A Novel Fuzzy-Logic-Based Multi-Criteria Metric for Performance Evaluation of Spam Email Detection Algorithms

Salman A. Khan 1,*, Kashif Iqbal 1, Nazeeruddin Mohammad 2, Rehan Akbar 3, Syed Saad Azhar Ali 4, and Ammar Ahmed Siddiqui 5

1 College of Computing and Information Sciences, Karachi Institute of Economics and Technology, Karachi 75190, Pakistan; ghani.kashif@gmail.com
2 Cyber Security Center, Prince Mohammad Bin Fahd University, Dharan 34754, Saudi Arabia; nmohammad@pmu.edu.sa
3 Computer and Information Sciences Department, Universiti Teknologi PETRONAS, Seri Iskandar 32610, Malaysia; rehan.akbar@utp.edu.my
4 Center for Intelligent Signal and Imaging Research (CISIR), Department of Electrical and Electronics Engineering, Universiti Teknologi PETRONAS, Seri Iskandar 32610, Malaysia; saad.azhar@utp.edu.my
5 Department of Business Administration, Iqra University, Karachi 75500, Pakistan; ammarsiddiqui@iqra.edu.pk

* Correspondence: sakhan@kiet.edu.pk or sakhan@pafkiet.edu.pk

Abstract: The increasing volume of unsolicited bulk emails has become a major threat to global security. While a significant amount of research has been carried out in terms of proposing new and better algorithms for email spam detection, relatively less attention has been given to evaluation metrics. Some widely used metrics include accuracy, recall, precision, and F-score. This paper proposes a new evaluation metric based on the concepts of fuzzy logic. The proposed metric, termed \( \mu_{O} \), combines accuracy, recall, and precision into a multi-criteria fuzzy function. Several possible evaluation rules are proposed. As proof of concept, a preliminary empirical analysis of the proposed scheme is carried out using two models, namely BERT (Bidirectional Encoder Representations from Transformers) and LSTM (Long short-term memory) from the domain of deep learning, while utilizing three benchmark datasets. Results indicate that for the Enron and PU datasets, LSTM produces better results of \( \mu_{O} \), with the values in the range of 0.88 to 0.96, whereas BERT generates better values of \( \mu_{O} \) in the range of 0.94 to 0.96 for Lingspam dataset. Furthermore, extrinsic evaluation confirms the effectiveness of the proposed fuzzy logic metric.

Keywords: spam detection; performance evaluation; fuzzy logic; multi-criteria decision-making; BERT; LSTM

1. Introduction

Emails have become a fundamental part of today’s technology-augmented modern lifestyle. Emails have enabled people to collaborate with each other by providing one of the cheapest and fastest means of communication [1]. Since their emergence in the public domain in the mid 1990s, the positive impact of emails usage can be clearly seen in the growth of several areas, including business, education, healthcare, and industry, among others.

Despite the positive effects of emails, the most prominent weakness of the technology is the continuous increase in the unsolicited email messages that end up in a recipient’s inbox. Therefore, it is a task of utmost importance to segregate legitimate emails, called hams, from undesired emails known as spam. According to a recent study [2], spam messages account for 56.87% of email traffic worldwide, and the most common type of spam emails are healthcare and dating spam. Similarly, another recent study [3] indicates that 33.2 billion emails are expected to be generated worldwide on daily basis in the year...
2022. This number is estimated to reach 376.4 billion in 2025, with 4.6 billion users [3]. Consequently, the increase in spam emails is proportional to the increase in overall email traffic. The continuous increase in spam emails at a global scale has resulted in a high cost in terms of time, storage requirements, network bandwidth usage, and indirect costs of protecting privacy and security [4]. In addition, the nuisance caused by spam emails on human mental health and moods may have its own role, as indicated by studies [5,6].

The problem of spam email detection has several dimensions. The first dimension is concerned with the algorithms used for the spam detection. The prime objective of a spam detection algorithm is to segregate legitimate and spam emails as correctly as possible. For over two decades, researchers have proposed various solutions and methods, such as the use of Blacklist [7], Real-Time Blackhole List [8], and Content-Based Filters [9] to identify and filter spam from hams. Active research is still being conducted to develop methods that are faster and more accurate. In particular, artificial intelligence (AI)-based methods have received notable attention from researchers in recent years. Special emphasis has been placed on the methods based on machine learning [1,10–25]. In addition, deep learning methods have recently been successfully applied to spam email detection [10–14,26–28]. The outcomes of these studies indicate that machine learning and deep learning algorithms provide an effective platform for efficiently solving the spam detection problem; yet there is still a room for further improvement and development.

The second dimension of the spam detection problem is the datasets used to assess the performance of the spam detection algorithm. The dataset contained in a corpus plays a critical role in evaluating the performance of any spam detector [2]. A number of datasets have been reported in the literature, such as Spambase [29], Lingspam [29], PU1 [30], PU2 [31], PU3 [31], PU A [31], Enron [32,33], Genspam [34], Trec 2005 [35], Trec 2006 [36], and Trec 2007 [37]. A comprehensive list of email spam datasets available in the public domain is reported by Dada et al. [2]. The datasets have distinctive features and their suitability and effectiveness is determined by the type of experiments performed to evaluate the performance of the spam detection algorithm. In addition, several researchers use datasets that are not well known, or developed by the researchers themselves [1,12,17,19–22,24,28].

The third dimension is the performance metrics. In order to evaluate the performance of a spam detection algorithm using a dataset, an appropriate performance measure is necessary. One of the common metrics used for the evaluation is the accuracy, which measures the proportion of the total number of messages correctly classified (whether ham or spam) [15]. However, using accuracy as the only performance measure is not sufficient [2]. Therefore, other performance metrics such as recall and precision are also used [15]. The recall measures the number of messages correctly classified penalised by the number of missed entries, whereas the precision measures the number of messages correctly classified penalised by the number of incorrect classifications [15]. In addition to the aforementioned three metrics, which are frequently used in studies, other metrics such as the F-score [15], geometric mean (G-mean) [15], false positive rates (FPR) [2], false negative rate (FNR) [2], receiver operating characteristics (ROC) [15], and loss [26] are also employed. The use of the different performance metrics is vital in the spam detection process due to the costs associated with misclassification. This misclassification is twofold, since it is concerned with identifying a spam as a ham or vice versa. Although identifying a spam as a ham is a matter of concern, it is usually not a critical issue because the only action the user has to take is to delete such message. However, in the case of misclassifying a ham message as a spam, the gravity of this situation can lead to chaos, since valuable information may be lost to the classification error of the underlying detection algorithm. Therefore, it is of utmost importance that a spam detection algorithm is tested and evaluated with multiple metrics. Continuing with the performance metrics, the aforementioned discussion as well as the review of literature in Section 2 point towards classifying the metrics into different levels. The first-level metrics are the ones that are used at the atomic level. These include true positive, true negative, false positive, and false negative. Then there are second-level metrics such as accuracy, precision, and recall, which are comprised of a
certain combination of primary metrics. Finally, there are third-level metrics such as F-score and G-mean, which are based on secondary level metrics. Note that both F-score and G-mean combine two second-level metrics (F1 combines precision and recall, whereas G-mean combines specificity and recall). While there has been a great deal of effort in terms of developing new algorithms and datasets for the spam email detection problem, little emphasis has been placed on the evolution of new performance metrics, especially the third-level metrics. More specifically, while the first- and second-level metrics are frequently used in studies, as highlighted by the review of literature in Section 2, the impact of existing third-level metrics or the development of new metrics has received little attention from the research community.

The main motivation behind the underlying study is to propose a metric that combines three second-level metrics—accuracy, recall, and precision—and therefore aggregate the effect of all three in one measure, rather than looking at the three individually. This aggregated approach gives convenience to the decision maker in reaching a conclusive decision. For example, consider a situation where we have three algorithms A, B, and C. Assume some hypothetical results as shown in Table 1 below:

| Algorithm | Accuracy | Precision | Recall |
|-----------|----------|-----------|--------|
| A         | 85%      | 80%       | 88%    |
| B         | 76%      | 87%       | 72%    |
| C         | 89%      | 78%       | 81%    |

In the above scenario, A performs best in terms of Accuracy, B performs best in terms of Precision, and C performs best in terms of Recall, but which algorithm is the best in all respects? The answer is difficult and requires human intervention and expertise to reach a conclusion. In contrast, our proposed fuzzy measure aggregates the impact of all three measures, and therefore gives a single numerical value which can be easily and clearly evaluated without any confusion or human intervention. Furthermore, the commonly used measures of F1 score or G-mean would fail here, because they only aggregate the impact of only two second-level measures (as highlighted above). In this case, our proposed fuzzy metric is a suitable solution if it comes to measuring the combined effect of the three second-level metrics.

Motivated by the aforementioned observations, the major and minor contributions of this study can be enumerated as follows:

1. As a major contribution, a fuzzy-logic-based performance measure is proposed. This measure combines accuracy, precision, and recall into a single fuzzy function using the Unified And-Or (UAO) fuzzy operator. Several fuzzy decision rules are developed that can be used in the performance evaluation process. To the best of our knowledge, no such attempt has been reported in the literature. It is worth mentioning that while the proposed measure is evaluated in the context of spam email detection, the measure is generic and can be used in any problem domain concerned with performance evaluation.

2. As proof of concept, two deep learning models are used for comparative performance evaluation. These models are Bidirectional Encoder Representations from Transformers (BERT) and Long Short-Term Memory (LSTM). The use of BERT and LSTM is motivated by the fact that the models have seen a limited use for spam detection (as highlighted in the literature review in Section 2). This is a minor contribution, which supports the major contribution listed above.

3. As another minor contribution, a preliminary performance evaluation is carried out using several commonly used benchmark datasets. These datasets include Enron [32], Lingspam [38], and PU [39].
The rest of the paper is organized as follows. Section 2 presents the literature review. Section 3 provides a brief description of two deep learning models, namely BERT and LSTM, as used in this study. Then in Section 4, the fuzzy-logic-based performance measure is discussed, with details on the Unified And-Or (UAO) aggregation operator (which serves as the basis of the new performance metric), the fuzzy membership functions, and the decision rules. Results and discussions are provided in Section 5. Finally, a conclusion and future research directions are given in Section 6.

2. Literature Review

Table 2 provides a summary of the studies showing approaches, datasets, and performance metrics used in different studies. AbdulNabi and Yaseen [10] utilize BERT for spam detection and compare the performance with Deep Neural Network (Bidirectional LSTM), KNN, and NB. Spambase and Spamfilter (from Kaggle) are used, with accuracy and F-score as the performance measures. Islam et al. [11] mutually compare several algorithms which include regression, xboost, SVM, random forest, word embedding, and LSTM. Five datasets are used, which are TrecSpam 2007, Enron, PU, Lingspam, and Basket (which is the combination of the four datasets). Accuracy, loss, precision, and recall are used as the performance metrics. Srinivasan et al. [15] develop the DeepSpamNet algorithm by combining LSTM and CNN. The proposed algorithm is compared with logistic regression (LR), Gaussian NB, KNN, decision tree, adaboost, random forest, and SVM. Four datasets, namely Lingspam, PU, Spam Assasin, and Enron, are used. The study employs accuracy, precision, recall, F-score, and G-mean as the performance measures. Kumar et al. [16] also evaluate the performance of SVR, KNN, NB, decision tree, random forest, adaboost, and bagging. However, only a single dataset is used, which is Spambase from Kaggle. Moreover, accuracy and precision are used as the performance metrics.

Anitha et al. [17] utilize xboost for spam email detection. The xboost algorithm is compared with several algorithms, including SVM, rotation forest, MLP, decision tree (J48), and NB. Performance is evaluated using a single dataset called Spam_ham_dataset from Kaggle. Five performance measures are employed, which are accuracy, precision, recall, specificity, and F-score. Siddique et al. [12] employ NB, CNN, SVM, and LSTM for spam email detection in Urdu. A self-collected dataset named “Urduemaidataset” is used for evaluation. Four metrics, namely accuracy, precision, recall, and F-score, are used for performance evaluation. Sethi et al. [18] use a single dataset (SpamAssassin) to compare the performance of NN, Logistic regression, SVM, and NB. Performance is measured with respect to accuracy, precision, recall, and F-score.

Bagui et al. [19] apply NB, SVM, decision tree, CNN, and LSTM for the spam detection problem. However, they only use accuracy as the performance measure while using a single dataset which consists of self-collected emails. Nayak et al. [20] employ NB and decision tree (J48) while using a single dataset from Kaggle. The performance metrics are accuracy, precision, recall, and F-score. Sheneamer [13] utilize several algorithms including CNN, LSTM, Random forest, SVM, NB, and decision tree while using only the Enron dataset. Three metrics, namely accuracy, recall, and precision are used for performance evaluation. Euna et al. [21] also evaluate the performance of several algorithms, which include SVM, decision tree, logistic regression, and multinomial NB. A single dataset from Kaggle is used with accuracy, precision, recall, and F-score as the performance measures.

Zamir et al. [28] mutually compare adaboost, random forest, decision tree (J48), SVM, bagging, and deep learning. Performance is evaluated with accuracy, precision, recall, and F1-score while using a single dataset called CSDM2010_SPAM. Kaddoura et al. [27] evaluate BERT using the Enron dataset. A single evaluation metric, namely the F-score, is used. Feng et al. [1] propose an algorithm combining SVM and NB and call it SVM-NB. The proposed algorithm is compared with SVM and NB algorithms using the DATAMALL dataset. Recall and precision are used for performance evaluation. Chakraborty and Mondal [22] employ NB, J48 decision tree, and logistic tree classifier algorithms for spam
detection. They measure the performance of the three algorithms using accuracy and precision while utilizing a dataset of self-collected emails.

Mallampati and Hegde [14] implement NB, J48 decision tree, and deep neural network. The performance of the three algorithms is evaluated using the Spam Assassin dataset, with accuracy, precision, and recall as the performance metrics. Rusland et al. [23] use NB with two datasets, namely Spambase and Spamdata. Performance is evaluated using accuracy, recall, precision, and F-score. Bibi et al. [24] employ NB and SVM while using a dataset from Github with accuracy, recall, precision, and F-score. Iqbal et al. [26] implement BERT and evaluate the algorithm over the Enron dataset. Accuracy is used as the performance measure.

Novo-Lourés et al. [40] apply Adaboost, Flexible Bayes, NB, RF, SVM to email spam detection while using Spamassassin dataset. The performance is measured using accuracy, precision, recall, F-score, and other metrics. Occhipinti et al. [41] employ twelve machine learning algorithms which include K-NN, Xboost, and several variants of regression, SVM, and NB. The performance is evaluated using the Enron dataset while using recall, precision, F-score, and a couple of other metrics. Guo et al. [42] use several algorithms including SVM, MLP, and CNN using Twitter and SinaWeibo datasets. The performance metrics are accuracy, recall, precision, and F-score. Venkateswarlu et al. [43] propose a variant of generative adversarial network (GAN) for twitter spam detection using a single dataset (i.e., Twitter Spam). They use precision, recall, and F-score as the evaluation metrics.
Table 2. Summary of studies on email spam detection. A—accuracy; R—recall author; P—precision; L—loss, F—f-score/F1 score; G—G-mean.

| Reference | Year | Algorithms | Datasets | Performance Metrics |
|-----------|------|------------|----------|---------------------|
| [1]       | 2016 | CVM-NB, SVM, NB | Datamall | A X R X P X L X F X G X |
| [10]      | 2021 | BERT, Bidirectional LSTM, KNN, NB | Spambase, Spam filter from Kaggle | X |
| [11]      | 2021 | Regression, Xgboost, SVM, RF, WE, LSTM | Trec spam Dataset-2007, Enron PU, Lingspam, Basket(combo of four) | X X X X X |
| [12]      | 2021 | NB, CNN, SVM, LSTM | Urduemaildataset | X X X X |
| [13]      | 2021 | CNN, LSTM, RF, SVM, NB, DT | Enron | X X X |
| [14]      | 2020 | NB, DT (J48), DNN | SpamAssasin | X X X |
| [15]      | 2021 | LR, GNB, KNN, DT, Adaboost, RF, SVM, DeepSpamNet | Lingspam, PU, Spam Assassin, Enron | X X X X X |
| [16]      | 2020 | SVC, KNN, NB, DT, RF, Adaboost, Bagging | Spamcsv from Kaggale | X X |
| [17]      | 2021 | SVM, CNS-FFO, Rotation Forest, MLP, DT (J48), NB, Xboost | Spam_ham_dataset from Kaggle | X X X X X |
| [18]      | 2021 | NN, LR, SVM, NB | Spam Assassin | X X X X |
| [19]      | 2021 | NV, SVM, DT, CNN, LSTM | Self-collected | X |
| [20]      | 2021 | NB, DT (J48) | Dataset from Kaggle | X X X |
| [21]      | 2021 | SVM, DT, LR, MNB | Dataset from Kaggle | X X X |
| [22]      | 2012 | NB, DT (J48), LTC | Self-collected | X X |
| [23]      | 2017 | NB | Spambase, Spamdata | X X X |
| [24]      | 2020 | NB, SVM | Dataset from github | X X X |
| [26]      | 2022 | BERT | Enron | X X |
| [27]      | 2020 | BERT | Enron | X |
| [28]      | 2020 | Adaboost, RF, DT (J48), SVM, Baggin, DL | CSDMC2010.SPAM | X X X X |
| [40]      | 2022 | Adaboost, Flexible Bayes, NB, RF, SVM | Spam Assassin | X X X X |
| [41]      | 2022 | KNN, MPNN, LR, RF, XGBoost, MNB, GNB, BNB, RBF SVM, LSVM, PSVM, SSVM | Enron | X X X X |
| [42]      | 2021 | LDA+k-means, LSTM+lr, SVM, MLP, CNN | Twitter, SinaWeibo | X X X |
| [43]      | 2021 | GAN, NB, LSTM, SVM | Twitter Spam | X X X |
3. Brief Background of BERT and LSTM

It is important to mention that the focus of this study is not on the algorithm performance, but to assess the impact of a fuzzy-logic-based metric when used in conjunction with an email spam detection algorithm. Therefore, as proof of concept, two models from the domain of deep learning have been used, which are BERT and LSTM. A brief description of these models is given below.

3.1. Bidirectional Encoders Representations from Transformers (BERT)

BERT [44] is a text representation model. The model combines bidirectional encoder LSTM and Transformers. Developed in 2018, the model has worked effectively for natural language processing tasks such as text classification, text summarization, and text generation. BERT is capable of pretraining deep bidirectional representations from unlabeled text [44]. This is achieved by jointly conditioning on both the left and right context in all layers [44]. Due to this, the pretrained BERT model can be fine-tuned with just one additional output layer. This creates cutting-edge models for a variety of tasks such as question answering and language inference, with only small modifications in a task-specific architecture. Few studies have used BERT for spam email detection [10, 26, 27]. A detailed discussion on BERT and its architecture can be found in Devlin et al. [44].

We have accomplished creation of a more efficient Natural Language Processing (NLP) model with BERT Tokenizer along with TensorFlow 2.0 (TF 2.0). Studies show that BERT is effective for various NLP tasks such text classification, text generation, and text summarization. On the basis of text classification, our proposed model (BERT + TF 2.0) is used to classify email (ham or spam) in the underlying study.

3.2. Long Short-Term Memory (LSTM)

The LSTMs falls in the domain of deep learning [45]. LSTM belongs to a particular class of recurrent neural networks (RNNs). LSTMs utilize their inherent memory capabilities, making use of the past information to make efficient predictions. As part of RNN, LSTM have demonstrated a great potential in capturing both the long-term and the short-term temporal dependencies within the time series data. A primary advantage of the LSTM network is that it overcomes the vanishing and the exploding gradient problem inherent in the conventional RNNs. An LSTM network accepts sequential time series data as input. The use of LSTM for email spam detection has been reported in several studies [10–12]. A detailed discussion on LSTM can be found in Hochreiter and Schmidhuber [45].

In the current study, LSTM is used as another model so that results can be compared with BERT, since it is widely used in sequence models.

4. Fuzzy-Logic-Based Multi-Criteria Evaluation Metric for Spam Detection Approaches

The foundations of fuzzy logic (FL) were developed by Lotfi Zadeh [46]. In the traditional set theory, a set is defined as collection of elements $x \in X$, where an element can either belong to the set or not. In other words, the belonging of an element to a set is either TRUE or FALSE. However, in many practical scenarios, this rigid, binary approach of belonging to a set is not effective. There could be situations where an element may partially belong to a set, thus making this belonging partially true (and partially false). This observation leads to the development of fuzzy rules, and application of FL in the domain of multi-criteria decision-making (MCDM).

The domain of MCDM deals with decision problems where multiple factors (i.e., criteria) are used in the decision process [47,48]. More specifically, the main purpose of MCDM is to identify the best solution among a number of alternatives for a given problem. Over the years, several methods have been proposed to deal with the MCDM problems. As such, fuzzy logic has also been applied effectively to solve MCDM problems. The main ingredients of the FL based MCDM approach are the decision criteria, membership functions for the decision criteria, decision rules, and mathematical representation of the decision rule. These concepts are discussed in the next section in the context of the spam detection problem.
4.1. Decision Criteria for the Spam Detection Metric

The first issue in the development of the fuzzy-logic-based metric is the identification of decision criteria. Decision criteria play a pivotal role in the decision process. The domain expert, who has complete knowledge of the problem under study, identifies the most appropriate decision criteria. Selection of inappropriate decision criteria leads to inaccurate solutions. In the context of the current study, the literature review has identified several commonly used metrics such as accuracy, recall, precision, and F-score. Note that accuracy, recall, and precision are based on the four primary level attributes given below, and F-score is a derived metric (considered as a third-level metric) based on recall and precision, and therefore is not considered as a decision criterion in this study. The four primary level attributes are as follows [15]:

- True positive ($T_P$): a legitimate email correctly classified as a legitimate email.
- True negative ($T_N$): a spam email correctly classified as spam email.
- False positive ($F_P$): a legitimate email incorrectly classified as spam.
- False negative ($F_N$): a spam email incorrectly classified as legitimate.

Based on the above four attributes, accuracy, which measures the proportion of the total number of correct classifications [15] is mathematically represented as follows:

$$\text{Accuracy} = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \quad (1)$$

Similarly, recall measures the number of correct classifications penalised by the number of missed entries [15], and is represented by the following equation:

$$\text{Recall} = \frac{T_P}{T_P + F_N} \quad (2)$$

Likewise, the precision is a measure of the number of correct classifications penalised by the number of incorrect classifications, as given by the following equation

$$\text{Precision} = \frac{T_P}{T_P + F_P} \quad (3)$$

It should be noted that while the F-score reflects the impact of precision and recall, no such measure exists that combines the effect of accuracy, precision, and recall in a single function. Therefore, there is a need to develop such a measure.

4.2. Membership Functions for the Decision Criteria

The next step in the process is the employment of fuzzy logic for combining the three decision criteria of accuracy, precision, and recall into a single, compound metric. This compound metric evaluates the quality of a solution in terms of membership values of accuracy, precision, and recall. A membership function maps the actual values of the three criteria onto a [0–1] scale. The mapped value is referred to as the membership value and is commonly represented by the symbol $\mu$.

To formulate the fuzzy-logic-based compound metric, the values of individual criteria need to be determined first, through membership functions that need to be defined for each criterion. These membership functions are Min–Max functions, where the x-axis defines the upper and lower limits for a given criterion, while the y-axis represents the corresponding membership value between 0 and 1. Accordingly, the membership functions for the accuracy, precision, and recall are represented by $\mu_A(x)$, $\mu_P(x)$, and $\mu_R(x)$. The corresponding mathematical representations are given in Equations (4)–(6). Since all three criteria vary between 0% and 100%, the upper and lower limits on the x-axis are defined as such.
4.3. Fuzzy Decision Rules for Spam Detection

Khan and Engelbrecht [49] proposed an approach of using fuzzy decision rules for MCDM problems. The approach can be extended to the problem addressed in this paper. The approach suggests that several fuzzy decision rules can be developed using the metrics of accuracy, recall, and precision. In each rule, the impact of the three decision metrics is reflected in different ways. The result of each rule classifies the performance of a given spam detection technique. The rule-based approach gives flexibility to the decision-maker to define a rule depending upon how many decision metrics are desired in the decision process. Accordingly, the different possible cases are discussed below. Note that the rules are scalable and more decision metrics can be added to the decision process. However, as proof of concept, only the aforementioned three decision metrics are considered.

Case 1: Performance evaluation based on all three decision metrics
This is one extreme in which all three decision metrics are considered in the performance evaluation. This scenario is represented by the following fuzzy rule. In this case, the fuzzy rule would be as follows:

- Rule R1: IF Accuracy is high AND Precision is high AND Recall is high THEN the Performance is Excellent.

Case 2: Performance evaluation based on any two decision metrics
In this case, any two of the three metrics are considered in the decision process. This leads to several fuzzy rules, as follows:

- Rule R2a: IF Accuracy is high AND (Precision is high OR Recall is high) THEN the Performance is Excellent.
- Rule R2b: IF (Accuracy is high OR Precision is high) AND Recall is high THEN the Performance is Excellent.
- Rule R2c: IF (Accuracy is high OR Recall is high) AND Precision is high THEN the Performance is Excellent.

Case 3: Performance evaluation based on any one decision metric
The other extreme is the case where any one metric is used in the decision process. The fuzzy rule is as given below:

- Rule R3: IF Accuracy is high OR Precision is high OR Recall is high THEN the Performance is Excellent.

4.4. Mathematical Representation of Decision Rules

In order to evaluate the impact of a fuzzy decision rule, transformation of the rule into a mathematical form is essential for a tangible outcome. This is achieved through a
fuzzy aggregation function. Over the years, a number of fuzzy aggregation functions have been proposed in the literature. Some well-known functions are Werner’s function, Dubois and Prade functions, Yager’s ordered weighted average (OWA) functions, Hamacher’s function, Einstein’s function, and the Unified AND-OR (UAO) function, among many others [50]. Among these operators, the UAO function [49] has some unique features (see Khan and Engelbrecht [49] for details and mathematical properties of the function). Due to the unique characteristics of the UAO function, the function has been employed effectively in many diverse studies [51–57]. Therefore, the UAO function is also used in the underlying study. The prime uniqueness of the UAO function is that it can behave as the “AND” function or the “OR” function. This “ANDing” or “ORing” is controlled by a variable \( \nu \geq 0 \), whose value decides the behavior of the function. For the spam detection problem, the aforementioned three membership functions, accuracy, precision, and recall, are embedded in the UAO function as follows:

\[
\mu_O(x) = \frac{\mu_A(x)\mu_P(x)\mu_R(x) + \nu \max\{\mu_A(x), \mu_P(x), \mu_R(x)\}}{\nu + \max\{\mu_A(x), \mu_P(x), \mu_R(x)\}} = \begin{cases} 
I_A = \mu_A(x) & \text{if } \nu > 1 \\
I_O = \mu_A(x)\mu_P(x)\mu_R(x) & \text{if } \nu < 1
\end{cases}
\]

where \( \mu_O \) represents the overall membership value of the decision. \( I_A \) signifies the AND function while \( I_O \) represents the OR function. With \( 0 < \nu < 1 \), UAO behaves as the AND function, while \( \nu > 0 \) gives the OR behavior. Using Equation (7), the mathematical structures corresponding to the fuzzy decision rules given in Section 4.3 are represented as follows:

- Rule 1: \( \mu_O = I_A(x) \)
- Rule 2a: \( \mu_O = I_A(x)I_O(x) \)
- Rule 2b: \( \mu_O = I_A(x)I_O(x)\mu_R(x) \)
- Rule 2c: \( \mu_O = I_A(x)I_O(x)\mu_P(x) \)
- Rule 3: \( \mu_O = I_A(x)I_O(x)\mu_R(x) \)

In the above mathematical representations, the value of \( \mu_O \) is proportional to the individual membership values (i.e., \( \mu_A \), \( \mu_P \), and \( \mu_R \)). Therefore, high values of input membership values result in a high value of \( \mu_O \) and vice versa. Consequently, a high value of \( \mu_O \) indicates that the performance of the algorithm is on the high side, while a low \( \mu_O \) suggests a degraded performance. For example, a \( \mu_O = 0.95 \) indicates that the performance of a given algorithm is near the ideal performance (since \( \mu_O = 1 \) defines the ideal condition as per the fundamentals of fuzzy logic). Likewise, \( \mu_O = 0.85 \) indicates that the performance is good enough, but there is room for improvement.

5. Results and Discussion

To demonstrate the utility of the proposed fuzzy evaluation metric, empirical experiments were carried out. The following subsections show the datasets used for the study, as well as the reasons for selecting these datasets. Then, the details of the experiments and results are provided.

5.1. Characteristics of Datasets

Several email datasets have been reported in the literature. Two recent review articles by Dada et al. [2] in 2019 and Bhowmick et al. [58] in 2018 have listed a number of datasets, which are shown in Table 3. As seen from the table, the datasets were developed during the years 1998 and 2007. Since 2007, no new email dataset has been reported in the literature. Furthermore, another article [59] in 2020 identifies that among the known datasets (as given in Table 3), SpamAssassin, Enron, and PU datasets are publicly available datasets, and other datasets are generally not available or easily accessible in the public domain.

As far as the current study is concerned, three datasets, namely Enron [32], Lingspam [38], and PU [39], were used. The reasons for selecting these datasets are as follows:

- Many recent studies utilized the Enron, Lingspam, and PU datasets. As shown in Table 2, the use of Enron dataset has been reported in several studies between the
years 2020 and 2022 [11,13,15,26,27,41]. Similarly, the use of Lingspam and PU has also been reported in two recent studies published in 2021 [11,15]. This indicates that the datasets are still in active use in the research community.

- The emails in these datasets were generated between 2000 and 2010. This scenario characterizes the change in wordings and writing patterns in emails over a period of ten years [15]. Secondly, the Enron dataset is employed due to its bias towards spam class [15]. Thirdly, the LingSpam dataset is used due to its domain-specific ham mails, which are extracted from scholarly linguistic discussions [15]. Lastly, the PU dataset is used, since it is less frequently used by researchers, thus encouraging us to study the behavior of this dataset with respect to the aforementioned three algorithms and the proposed fuzzy-logic-based performance metric.

Table 3. A list of existing email datasets [2,58].

| Dataset                  | Total Emails | Spam | Ham | % of Spam | Year |
|--------------------------|--------------|------|-----|-----------|------|
| Spam archive             | 15,090       | 15,090 | 0   | 100%      | 1998 |
| Spambase                 | 4601         | 1813 | 2788 | 39%       | 1999 |
| Lingspam                 | 2893         | 481  | 2412 | 17%       | 2000 |
| PU1                      | 1099         | 481  | 618  | 44%       | 2000 |
| Spamassassin             | 6047         | 1897 | 4150 | 31%       | 2002 |
| PU2                      | 721          | 142  | 579  | 20%       | 2003 |
| PU3                      | 4139         | 1826 | 2313 | 44%       | 2003 |
| PUA                      | 1142         | 571  | 571  | 50%       | 2003 |
| Zh1                      | 1633         | 1205 | 428  | 74%       | 2004 |
| Gen spam                 | 40,408       | 31,196 | 9212 | 78%       | 2005 |
| Trec 2005                | 92,189       | 52,790 | 39,399 | 66%     | 2005 |
| Biggio                   | 8549         | 8549 | 0    | 100%      | 2005 |
| Phishing corpus          | 415          | 415  | 0    | 100%      | 2005 |
| Enron                    | 36,715       | 20,170 | 16,545 | 55%     | 2006 |
| Trec 2006                | 37,822       | 24,912 | 12,910 | 66%     | 2006 |
| Trec 2007                | 75,419       | 50,199 | 25,220 | 67%     | 2007 |

5.2. Empirical Results

Two sets of experiments were carried out. In the first set of experiments, the performance of the BERT and LSTM models was assessed using only Rule 1 as proof of concept. In the second set of experiments, the relative performance of the two algorithms was evaluated. The data in the three email datasets were divided in a 70/30 ratio, with 70% data for training and 30% for testing. Furthermore, binary cross entropy [15] was used as the loss function. Note that the purpose of this study is to show the applicability of the proposed fuzzy measure, and therefore only necessary details are provided without going into in-depth comparative analysis of the algorithms.

5.2.1. Evaluation of BERT and LSTM with Respect to the Three Datasets Using Rule R1

To show the applicability of the proposed scheme, Rule 1 was assessed as an example. A value of $\nu = 0.5$ was used. Tables 4 and 5 provide the results of the two models with respect to each dataset using Rule 1. In Table 4, it is observed that the BERT algorithm gives the best results for the LingSpam dataset, where the best value of $\mu_O$ was 0.96 (for 10 and 20 epochs). Furthermore, for the Enron and PU datasets, the best $\mu_O$ were 0.89 (for 30 epochs), and 0.82 (for 10 epochs), respectively. The interpretation is that for the LingSpam dataset, the performance of BERT (with regard to Rule 1) is near ideal (for 10 and 20 epochs), since $\mu_O = 0.96$ is very close to the ideal membership of $\mu_O = 1$. In the same sense, for the Enron and
PU datasets, the performance of BERT is somewhat distant from the ideal value of $\mu_O = 1$, since values of $\mu_O = 0.89$ and $\mu_O = 0.82$ were observed.

With regard to the LSTM model, the results in Table 5 are different from that of BERT. The results indicate that the best results in terms of $\mu_O$ were obtained for the Enron dataset, with the best $\mu_O$ of 0.96. The results of $\mu_O = 0.94$ were obtained for PU with 30 epochs, while for PU, the best $\mu_O$ was 0.92 for 20 and 30 epochs. Again, the results can be interpreted as explained in the paragraph above.

### Table 4. Results for testing data for different datasets using the BERT model.

| Dataset | Epochs | Accuracy | Precision | Recall | $\mu_O$ |
|---------|--------|----------|-----------|--------|---------|
| Enron   | 10     | 0.92     | 0.95      | 0.90   | 0.87    |
|         | 20     | 0.94     | 0.95      | 0.9    | 0.88    |
|         | 30     | 0.94     | 0.95      | 0.92   | 0.89    |
| LingSpam| 10     | 0.98     | 0.97      | 0.98   | 0.96    |
|         | 20     | 0.98     | 0.97      | 0.98   | 0.96    |
|         | 30     | 0.98     | 0.95      | 0.97   | 0.94    |
| PU      | 10     | 0.92     | 0.89      | 0.86   | 0.82    |
|         | 20     | 0.92     | 0.84      | 0.87   | 0.80    |
|         | 30     | 0.92     | 0.88      | 0.85   | 0.81    |

### Table 5. Results for testing data for different datasets using the LSTM model.

| Dataset | Epochs | Accuracy | Precision | Recall | $\mu_O$ |
|---------|--------|----------|-----------|--------|---------|
| Enron   | 10     | 0.96     | 0.98      | 0.98   | 0.95    |
|         | 20     | 0.97     | 0.98      | 0.98   | 0.96    |
|         | 30     | 0.97     | 0.98      | 0.98   | 0.96    |
| LingSpam| 10     | 0.96     | 0.96      | 0.94   | 0.92    |
|         | 20     | 0.98     | 0.96      | 0.95   | 0.93    |
|         | 30     | 0.98     | 0.95      | 0.97   | 0.94    |
| PU      | 10     | 0.93     | 0.95      | 0.9    | 0.88    |
|         | 20     | 0.95     | 0.95      | 0.95   | 0.92    |
|         | 30     | 0.96     | 0.96      | 0.94   | 0.92    |

### 5.2.2. Mutual Comparison of BERT and LSTM Using Rule R1

A mutual comparison of the two models was also performed with respect to the proposed fuzzy metric. Figure 1 shows the comparisons of the two models with respect to each dataset, while showing the value of $\mu_O$ for 10, 20, and 30 epochs. In Figure 1a, the plots indicate that as far as the Enron dataset is concerned, LSTM has a better performance compared to BERT for all epochs. With regard to the results for Lingspam dataset as given in Figure 1b, BERT shows better performance than LSTM for 10 and 20 epochs, but has equal performance for 30 epochs. Finally, for the PU dataset, the trends in Figure 1c are similar to what was observed for the Enron dataset as far as BERT is concerned; BERT has an inferior performance for all epochs.

### 5.3. Extrinsic Evaluation

A focus group was formed for extrinsic valuation based on human judgements. As depicted in Table 6, a focus group was formed comprising of 34 male and 7 female undergraduate university students with a computer science major. As an evaluation requirement, the focus group was provided with the same environment, data, situation, and time period for analyzing and decision-making. As each member of the focused group has a different mental approach towards the decision, this may result in creating a gray area/bias. Moreover, each member of the focused group was briefed about the objectives of research and steps involved in the basic evaluation process.
Table 6. Description of focus group demographics.

| Gender | Participants | Population | Occupation |
|--------|--------------|------------|------------|
| Male   | 34           | 83%        | Students   |
| Female | 7            | 17%        | Students   |
| Total  | 41           | 100%       | -          |

Figure 1. Comparison of BERT and LSTM using $\mu_O$ for (a) Enron (b) Lingspam (c) PU.

In the BERT model scenarios, decisions were based on the values of accuracy, precision and recall, as shown in Table 7. Condition/restriction was also incorporated, where each member had to opt for only one choice, and once. Considering the Enron dataset, the focused group decision-making results shows that all 34 (i.e., 100%) males opted for results
obtained with 30 epochs, with all 7 females (i.e., 100%) also opting for the same option. For the Lingpam dataset, the decision-making results for the focus group shows that 59% of the males opted for 20 epochs, while 41% females chose results obtained with 30 epochs. Furthermore, out of seven females, three opted for results with 20 epochs, while four preferred the results obtained with 30 epochs. Finally, for the PU dataset, the decision-making results for the focus group showed that out of 34 males, 10 opted for results obtained with 10 epochs, 17 opted for results obtained with 20 epochs, while 7 preferred the results obtained with 30 epochs. The results for females were somewhat showing similar trends, where two out of seven females chose results of 10 epochs, one opted for 20 epochs, while four opted for results with 30 epochs.

Table 7. Summary of extrinsic evaluation for the BERT model.

| Dataset | Epochs | Accuracy | Precision | Recall | Decision (Male) | Percentage (Population) | Decision (Female) | Percentage (Population) |
|---------|--------|----------|-----------|--------|----------------|-------------------------|-------------------|-------------------------|
| Enron   | 10     | 0.92     | 0.95      | 0.90   | 0              | 0%                     | 0                 | 0%                     |
|         | 20     | 0.94     | 0.95      | 0.9    | 0              | 0%                     | 0                 | 0%                     |
|         | 30     | 0.94     | 0.95      | 0.92   | 34             | 100%                   | 7                 | 100%                   |
| LingSpam| 10     | 0.98     | 0.97      | 0.98   | 20             | 59%                    | 3                 | 43%                    |
|         | 20     | 0.98     | 0.97      | 0.98   | 14             | 41%                    | 4                 | 57%                    |
|         | 30     | 0.98     | 0.95      | 0.97   | 0              | 0%                     | 0                 | 0%                     |
| PU      | 10     | 0.92     | 0.89      | 0.86   | 10             | 29%                    | 2                 | 29%                    |
|         | 20     | 0.92     | 0.84      | 0.87   | 17             | 50%                    | 1                 | 14%                    |
|         | 30     | 0.92     | 0.88      | 0.85   | 7              | 21%                    | 4                 | 57%                    |

As far as the results for LSTM are concerned, Table 8 illustrates the scenarios where the decisions were based on the values of accuracy, precision, and recall derived from the LSTM model. The same condition/restriction as for the BERT model was applied. For the Enron dataset, the decision-making results for the focused group showed that 21 males and 6 females opted for results obtained with 20 epochs, while 13 males and 1 female chose results obtained with 30 epochs. Furthermore, none of the males and females chose results with 10 epochs. With regard to the Lingspam dataset, the results indicate that 15 males and 3 females selected results of 20 epochs, while 19 males and 4 females opted for results with 30 epochs. Similar to the observations of Enron dataset, none of the males and female subjects opted for results with 10 epochs. Finally, for the PU dataset, 10 males and 2 females selected results of 20 epochs, while 24 males and 5 females selected results with 30 epochs. Furthermore, no males or females opted for results with 10 epochs.

Table 8. Summary of extrinsic evaluation for the LSTM model.

| Dataset | Epochs | Accuracy | Precision | Recall | Decision (Male) | Percentage (Population) | Decision (Female) | Percentage (Population) |
|---------|--------|----------|-----------|--------|----------------|-------------------------|-------------------|-------------------------|
| Enron   | 10     | 0.96     | 0.98      | 0.98   | 0              | 0%                     | 0                 | 0%                     |
|         | 20     | 0.97     | 0.98      | 0.98   | 21             | 62%                    | 6                 | 57%                    |
|         | 30     | 0.97     | 0.98      | 0.98   | 13             | 38%                    | 1                 | 43%                    |
| LingSpam| 10     | 0.96     | 0.96      | 0.94   | 0              | 0%                     | 0                 | 0%                     |
|         | 20     | 0.98     | 0.96      | 0.95   | 15             | 44%                    | 3                 | 43%                    |
|         | 30     | 0.98     | 0.95      | 0.97   | 19             | 56%                    | 4                 | 57%                    |
| PU      | 10     | 0.93     | 0.95      | 0.9    | 0              | 0%                     | 0                 | 0%                     |
|         | 20     | 0.95     | 0.95      | 0.95   | 10             | 29%                    | 2                 | 29%                    |
|         | 30     | 0.96     | 0.96      | 0.94   | 24             | 71%                    | 5                 | 71%                    |

While the results with respect to Enron and Lingspam datasets are non-conflicting for both BERT and LSTM models, a divided opinion is observed with the regard to the PU datasets for both BERT and LSTM models. That is, with accuracy, precision, and recall as the performance measures, it was not simple to select the best option for the human subjects, and therefore an automated decision-making approach that provides a concrete decision
was inevitable. What we conclude from the above results for the PU dataset is that it is very confusing and not easy to evaluate such human decisions, as in some cases, the respondents preferred 20 epochs over 30 (or vice versa) just because of different membership values for the three attributes. Moreover, in other cases, there is contradiction between males and females decision-making. For instance, for the PU dataset with the BERT Model, most males preferred results with 20 epochs, while most females preferred results obtained with 30 epochs. This shows that there is a need for a concluding matrix with a single value which would result in simplified and standardized decisions. This automated and standardized decision-making was provided by the proposed fuzzy-logic-based metric.

6. Conclusions

Spam emails are a major threat to network and data resources of an individual as well as organizations. Significant research has been carried out in the past to address spam email detection. Several elements are involved in this process. Among these elements, an effective performance metric that is able to incorporate the domain knowledge and reflects the performance of a spam detection algorithm is vital to the efficiency of the process. In this connection, a new performance metric has been proposed in this study. The new metric is based on fuzzy logic and combines the impact of accuracy, precision, and recall in a single, multi-criteria measure. For this purpose, decision rules have been developed, along with their corresponding mathematical representations in fuzzy logic. A preliminary analysis to assess the applicability of the proposed scheme has been carried out using one such fuzzy rule. The analysis consists of a comparative evaluation of two machine learning models, namely BERT and LSTM, while using three benchmark datasets, Enron, Lingspam, and PU. Results indicate that for the Enron and PU datasets, LSTM demonstrated a better performance. For the Lingspam dataset, the dominant trend indicates that BERT is a better option.

The proposed work has strong potential to evolve in several directions. For example, in addition to the three metrics used in the current study, other metrics (such as loss) can be added into the fuzzy model and the decision rules can be modified accordingly. Furthermore, the impact of other mathematical representations (such as the ones mentioned in Section 4.4) on the algorithm performance can be studied. Additionally, performance of other recent approaches such as XML and GAN can be evaluated while using more benchmark datasets. To add granularity to the fuzzy rule-based approach, preference rules [49] on top of the fuzzy decision rules can be added. Moreover, the performance of the proposed fuzzy metric can be evaluated for diverse problems requiring performance evaluation in various other domains.

Author Contributions: Conceptualization, S.A.K. and K.I.; methodology, S.A.K. and K.I. and A.A.S.; software, K.I.; validation, S.A.K., K.I., N.M., R.A., S.S.A.A. and A.A.S.; formal analysis, S.A.K. and K.I. and N.M.; investigation, S.A.K., K.I., N.M., R.A., S.S.A.A. and A.A.S.; resources, S.A.K. and N.M. and S.S.A.A. and R.A.; data curation, S.A.K. and K.I. and A.A.S.; writing–original draft preparation, S.A.K., K.I., N.M., R.A., S.S.A.A. and A.A.S.; writing–review and editing, S.A.K., K.I., N.M., R.A., S.S.A.A. and A.A.S.; supervision, S.A.K.; project administration, S.A.K.; funding acquisition, S.A.K. All authors have read and agreed to the published version of the manuscript.

Funding: The Cybersecurity Center at Prince Mohammad bin Fahd University is acknowledged for supporting this research under grant # PCC-Grant-202116. The last two authors (SSAA and RA) would like to acknowledge Murata Science Foundation and Univesiti Teknologi PETRONAS.

Institutional Review Board Statement: The study was conducted according to the guidelines of the Declaration of Helsinki, and approved by the Ethics Committee of College of Computing and Information Sciences at Karachi Institute of Economics and Technology as per approval dated 7 June 2022.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Conflicts of Interest: The authors declare no conflict of interest.
Abbreviations

FPR    False positive rates
FNR    False negative rates
ROC    Receiver operating characteristics
FL     Fuzzy logic
UAO    Unified An-Or
BERT   Bidirectional encoder representations from transformers
LSTM   Long short term memory
NB     Naïve Bayes
KNN    K-nearest neighbors
DNN    Deep neural network
SVM    Support vector machine
LSVM   Linear SVM
PLSV   Poly SVM
SSVM   Sigmoid SVM
RF     Random forest
WE     Word Embedding
LR     Logistic regression
GNB    Gaussian Naïve Bayes
DT     Decision Tree
SVC    Support vector classifier
CNN    Convolutional neural network
NN     Neural Network
MNB    Multinomial Naïve Bayes
BNB    Bernoulli Naïve Bayes
DL     Deep learning
LTC    Logistic tree classifier
RNN    Recurrent neural networks
MCDM   Multi-criteria decision-making
TP     True positive
TN     True negative
FP     False positive
FN     False negative
OWA    Ordered weighted average

References

1. Feng, W.; Sun, J.; Zhang, L.; Cao, C.; Yang, Q. A support vector machine based naive Bayes algorithm for spam filtering. In Proceedings of the 2016 IEEE 35th International Performance Computing and Communications Conference (IPCCC), Las Vegas, NV, USA, 9–11 December 2016; pp. 1–8.
2. Dada, E.G.; Bassi, J.S.; Chiroma, H.; Adetunmbi, A.O.; Ajibuwa, O.E.; Ajibuwa, O.E. Machine learning for email spam filtering: review, approaches and open research problems. Heliyon 2019, 5, e01802. [CrossRef]
3. Available online: https://www.statista.com/statistics/456500/daily-number-of-e-mails-worldwide/ (accessed on 11 April 2022).
4. Fonseca, O.; Fazzion, E.; Cunha, I.; Las-Casas, P.H.B.; Guedes, D.; Meira, W.; Hoepers, C.; Steding-Jessen, K.; Chaves, M.H. Measuring, characterizing, and avoiding spam traffic costs. IEEE Internet Comput. 2016, 20, 16–24. [CrossRef]
5. Park, I.; Sharman, R.; Rao, H.R.; Upadhyaya, S. The effect of spam and privacy concerns on e-mail users’ behavior. J. Inf. Syst. Secur. 2007, 3, 39–62.
6. Ogwu, S.; Sice, P.; Keogh, S.; Goodlet, C. An exploratory study of the application of mindsight in email communication. Heliyon 2020, 6, e04305. [CrossRef]
7. Cook, D.; Hartnett, J.; Manderson, K.; Scanlan, J. Catching spam before it arrives: domain specific dynamic blacklists. In Proceedings of the 2006 Australasian workshops on Grid Computing and E-Research-Volume 54, Hobart, Australia, 16–19 January 2006; Australian Computer Society: Darlinghurst, Australia, 2006; pp. 193–202.
8. Kshirsagar, D.; Patil, A. Blackhole attack detection and prevention by real time monitoring. In Proceedings of the 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Tiruchengode, India, 4–6 July 2013; pp. 1–5.
9. Wang, B.; Pan, W. A survey of content-based anti-spam email filtering. J. Chin. Inf. Process. 2005, 5.
10. Yaseen, Q.; et al. Spam Email Detection Using Deep Learning Techniques. *Procedia Comput. Sci.* 2021, 184, 853–858.

11. Islam, M.K.; Al Amin, M.; Islam, M.R.; Mahbub, M.N.I.; Showrov, M.I.H.; Kaushal, C. Spam-Detection with Comparative Analysis and Spawning Words Extractions. In Proceedings of the 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 3–4 September 2021; pp. 1–9.

12. Siddique, Z.B.; Khan, M.A.; Din, I.U.; Almogren, A.; Mohiuddin, I.; Nazir, S. Machine Learning-Based Detection of Spam Emails. *Sci. Program.* 2021, 2021, 6508784. [CrossRef]

13. Sheneamer, A. Comparison of Deep and Traditional Learning Methods for Email Spam Filtering. *Int. J. Adv. Comput. Sci. Appl.* 2021, 12, 560–565. [CrossRef]

14. Mallampati, D.; Hegde, N.P. A Machine Learning Based Email Spam Classification Framework Model: Related Challenges and Issues. *Int. J. Innov. Technol. Explor. Eng.* 2020, 9, 3137–3144. [CrossRef]

15. Srinivasan, S.; Ravi, V.; Alazab, M.; Ketha, S.; Al-Zoubi, A.M.; Kotti Padannayil, S. Spam emails detection based on distributed word embedding with deep learning. In *Machine Intelligence and Big Data Analytics for Cybersecurity Applications*; Springer: Berlin/Heidelberg, Germany, 2021; pp. 161–189.

16. Kumar, N.; Sonowal, S. Email spam detection using machine learning algorithms. In Proceedings of the 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 15–17 July 2020; pp. 108–113.

17. Anitha, P.; Rao, C.G.; Babu, D.S. Email Spam Filtering Using Machine Learning Based Xgboost Classifier Method. *Turk. J. Comput. Math. Educ.* 2021, 12, 2182–2190.

18. Sethi, M.; Chandra, S.; Chaudhary, V.; Yash. Email Spam Detection using Machine Learning and Neural Networks. *Int. Res. J. Eng. Technol.* 2021, 8, 349–355.

19. Bagui, S.; Nandi, D.; Bagui, S.; White, R.J. Machine Learning and Deep Learning for Phishing Email Classification using One-Hot Encoding. *J. Comput. Sci.* 2021, 17, 610–623. [CrossRef]

20. Nayak, R.; Jiwani, S.A.; Rajitha, B. Spam email detection using machine learning algorithm. *Mater. Today Proc.* 2021. [CrossRef]

21. Euna, N.J.; Hossain, S.M.M.; Anwar, M.M.; Sarker, I.H. Content-based Spam Email Detection Using N-gram Machine Learning Approach. *Preprints 2021, 2021100236.* [CrossRef]

22. Chakraborty, S.; Mondal, B. Spam mail filtering technique using different decision tree classifiers through data mining approach—a comparative performance analysis. *Int. J. Comput. Appl.* 2012, 47, 26–31. [CrossRef]

23. Rusland, N.F.; Wahid, N.; Kasim, S.; Hafit, H. Analysis of Naïve Bayes algorithm for email spam filtering across multiple datasets. In *IOP Conference Series: Materials Science and Engineering*; IOP Publishing: 2017, Bristol, UK; Volume 226, p. 012091.

24. Bibi, A.; Latif, R.; Khalid, S.; Ahmed, W.; Shabir, R.A.; Shahryar, T. Spam mail scanning using machine learning algorithm. *J. Comput. 2020, 15, 73–84. [CrossRef]

25. Guo, Z.; Yu, K.; Jolfaei, A.; Ding, F.; Zhang, N. Fuz-spam: label smoothing-based fuzzy detection of spammers in internet of things. *IEEE Trans. Fuzzy Syst.* 2021. [CrossRef]

26. Iqbal, K.; Khan, S.A.; Anisa, S.; Tasneem, A.; Mohammad, N. A Preliminary Study on Personalized Spam E-mail Filtering Using Bidirectional Encoder Representations from Transformers (BERT) and TensorFlow 2.0. *Int. J. Comput. Digit. Syst.* 2022, 11, 893–903 [CrossRef]

27. Kaddoura, S.; Alfandi, O.; Dahmani, N. A spam email detection mechanism for english language text emails using deep learning approach. In Proceedings of the 2020 IEEE 29th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Virtual, 10–13 September 2020; pp. 193–198.

28. Zamir, A.; Khan, H.U.; Mehmoood, W.; Iqbal, T.; Akram, A.U. A feature-centric spam email detection model using diverse supervised machine learning algorithms. *Electron. Libr.* 2020, 38, 633–657. [CrossRef]

29. Sakkis, G.; Andrououtsopoulos, I.; Palouras, G.; Karkaletis, V.; Spyropoulos, C.D.; Stamatopoulos, P. Stacking classifiers for anti-spam filtering of e-mail. In *Empirical Methods in Natural Language Processing*; Carnegie Mellon University: Pittsburgh, PA, USA, 2001; pp. 44–50. [CrossRef]

30. Attar, A.; Rad, R.M.; Atani, R.E. A survey of image spamming and filtering techniques. *Artif. Intell. Rev.* 2013, 40, 71–105. [CrossRef]

31. Zhang, L.; Zhu, J.; Yao, T. An evaluation of statistical spam filtering techniques. *ACM Trans. Asian Lang. Inf. Process. (TALIP)* 2004, 3, 243–269. [CrossRef]

32. Available online: [https://www.cs.cmu.edu/~enron/](https://www.cs.cmu.edu/~enron/) (accessed on 22 December 2021).

33. Koprinska, I.; Poon, J.; Clark, J.; Chan, J. Learning to classify email. *Inf. Sci.* 2007, 177, 2167–2187. [CrossRef]

34. Cormack, G.V.; Lynam, T.R. Online supervised spam filter evaluation. *ACM Trans. Inf. Syst.* 2007, 25, 11. [CrossRef]

35. Andrououtsopoulos, I.; Koutsihas, J.; Chandrinos, K.V.; Palouras, G.; Spyropoulos, C.D. An evaluation of naive bayesian anti-spam filtering. In Proceedings of the 11th European Conference on Machine Learning (ECML 2000), Barcelona, Spain, 31 May–2 June 2000; Springer: Berlin/Heidelberg, Germany, 2000; pp. 9–17.

36. Andrououtsopoulos, I.; Palouras, G.; Karkaletis, V.; Sakkis, G.; Spyropoulos, C.D.; Stamatopoulos, P. Learning to filter spam e-mail: A comparison of a naive bayesian and a memory-based approach. In Proceedings of the 4th European Conference on Principles and Practice of Knowledge Discovery in Databases, Pisa, Italy, 20–24 September 2000; Springer: Berlin/Heidelberg, Germany, 2000; pp. 1–12.

37. DeBarr, D.; Wechsler, H. Spam detection using clustering, random forests, and active learning. In Proceedings of the Sixth Conference on Email and Anti-Spam, Mountain View, CA, USA, 16–17 July 2009; pp. 1–6.
38. Available online: http://www.aueb.gr/users/ion/data/lingspam_public.tar.gz (accessed on 17 December 2021).
39. Available online: http://www.aueb.gr/users/ion/data/PU123ACorpora.tar.gz (accessed on 19 December 2021).
40. Novo-Lourés, M.; Ruano-Ordás, D.; Pavón, R.; Laza, R.; Gómez-Meire, S.; Méndez, J.R. Enhancing representation in the context of multiple-channel spam filtering. *Inf. Process. Manag.* 2022, 59, 102812. [CrossRef]
41. Occhipinti, A.; Rogers, L.; Angione, C. A pipeline and comparative study of 12 machine learning models for text classification. *Expert Syst. Appl.* 2022, 201, 117193. [CrossRef]
42. Guo, Z.; Tang, L.; Guo, T.; Yu, K.; Alazab, M.; Shalaginov, A. Deep graph neural network-based spammer detection under the perspective of heterogeneous cyberspace. *Future Gener. Comput. Syst.* 2021, 117, 205–218. [CrossRef]
43. Venkateswarlu, B.; Shenoi, V. Optimized generative adversarial network with fractional calculus based feature fusion using Twitter stream for spam detection. *Inf. Secur. J. Glob. Perspect.* 2021. [CrossRef]
44. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* 2018, arXiv:1810.04805.
45. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* 1997, 9, 1735–1780. [CrossRef]
46. Zadeh, L.A. Information and control. *Fuzzy Sets* 1965, 8, 338–353.
47. Rehman, S.; Khan, S.A.; Alhems, L.M. A rule-based fuzzy logic methodology for multi-criteria selection of wind turbines. *Sustainability* 2020, 12, 8467. [CrossRef]
48. Rehman, S.; Khan, S.A.; Alhems, L.M. Application of TOPSIS approach to multi-criteria selection of wind turbines for on-shore sites. *Appl. Sci.* 2020, 10, 7595. [CrossRef]
49. Khan, S.A.; Engelbrecht, A.P. A new fuzzy operator and its application to topology design of distributed local area networks. *Inf. Sci.* 2007, 177, 2692–2711. [CrossRef]
50. Khan, S.A. Design and Analysis of Evolutionary and Swarm Intelligence Techniques for Topology Design of Distributed Local Area Networks. Ph.D. Thesis, University of Pretoria, Pretoria, South Africa, 2009.
51. Khan, S.A. A STRIDE Model based Threat Modelling using Unified and-Or Fuzzy Operator for Computer Network Security. *Int. J. Comput. Netw. Technol.* 2017, 5, 13–20. [CrossRef]
52. Khan, S.A. Fuzzy preferences based STRIDE threat model for network intrusion detection. *Int. J. Comput. Netw. Technol.* 2017, 5, 107–111. [CrossRef]
53. Mohiuddin, M.A.; Khan, S.A.; Engelbrecht, A.P. Simulated evolution and simulated annealing algorithms for solving multi-objective open shortest path first weight setting problem. *Appl. Intell.* 2014, 41, 348–365. [CrossRef]
54. Mohiuddin, M.A.; Khan, S.A.; Engelbrecht, A.P. Fuzzy particle swarm optimization algorithms for the open shortest path first weight setting problem. *Appl. Intell.* 2016, 45, 598–621. [CrossRef]
55. Zarghami, M.; Szidarovszky, F. Fuzzy quantifiers in sensitivity analysis of OWA operator. *Comput. Ind. Eng.* 2008, 54, 1006–1018. [CrossRef]
56. Hu, H.; Li, G. Granular risk-based design optimization. *IEEE Trans. Fuzzy Syst.* 2014, 23, 340–353. [CrossRef]
57. Gao, Q.; Cai, X.; Zhu, J.; Guo, X. Multi-objective optimization and fuzzy evaluation of a horizontal axis wind turbine composite blade. *J. Renew. Sustain. Energy* 2015, 7, 063109. [CrossRef]
58. Bhowmick, A.; Hazarika, S.M. E-mail spam filtering: A review of techniques and trends. *Adv. Electron. Commun. Comput.* 2018, 443, 583–590. [CrossRef]
59. Baeldung. Available online: https://www.baeldung.com/cs/spam-filter-training-sets (accessed on 28 May 2022).