriculture. It addresses challenges such as large fluctuations and low precision in measured trenching depth data. The algorithm combines wavelet denoising and Kalman filtering to improve data quality. The study evaluates the algorithm's performance using various metrics and explores the impact of different Q/R values. Results show that the proposed algorithm effectively reduces noise and fluctuations, leading to a more accurate representation of trenching depth. In conclusion, the algorithm significantly enhances the quality of trenching depth data, making it suitable for precision agriculture applications and providing valuable insights for advanced monitoring and control systems in agricultural machinery operations.

Hoang Vuong, Daniel Jilani, Samir Malhotra, Michael P.H. Lau, [7] highlights the strengths and limitations of various ECG denoising methods, emphasizing the need for effective solutions, especially in the context of wearables and healthcare IoT.

Jun Shi, Gong Chen, Yanan Zhao,[9] discusses the challenges associated with analyzing real-world signals that are inherently non-stationary, such as those found in human speech, animal sounds, mechanical vibrations, seismic waves, radar, sonar, and biomedical signals. These signals often contain multiple components with time-varying spectral features. The traditional approach to analyzing such signals involves using time-frequency representations (TFRs). The short-time Fourier transform (STFT) and

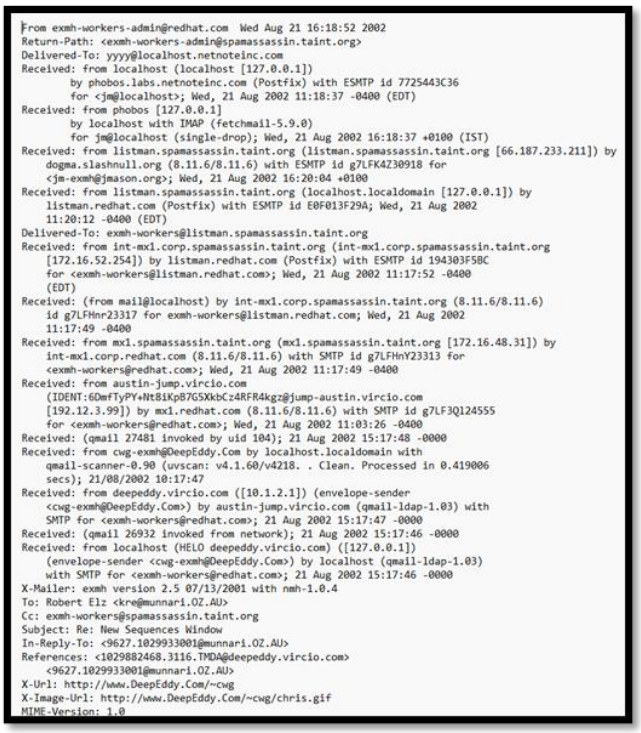
wavelet transform (WT) are two commonly used TFR methods.

Arman Kheirati Roonizi and Ivan W. Selesnick,[10] introduces a novel Kalman filter framework for signal denoising that integrates conventional linear time-invariant (LTI) filtering with total variation (TV) denoising. The proposed approach models the desired signal as a combination of two distinct components: a band-limited signal (e.g., low-frequency component) and a sparse-derivative signal (e.g., high-frequency component). The framework employs an iterative Kalman filter/smoother methodology, utilizing zero-phase LTI filtering to estimate the band-limited signal and TV denoising to estimate the sparse-derivative signal. By simultaneously considering these components, the approach aims to achieve improved denoising performance for signals with both low and high-frequency components.

III. METHODOLOGY

A. *Data Set Description:*

- Data set format : TEXT
- Data set Composition:
- Train – 2000 text emails
- Test- 800 text email
- Each mail has receiver's Information and Senders information.
- And also it has time, date information also.



```
From exmh-workers-admin@redhat.com  Wed Aug 21 16:18:52 2002
Return-Path: <exmh-workers-admin@spamassassin.taint.org>
Delivered-To: yyyy@localhost.netnoteinc.com
Received: from localhost (localhost [127.0.0.1])
        by phobos.labs.netnoteinc.com (Postfix) with ESMTP id 7725443C36
        for <jm@localhost>; Wed, 21 Aug 2002 11:18:37 -0400 (EDT)
Received: from phobos [127.0.0.1]
        by localhost with IMAP (fetchmail-5.9.0)
        for jm@localhost (single-drop); Wed, 21 Aug 2002 16:18:37 +0100 (IST)
Received: from listman.spamassassin.taint.org (listman.spamassassin.taint.org [66.187.233.211]) by
    dogma.slashnull.org (8.11.6/8.11.6) with ESMTP id g7LFK4Z30918 for
    <jm-exmh@jmason.org>; Wed, 21 Aug 2002 16:20:04 +0100
Received: from listman.spamassassin.taint.org (localhost.localdomain [127.0.0.1]) by
    listman.redhat.com (Postfix) with ESMTP id E0F013F29A; Wed, 21 Aug 2002
    11:20:12 -0400 (EDT)
Delivered-To: exmh-workers@listman.spamassassin.taint.org
Received: from int-mx1.corp.spamassassin.taint.org (int-mx1.corp.spamassassin.taint.org
    [172.16.52.254]) by listman.redhat.com (Postfix) with ESMTP id 194303F5BC
    for <exmh-workers@listman.redhat.com>; Wed, 21 Aug 2002 11:17:52 -0400
    (EDT)
Received: (from mail@localhost) by int-mx1.corp.spamassassin.taint.org (8.11.6/8.11.6)
    id g7LFHnr23317 for exmh-workers@listman.redhat.com; Wed, 21 Aug 2002
    11:17:49 -0400
Received: from mx1.spamassassin.taint.org (mx1.spamassassin.taint.org [172.16.48.31]) by
    int-mx1.corp.redhat.com (8.11.6/8.11.6) with SMTP id g7LFHnY23313 for
    <exmh-workers@redhat.com>; Wed, 21 Aug 2002 11:17:49 -0400
Received: from austin-jump.virtio.com
    (IDENT:6DmfTyPY+Nt8iKpB7G5XkbCz4RFR4kgz@jump-austin.virtio.com
    [192.12.3.99]) by mx1.redhat.com (8.11.6/8.11.6) with SMTP id g7LF3Q124555
    for <exmh-workers@redhat.com>; Wed, 21 Aug 2002 11:03:26 -0400
Received: (qmail 27481 invoked by uid 104); 21 Aug 2002 15:17:48 -0000
Received: from cwg-exmh@DeepEddy.Com by localhost.localdomain with
    qmail-scanner-0.90 (uvscan: v4.1.60/v4218. . Clean. Processed in 0.419006
    secs); 21/08/2002 10:17:47
Received: from deepeddy.virtio.com ([10.1.2.1]) (envelope-sender
    <cwg-exmh@DeepEddy.Com>) by austin-jump.virtio.com (qmail-ldap-1.03) with
    SMTP for <exmh-workers@redhat.com>; 21 Aug 2002 15:17:47 -0000
Received: (qmail 26932 invoked from network); 21 Aug 2002 15:17:46 -0000
Received: from localhost (HELO deepeddy.virtio.com) ([127.0.0.1])
    (envelope-sender <cwg-exmh@DeepEddy.Com>) by localhost (qmail-ldap-1.03)
    with SMTP for <exmh-workers@redhat.com>; 21 Aug 2002 15:17:46 -0000
X-Mailer: exmh version 2.5 07/13/2001 with nmh-1.0.4
To: Robert Elz <kre@munnari.OZ.AU>
Cc: exmh-workers@spamassassin.taint.org
Subject: Re: New Sequences Window
In-Reply-To: <9627.1029933001@munnari.OZ.AU>
References: <1029882468.3116.TMDA@deepeddy.virtio.com>
    <9627.1029933001@munnari.OZ.AU>
X-Url: http://www.DeepEddy.Com/~cwg
X-Image-Url: http://www.DeepEddy.Com/~cwg/chris.gif
MIME-Version: 1.0
```

B. BAYESIAN NETWORKS

A Bayesian network is a graphical model that represents probabilistic relationships among a set of variables. The relationships are depicted as a directed acyclic graph (DAG), where nodes represent variables, and edges represent probabilistic dependencies between the variables.

Components:

- **Nodes** (Vertices): Represent random variables.
- **Edges** (Directed Arrows): Indicate probabilistic dependencies between variables.
- **Conditional Probability Tables (CPTs)**: Assign probabilities to each variable given the values of its parents..

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

Where:

- P(A|B) is the posterior probability of event A given evidence B.
- P(B|A) is the likelihood of evidence B given that A is true.
- P(A) is the prior probability of event A.
- P(B) is the probability of evidence B.

Bayesian Networks Formula:

The joint probability distribution of a Bayesian network can be factorized using the chain rule of probability:

$$P(X_1, X_2, \ldots, X_n) = P(X_1) \cdot P(X_2|X_1) \cdot P(X_3|X_1, X_2) \cdot \ldots \cdot P(X_n|X_1, X_2, \ldots, X_{n-1})$$

Inference in Bayesian Networks:

Bayesian networks allow us to perform inference, which involves updating beliefs about certain variables given observed evidence. This is done using the following formula:

$$P(X|E) = \frac{P(E|X) \cdot P(X)}{P(E)}$$

Where:

- P(X | E) is the posterior probability of variable X given evidence E.

- P(E | X) is the likelihood of evidence E given that X is true.
- P(X) is the prior probability of variable X.
- P(E) is the probability of evidence E.

Bayesian networks can be learned from data, a process known as structure learning. This involves identifying the relationships between variables and determining the structure of the graph. Parameter learning involves estimating the probabilities associated with each node in the network. Learning algorithms such as the Expectation-Maximization (EM) algorithm and the Hill-Climbing algorithm are commonly used for this purpose.

Various algorithms are employed for performing inference in Bayesian networks. The most common one is the Variable Elimination algorithm, but others, like the Junction Tree algorithm or the Gibbs Sampling algorithm, are also used. These algorithms enable the computation of probabilities and predictions based on the observed evidence in the network.

Bayesian networks find extensive applications in healthcare, ranging from disease diagnosis to treatment planning. They can model complex relationships between symptoms, medical history, and test results, providing a probabilistic framework for medical decision support systems. As technology continues to advance, Bayesian networks will likely play an increasingly important role in addressing complex and uncertain scenarios in diverse fields.

IV. EXPERIMENTAL SETUP

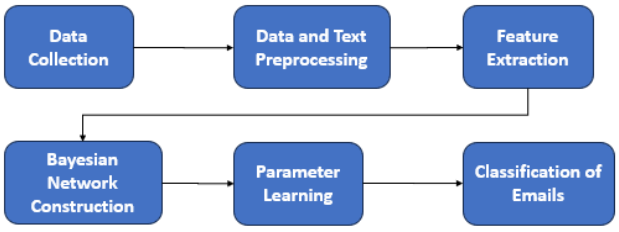


Fig-1 Flow diagram.

A. *Input Data:*

The input data is a text data which consists of 2000 ham and spam emails which is in the text format. It contains all the details like date, time, sender, receiver.

B. *Data Preprocessing:*

The collected data is processed by going under through text processing methods like removing unwanted words, phrases.

C. *Feature Extraction:*

The features are extracted so that it decides how the word is depends on the previous word. Mainly like the conditional probabilities.

D. *Bayesian Network Construction:*

The model is constructed based on the features extracted from the feature extraction that is from the previous step.

E. *Parametric Learning:*

The model is trained again and again so that accuracy should be so much high, and it should predict the email correctly.

F. *Classification of E-mails:*

Now the model is able to predict the email belongs to spam or ham. All the test data is loaded in to the model and predicted. After the prediction the metrics were calculated.

V. RESULTS

The following are the results of the Bayesian Net work model…Preprocessing, Confusion Matrix, Metrics.

```
loading train files...
loading test files...
processing train documents...
building vocabulary...
storing the vocabulary in results/model.txt...

creating NaiveBayesClassifier Model...
feeding vocabulary to classifier...
running the classifier on test documents...

classification done, result stored at results/result.txt...
printing the perfomance measures...
```

Fig-2 Preprocessing

```
          CONFUSION_MATRIX

                +----------------------+----------------------+
                |  (Predicted) SPAM    |  (Predicted) HAM     |
+---------------+----------------------+----------------------+
| (Actual) SPAM |         336          |          64          |
+---------------+----------------------+----------------------+
| (Actual) HAM  |          6           |         394          |
+---------------+----------------------+----------------------+

Accuracy measure:  0.996294875

Precision measure: 0.9824561403508771
...
recall measure:    0.84

f1-measure:        0.9056603773584906
```

Fig-3 Confusion Matrix

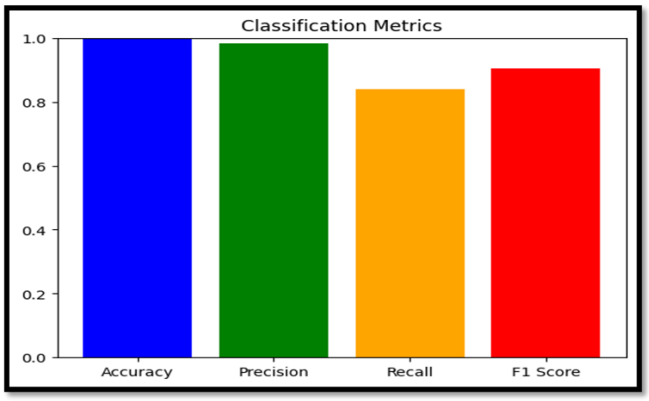


Fig-4 Metrics

results > result.txt

1	1	test-ham-00001.txt	ham	-935.3071953153601	-1160.1305098227383	ham	right
2	2	test-ham-00002.txt	ham	-963.3446868449001	-1105.9126400822836	ham	right
3	3	test-ham-00003.txt	ham	-1294.7894765449334	-1526.3732275114962	ham	right
4	4	test-ham-00004.txt	ham	-1148.3350997328148	-1419.0451968460818	ham	right
5	5	test-ham-00005.txt	ham	-1071.2899997467925	-1218.3029912116874	ham	right
6	6	test-ham-00006.txt	ham	-1890.8647986098306	-2130.4781496430360	ham	right
7	7	test-ham-00007.txt	ham	-3969.0310637028015	-4248.341480330589	ham	right
8	8	test-ham-00008.txt	ham	-988.1818707783078	-1199.7526538846598	ham	right
9	9	test-ham-00009.txt	ham	-1313.368563063242	-1552.1457918173558	ham	right
10	10	test-ham-00010.txt	ham	-1116.056509358293	-1365.1981297939562	ham	right
11	11	test-ham-00011.txt	ham	-1570.226867762133	-1820.9407859513356	ham	right
12	12	test-ham-00012.txt	ham	-1737.34188756435 76	-2019.3647940639455	ham	right
13	13	test-ham-00013.txt	ham	-1194.0770477659562	-1399.572692579172	ham	right
14	14	test-ham-00014.txt	ham	-1661.1312423048664	-1939.7145084324393	ham	right
15	15	test-ham-00015.txt	ham	-962.1118016917396	-1125.5726625376 61	ham	right
16	16	test-ham-00016.txt	ham	-972.5731803393513	-1135.9228331318852	ham	right
17	17	test-ham-00017.txt	ham	-962.1717305707627	-1124.938302786393	ham	right
18	18	test-ham-00018.txt	ham	-8763.220833413858	-8966.73897769752	ham	right
19	19	test-ham-00019.txt	ham	-1261.6396289348813	-1499.007037274292	ham	right
20	20	test-ham-00020.txt	ham	-1673.8329228308862	-1869.7491117408288	ham	right
21	21	test-ham-00021.txt	ham	-2553.3016149724 76	-2771.6026826365583	ham	right
22	22	test-ham-00022.txt	ham	-6515.486117158342	-6749.445348215974	ham	right
23	23	test-ham-00023.txt	ham	-1769.115847433908	-1957.8103294152102	ham	right
24	24	test-ham-00024.txt	ham	-947.8685981716214	-1110.1443602708941	ham	right
25	25	test-ham-00025.txt	ham	-1264.3965712427623	-1463.5636017028078	ham	right
26	26	test-ham-00026.txt	ham	-1053.9833289983626	-1210.6284931171897	ham	right
27	27	test-ham-00027.txt	ham	-1373.6035682424676	-1544.7953145746237	ham	right
28	28	test-ham-00028.txt	ham	-2728.904605486003	-2959.0273410807854	ham	right

Fig-3 Results.txt

VI. CONCLUSION

In conclusion, Bayesian networks offer a powerful and versatile framework for modelling and reasoning under uncertainty, making them invaluable in the realm of spam mail detection. The probabilistic graphical models provided by Bayesian networks allow us to represent complex relationships among variables in a succinct and interpretable manner. Through the lens of Bayes' theorem, Bayesian networks enable the dynamic updating of probabilities as new evidence emerges, facilitating adaptive and learning-centric systems. This adaptability is crucial in addressing the ever-evolving landscape of spam tactics, where traditional methods often fall short. The personalized touch of Bayesian networks, incorporating user feedback and preferences, enhances the accuracy of spam detection while minimizing false positives. This user-centric approach ensures that legitimate messages are not erroneously flagged, providing a more tailored and efficient email filtering experience. Moreover, the scalability of Bayesian networks makes them well-suited for handling the exponential growth in email exchanges, processing large datasets with efficiency. As we continue to navigate the challenges of our interconnected digital world, the robust and scalable nature of Bayesian networks positions them as a transformative solution in fortifying our email security infrastructure.

As we navigate the intricate landscape of email security, Bayesian networks emerge as a beacon of innovation, offering a holistic and intelligent approach to spam mail detection. Their ability to adapt, learn, and personalize sets the stage for a more secure and efficient email communication environment. By embracing the power of Bayesian networks, we embark on a journey towards a future where our inboxes are not just filtered but safeguarded with a level of intelligence that matches the dynamic nature of the digital era.

REFERENCES

[1] Ahmed, N., Amin, R., Aldabbas, H., Koundal, D., Alouffi, B., & Shah, T. (2022). Machine learning techniques for spam detection in email and IOT platforms: Analysis and research challenges. Security and Communication Networks, 2022, 1–19.

[2] Tida, V. S., & Hsu, S. H. (2022). Universal spam detection using transfer learning of Bert model. In Proceedings of the Annual Hawaii International Conference on System Sciences.

[3] Al Nabki W, Fidalgo E, Alegre E, Alaiz R (2020) File name classifcation approach to identify child sexualabuse. In: Conference: 9th international conference on pattern recognition applications and methods, pp 228–234.

[4] Annadatha A, Stamp M (2016) Image spam analysis and detection. J Comput Virol Hacking Tech 14(1):39–52.

[5] Grobbelaar, M., Phadikar, S., Ghaderpour, E., Struck, A.F., Sinha, N., Ghosh, R. and Ahmed, M.Z.I., 2022. A survey on denoising techniques of electroencephalogram signals using wavelet transform. *Signals*, *3*(3), pp.577-586.

[6] Zhou, X., Kan, Z., Meng, H. and Li, Y., 2023. Research on Trenching Data Correction Method Based on Wavelet Denoising-Kalman Filtering Algorithm. *Arabian Journal for Science and Engineering*, *48*(2), pp.1097-1117.

[7] Sarafan, S., Vuong, H., Jilani, D., Malhotra, S., Lau, M.P., Vishwanath, M., Ghirmai, T. and Cao, H., 2022, July. A novel ecg denoising scheme using the ensemble Kalman filter. In 2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC) (pp. 2005-2008). IEEE.

[8] You, N., Han, L., Zhu, D. and Song, W., 2023. Research on image denoising in edge detection based on wavelet transform. Applied Sciences, 13(3), p.1837.

[9] Shi, J., Chen, G., Zhao, Y. and Tao, R., 2023. Synchrosqueezed Fractional Wavelet Transform: A New High-Resolution Time-Frequency Representation. IEEE Transactions on Signal Processing, 71, pp.264-278.

[10] Roonizi, A.K. and Selesnick, I.W., 2022. A Kalman Filter Framework for Simultaneous LTI Filtering and Total Variation Denoising. IEEE Transactions on Signal Processing, 70, pp.4543-4554.