JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

# E-Mail Spam Filtring Using Machine Learning Classification Technique

**Neetu Ahirwar**
M. Tech Scholar
Department of CSE/IT
Patel College of Science & Technology, Indore
ahirwarneetu96@gmail.com

**Reshma Shivhare**
Designation - HOD
Department of CSE/IT
Patel College of Science & Technology, Indore
reshma.rai7@gmail.com

**Abstract:** Single-modal spam filtering systems have a high text spam detection rate. Spammers introduce garbage information into the multi-modality component of an email to diminish the single-modal spam filtering systems' identification rate and evade detection. A novel model dubbed text-based dataset modal architecture based on model fusion (MMA-MF) is suggested to successfully filter spam buried in text. The spam-filtering model combines a CNN and LSTM. Using the LSTM model and the CNN model to evaluate email text independently, two classification probability values are obtained and combined to determine whether an email is spam. For the MMA-MF model's hyperparameters, we apply grid search optimization and k-fold cross-validation to assess its performance. Our experiments demonstrate that our model is more accurate than typical spam filtering systems (92.64–98.48%).

**Keywords:** Spam Filtering System; Multi-Modal; MMA-MF; Fusion Model; LSTM; CNN

## I. INTRODUCTION

A message sent by email that is considered to be spam is one that includes unsolicited mail [1]. Email is becoming an increasingly popular method of communication among Internet users as a result of the fast growth of the Internet. At the same time, the problem of spam is becoming worse, and the majority of messages sent in spam are sent with the intention of asking the receivers for money. In order to accomplish this goal, the company sells items that make amazing claims about their ability to treat a variety of illnesses, including diabetes, obesity, and hair loss. They might be of any kind, such as an advertising, a text email, an email including an image, or even an email that includes both text and picture data.

The annual average percentage of worldwide spam in total emails was as high as 56.63 percent or more in 2017, according to the spam analysis report published by Kaspersky Lab, a well-known firm operating in the area of information security [2]. This occurrence is an indication that spam is overwhelming the whole network, which causes cyber citizens to experience a degree of annoyance. Single-modal spam filtering systems have a high detection rate for text spam and image spam. However, spammers may insert junk information into the multi-modal part of an email, which we call hybrid spam, to reduce the detection rate of single-modal spam filtering systems, ultimately accomplishing their goal of evading detection. In order to avoid detection, spammers may insert junk information into the multi-modal part of an email, which we call hybrid spam. Because it contains more information than traditional spam and because it requires more network bandwidth and storage space for forwarding and delivery to mailbox servers, hybrid spam is more harmful than traditional spam. The reason for this is that hybrid spam contains more information than traditional spam. In addition, viruses or unwanted information delivered by hybrid spam are more difficult to identify, which presents enormous information security concerns to the communication of individuals. Therefore, it is of the utmost importance to acquire the skills necessary to correctly recognise hybrid spam.

Anti-spam strategies have been the subject of research for many years [3–15] within the areas of machine learning and cybersecurity. These techniques may be loosely grouped into three groups, which are as follows: (1) text-based spam detection; (2) image-based spam detection; and (3) multi-modal spam detection. The primary criteria used by the first and second categories, when determining whether an email is spam or not, are its textual content and its visual content, respectively. On the other hand, the last category scans both the textual and visual content of an email in order to identify and eliminate spam.

The suggested approach examines the content of an email's text in order to determine whether or not it contains garbage material and then filters it accordingly. To put it another way, one of the primary benefits of the MMA-MF model is that it is able to filter not just hybrid spam but also spam that contains solely text data. Based on the findings of the experiments, it seems that our approach is much superior to that of our competitors. The primary addition that we provide is that we use the CNN and LSTM models to process the text data that is found in an email, and then we merge these models using the logistic regression approach to create a fusion model. This is the first time, to the best of our knowledge, that we have cast light on this strategy in the email filtering systems.

The remaining parts of this essay are structured as follows: In Section 2, we discuss the design framework of the CNN, LSTM, and fusion model, which is a condensed version of the classification method for text spam. This section also covers the architecture of the MMA-MF model. The assessment metrics and validation methodologies will be presented in Section 3. The experimental findings and subsequent discussion are presented in Section 4. Section 5 is where the findings are presented at the very end.

## II. MMA-MF MODEL ARCHITECTURES

In its most fundamental form, the process of filtering spam involves a binary classification issue. We propose a kind of spam filtering framework that we refer to as MMA-MF. This will allow our model to not only filter hybrid spam, but also filter spam that consists only of text data. Figure 1 presents this organisational structure.

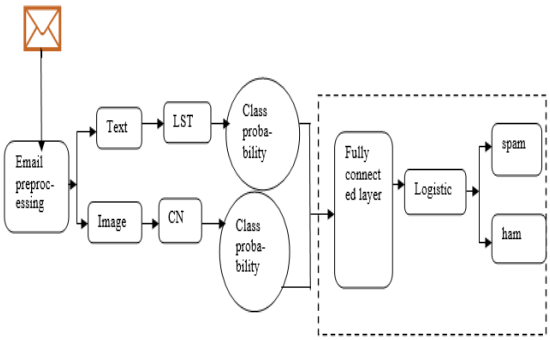


**Figure 1. MMA-MF Model Architecture.**

The following is a description of the particular procedures involved in the MMA-MF model for identifying spam:

1. In order to retrieve the text dataset, one must first do email preprocessing, which entails separating the text data from an email.

2. Getting the best possible classifiers: Using the text dataset, we train and tune the LSTM model and the CNN model, respectively, and we end up with the best possible LSTM model and the best possible CNN model.

3. Re-entering the data into the optimum CNN model in order to acquire the classification probability values of the as spam requires that the data be re-entered. In a similar fashion, the text dataset is re-entered into the best LSTM model so that the classification probability values of the text dataset as spam may be obtained. We utilise the dropout ideology to determine that the matching model output probability value for an email that just contains text data should be set to 0.5.

4. Obtaining the best possible fusion model: The two classification probability values are input into the fusion model so that it may be trained and optimised, eventually leading to the acquisition of the best possible fusion model.

The above explanations show how, by following steps 1, 3, and 4, it is possible to calculate the chance that a newly received email is spam. This is true regardless of whether the newly received email is of the hybrid or single-modal kind. In conclusion, we provide the overarching structure of the MMF-MA model as well as the concise procedures for determining the likelihood value of classifying an email as spam. Following this, we will go into depth about the internal structure of the LSTM model, the CNN model, and the fusion model, as well as the process of selecting the ideal hyperparameter values for each of the three models.

### 2.1 Text Classification Model: LSTM Model

Figure 2 provides a general representation of the LSTM model's internal structure. It is made up of three layers: one with a single word embedded in it, two LSTM layers, and one fully connected (FC) layer. The following is a list of the procedures that need to be taken when processing the text component of an email in order to acquire the categorization probability value of the email: acquiring the text data of an email by first using the preprocessing approach, and then utilising the word embedding technique in order to get the email's word vector representation. In this article, the word vector representation was obtained via the use of the word2vec toolbox. Following this step, we apply the two LSTM layers that we built in order to automatically extract features from the text input. In the end, we use the FC layer with the Softmax activation function to determine the probability value of classifying the text data as spam. The LSTM model is trained and optimised by making use of the log-likelihood function to minimise the loss function. [22].

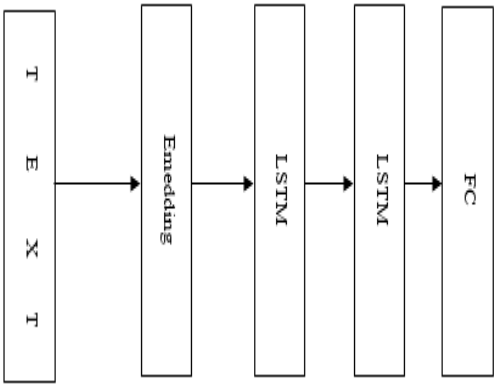


**Figure 2.** LSTM model framework.

We utilise the grid search optimization method to determine the best possible values for the LSTM model's five hyperparameters, which are the learning rate, batch size, epochs, dropout rate, and optimization technique. These five hyperparameters are responsible for the model's overall performance. Table 1 displays the range of values for these hyperparameters, as well as the values that the LSTM model determined to be optimum.

**Table 1. The range and optimal values of hyperparameters for LSTM.**

| Hyperparameter | Range | Optimal Value |
|---|---|---|
| learning rate | [0.001, 0.01, 0.1, 0.2] | 0.001 |
| batch size | [8, 16, 32] | 32 |
| epochs | [10, 20, 30] | 30 |
| dropout rate | [0.2, 0.3, 0.4] | 0.3 |
| optimization algorithm | [SGD [23], RMSprop [24], Adam [25]] | Adam |

In this section, we will provide a quick overview of the LSTM model using pseudo code. Please refer to the published material [10,26] for a more in-depth algorithm on the LSTM unit. Let's refer to the textual content of an email as T. Input T into the embedding stage in order to transform T into becoming a word vector x, where x = (x1,x2,• • •,xl), where xiRn is the n-dimensional word vectors for the i-th word in the document T and matrix xRlxn denotes the document T, where l is the maximum length of and l is less than 500. Following are the equations that are used to bring the memory ct and the hidden state ht up to date at each time step t.:

$$\begin{bmatrix} i_t \\ f_t \\ o_t \\ \hat{c}_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ tanh \end{bmatrix} W \cdot [h_{t-1}, x_t], \quad (1)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \hat{C}_t \quad (2)$$

$$h_t = 0_t \odot tanh(c_t) \quad (3)$$

where $x_t$ is the input at the current time-step, *i*, *f* and *o* is the input gate activation, forget gate activation and output gate activation, respectively, $\hat{c}_t$ is the current cell state, $\sigma$ denotes the logistic sigmoid function and Ⓢ denotes element-wise multiplication. Through training and optimizing the LSTM model, we could obtain the classification probability value of the text part as spam. The entire process of text spam classification algorithm is described in Algorithm 1.

**Algorithm 1** Text Spam Classification Algorithm.

**Input:** Text Document T

**Output:** Text spam classification probability value *e*

**1:** Input *T* into the word2vec toolkit to get the word vector *x*, *x*= ($x_1$,$x_2$,

**2:** For the first LSTM layer (64 LSTM units), input *x* at time *t* and complete the following calculations:

$$\begin{bmatrix} i_t \\ f_t \\ o_t \\ \hat{c}_t \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ tanh \end{bmatrix} W \cdot [h_{t-1}, x_t],$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \hat{C}_t$$

$$h_t = 0_t \odot tanh(c_t)$$

**3:** By the first LSTM layer, getting the text feature vector $h = (h_1,h_2,$

**4:** For the second LSTM layer(32 LSTM units), input *h* at time t and do the same as Equations (1)–(3).

Finally, getting more abstract text feature vector *k*, $k = (k_1,k_2,$

**5:** Input *k* to FC layer and using Softmax activation function to gain the text classification probability value *e*;

**6: return** ;

Input sequences in the form of phrases are combined with the results of the preceding LSTM unit before being introduced into the LSTM unit. This process is repeated with each new phrase that is supplied, and as a result, the LSTM units are able to continue preserving the essential characteristics. The number of LSTM units that store the most relevant characteristics is the variable in question. Therefore, by using the LSTM layer, the FC layer, and the Softmax activation function, we are able to get the classification probability value e for the text portion as spam.

### 2.2 Fusion Model

Figure 3 provides an illustration of the structure of the fusion model. In order to acquire the most accurate classification probability value of the email as spam, the goal is to combine the classification probability value of one email text part with the classification probability value of the same email text part. The following is a rundown of the overall steps: 1. Combining the two classification probability values from the LSTM and CNN models to get a feature vector with the notation q, where q is the length of the R1-space; 2. Inputting q into the FC layer, which consists of 64 neurons, in order to get a complete feature vector; 3. Inputting the complete feature vector into the logistic layer, which consists of two neurons and selects the logistic regression function as the activation function in order to get the most accurate classification probability value of the email being spam. We only use the grid search optimization algorithm to select the optimal values for the four hyperparameters, which are learning rate, batch size, epochs and optimization algorithm. The best hyperparameter for learning rate is equal to 0.01, batch size is equal to 16, epochs is equal to 30, and the optimization algorithm is the SGD algorithm. Taking into account the efficacy of our machine, we only use the grid search optimization algorithm to select the optimal values for the four hyperparameters.

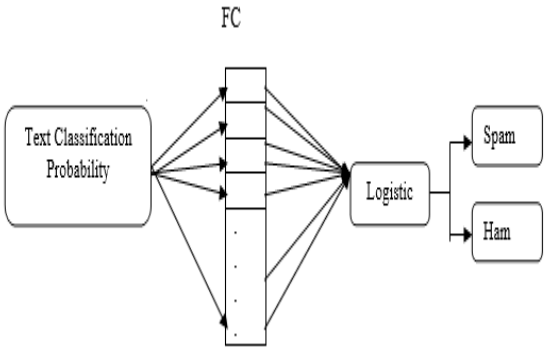


**Figure 3.** Fusion model structure.

Suppose that the classification probability dataset input to the fusion model is $D=\{(q_1,y_1), (q_2,y_2), \cdots, (q_v,y_v)\}$, $q_i \in R^{1\times 4}$, $y_i \in \{0,1\}$, in which the conditional probability distribution of the logistic regression function is as follows:

$$P(Y = 1|q) = \pi(q) \quad \frac{e^{-w^T.q}}{1+e^{-w^T.q}} \qquad (4)$$

$$P(Y = 0|q) = 1 - \pi(q) \quad \frac{1}{1+e^{-w^T.q}} \qquad (5)$$

We choose the log-likelihood function as the loss function, and the formula is as follows:

$$L(w) = \sum_{i=1}^{v} [y_i \log \pi(q_i) + (1 - y_i)\log(1 - \pi(q_i))]$$

$$\sum_{i=1}^{v} [y_i \, log \frac{\pi(q_i)}{1-\pi(q_i)} \log(1 - \pi(q_i))]$$

$$\sum_{i=1}^{v} [y_i(w \cdot q_i) - \log(1 + e^{(w \cdot q_i)})] \qquad (6)$$

The maximum value of $L(w)$ is obtained by the Adam algorithm. In addition, the optimal estimate value of the parameter $w$ can be obtained by optimizing $L(w)$. If $p > 0.5$, it means that the email is spam; otherwise, it is a normal email.

## III. EVALUATION METRICS AND VALIDATION SCHEME

### 3.1. Evaluation Metrics

In order to assess the effectiveness of the proposed method, different evaluation indicators have been used, including accuracy, recall, precision and f1-score, which are defined as follows:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN'} \qquad (7)$$

$$Recall = \frac{TP}{TP+FN'} \qquad (8)$$

$$Precision = \frac{TP}{tp+FP'} \qquad (9)$$

$$F1 - Score = \frac{2*(Precision*Recall)}{precision+Recall} \qquad (10)$$

The specific meanings of FP, FN, TP and TN are defined as follows:

- False Positive (FP): The number of legitimate emails (Ham) that are misclassified;
- False Negative (FN): The number of misclassified spam;
- True Positive (TP): The number of spam that are correctly classified;
- True Negative (TN): The number of legitimate emails (Ham) that are correctly classified.

For spam detection, the evaluation metrics about accuracy, recall, precision and f1-score are mainly based on the confusion matrix, which shows in Table 3:

**Table 3.** Confusion matrix.

| Prediction | Actual | |
|---|---|---|
| | Spam | Ham |
| Spam | TP | FN |
| Ham | FP | TN |

### 3.2 Validation Scheme

In prior research, a spam filtering system was put through its paces by using a rejection verification technique to measure how well it performed its function. In several studies, the training dataset is used to assess the performance of a model, while the testing dataset is used to get the accuracy of the model that was determined to be the ideal choice. The percentages of the training-test split utilised for data distribution vary from study to study. The hold out approach is the one that splits the dataset in half, with one half being used for training and the other being used for testing. This is the method that is the simplest and most straight-forward. The assessment suffers from the flaw that its accuracy is heavily dependent on the samples that are ultimately included in each collection. The k-fold cross-validation technique is an additional approach that may be used to lower the variance of the hold out method. In the k-fold cross-validation method, the dataset M is partitioned into k parts that are mutually exclusive from one another, and these parts are labelled M1,M2,• • •,Mk. After been trained on Mi/M, the inducer is next evaluated against Mi. This process is carried out k times with varying I where I may take the values 1, 2, • • •, and k. The following are the definitions of accuracy, recall, precision, and f1-score in relation to a k-fold test:

$$Accuracy = \sum_{i=1}^{n} Accuracy_i \qquad (11)$$

$$Recall = \sum_{i=1}^{k} Recall_i \qquad (12)$$

$$Precision = \sum_{i=1}^{k} Precision_i \qquad (13)$$

$$F1 - Score = \sum_{i=1}^{k} F1 - Score_i \qquad (14)$$

where $Accuracy_i$, $Recall_i$, $Precision_i$ and $F1-Score_i$ are the accuracy, recall, precision and f1-score for each of the k tests. Considering the performance of our computer, we choose a 5-fold cross-validation method throughout the experiments.

## IV. EXPERIMENTAL RESULTS AND DISCUSSION

### 4.1. Corpus

In this study, we opted to conduct our tests using three distinct varieties of email datasets: a dataset that only included text, a dataset that only contained text, and a dataset that includes text data. Each of these datasets is described in more detail below. The dataset that simply contains text originates from the Indian corpus [29], and out of 33,645 text emails, we only chose 6000 of them to analyse (4500 spam and 1500 ham). This was accomplished by deleting duplicates and picking emails at random. The sole dataset that contains text is called Personal text Ham, dataset 1, and it contains all of the data. The specifics of the datasets that were utilised in the tests are detailed in Table 4, which can be seen below.

**Table 4.** Datasets used in Experiments.

| Type | Original Dataset | Before Remove Duplicates | After Remove Duplicates |
|---|---|---|---|
| Text | Indian Ham | 17,108 | 1500 |
| | Indian Spam | 16,537 | 4500 |

For the mixed dataset 1, the number of text dataset, which contains 600 Spam (text Spam 600) and 600 Ham (text Ham 600 and tex Ham 600 are formed into 600 Ham email).

**Table 5.** Training and Testing Dataset Size.

| Type | Training Dataset Size | Testing Dataset Size |
|---|---|---|
| Text Dataset 1 | 5000 | 1000 |
| Text Dataset 2 | 960 | 240 |

### 4.2. Results and Discussion

In this part of the article, we will provide the findings of our assessment about the text spam classification, the text spam classification, and the mixed spam classification. In addition, we provide some analyses as well as some comments about the findings of the experiment.

We use the 5-fold cross-validation method to verify the performance of the MMA-MF model on the text dataset and the mixed datasets 1, and we obtain the experimental results of the MMA-MF model on the four datasets, as shown in Table 6, in which u means the average value of Accuracy, Recall, F 1-Score or Precision after using the 5-fold cross-validation method. Table 6: Experimental results of the MMA-MF model on the four datasets.

**Table 6. Experimental results in 5-fold cross-validation for the MMA-MF model.**

| Fold | Accuracy | Recall | F1-Score | Precision |
|---|---|---|---|---|
| MMA-MF Model for Text Dataset 1 | | | | |
| 1 | 98.42 | 97.84 | 97.24 | 98.5 |
| 2 | 98.67 | 98.15 | 97.47 | 98.5 |
| 3 | 98.67 | 98.19 | 97.65 | 99 |
| 4 | 98.25 | 97.71 | 97.27 | 98 |
| 5 | 98.42 | 97.89 | 97.53 | 98.5 |
| MMA-MF Model for Text Dataset 2 | | | | |
| 1 | 93.35 | 92.64 | 92.89 | 90.5 |
| 2 | 92.56 | 92.63 | 92.75 | 90.01 |
| 3 | 91.5 | 92.33 | 91.83 | 93.5 |
| 4 | 92.35 | 92.83 | 92.97 | 92 |
| 5 | 93.44 | 92.72 | 92.71 | 92.5 |

Based on Table 6, we can draw the conclusion that the MMA-MF model that was built for this study implements the filtering function of spam. This means that regardless of whether the spam is buried in the text or concealed in the text, we are all able to manage it and filter it out quite effectively. In conclusion, the following are some observations for us to make: Not only does it filter emails that are properly mixed, but it also filters emails that are just text for the MMA-MF model.

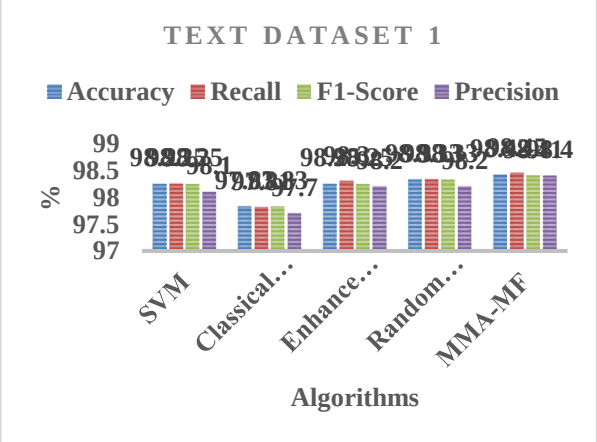


**Figure 4. 5-Fold Cross-Validation Chart for Text Dataset 1**

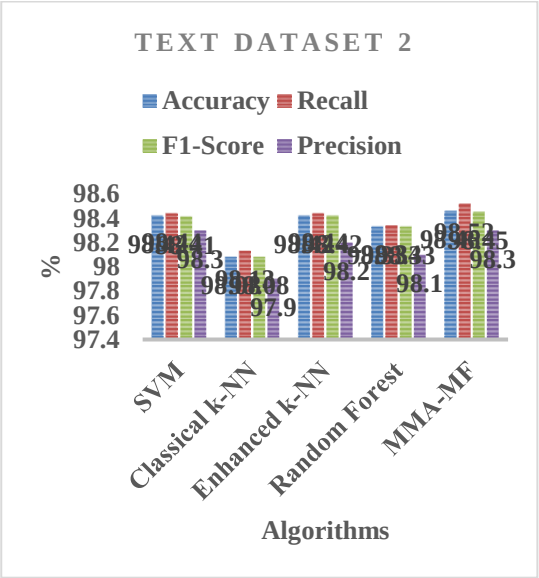


**Figure 5. 5-Fold Cross-Validation Chart for Text Dataset 1**

## V. CONCLUSIONS

We present the MMF-MF multi-modal fusion architecture. The model integrates the CNN, LSTM network, and logistic regression to enhance spam detection in a variety of email forms. Other models can only handle text-based spam, but our model can also filter hybrid spam.

Future work must address two concerns. (1) Table 5 shows our experimental dataset is balanced. Practically, spam detection datasets have a considerable disparity between spam and non-spam emails. (1) Because there is no true mixed email dataset for public use, the dataset is spliced.

In the future, we hope to use the new technique like the one-class classification method and a few-shot learning method to solve the discrepancy between spam and non-spam emails, and we will continue to collect more realistic mixed email datasets to improve our model's network structure and spam detection performance.

## References

1 Seth, S.; Biswas, S. Multimodal Spam Classification Using Deep Learning Techniques. In Proceedings of the 2017 13th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), Jaipur, India, 4–7 December 2017; pp. 346–349.

2 Bettencourt, J. Kaspersky Lab Spam and Phishing Report: FIFA 2018 and Bitcoin among 2017's Most Luring Topics. Available online: https://usa.kaspersky.com/about/press-releases/2018_fifa-2018-and-bitcoin-among-2017-most-luring-topics (accessed on 15 February 2018).

3 Carreras, X.; Marquez, L. Boosting trees for anti-spam email filtering. arXiv **2001**, arXiv:cs/0109015.

4 Androutsopoulos, I.; Paliouras, G.; Michelakis, E. Learning to Filter Unsolicited Commercial E-Mail; DEMOKRITOS; National Center for Scientific Research: Paris, French, 2014.

5 Sahami, M.; Dumais, S.; Heckerman, D.; Horvitz, E. A Bayesian approach to filtering junk e-mail. In Learning for Text Categorization: Papers from the 1998 Workshop; AAAI Technical Report WS-98-05; Monona Terrace Convention Center: Madison, WI, USA, 1998; Volume 62, pp. 98–105.

6 Anayat, S.; Ali, A.; Ahmad, H.F. Using a probable weight based Bayesian approach for spam filtering. In Proceedings of the 8th International Multitopic Conference 2004, Lahore, Pakistan, 24–26 December 2004; pp. 340–345.

7 Kim, H.J.; Shrestha, J.; Kim, H.N.; Jo, G.S. User action based adaptive learning with weighted bayesian classification for filtering spam mail. In Australasian Joint Conference on Artificial Intelligence; Springer: Berlin/Heidelberg, Germany, 2006; pp. 790–798.

8 Yang, Z.; Nie, X.; Xu, W.; Guo, J. An approach to spam detection by naive Bayes ensemble based on decision induction. In Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA'06), Jinan, China, 16–18 October 2006; Volume 2, pp. 861–866.

9 Androutsopoulos, I.; Paliouras, G.; Karkaletsis, 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. arXiv 2000, arXiv:cs/0009009

10 Jain, G.; Sharma, M.; Agarwal, B. Optimizing semantic LSTM for spam detection. Int. J. Inf. Technol. 2019, 11, 239–250. [CrossRef]

11 Abi-Haidar, A.; Rocha, L.M. Adaptive spam detection inspired by a cross-regulation model of immune dynamics: A study of concept drift. In International Conference on Artificial Immune Systems; Springer: Berlin/Heidelberg, Germany, 2008; pp. 36–47.

12 Shang, E.X.; Zhang, H.G. Image spam classification based on convolutional neural network. In Proceedings of the 2016 International Conference on Machine Learning and Cybernetics (ICMLC), Jeju, South Korea, 10–13 July 2016; Volume 1, pp. 398–403.

13 Wang, Z.; Josephson, W.K.; Lv, Q.; Charikar, M.; Li, K. Filtering Image Spam with Near-Duplicate Detection; CEAS: Mountain View, CA, USA, 2007.

14 Kumar, P.; Biswas, M. SVM with Gaussian kernel-based image spam detection on textual features. In Proceedings of the 2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT), Ghaziabad, India, 9–10 February 2017; pp. 1–6.

15 Xu, C.; Chiew, K.; Chen, Y.; Liu, J. Fusion of text and image features: A new approach to image spam filtering. In Practical Applications of Intelligent Systems; Springer: Berlin/Heidelberg, Germany, 2011; pp. 129–140.

16 Huamin, F.; Xinghua, Y.; Biao, L.; Chao, J. A spam filtering method based on multi-modal features fusion. In Proceedings of the 2011 Seventh International Conference on Computational Intelligence and Security, Hainan, China, 3–4 December 2011; pp. 421–426.

17 Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems; Curran Associates, Inc.: Red Hook, NY, USA, 2012; pp. 1097–1105.

18 Graves, A. Long short-term memory. In Supervised Sequence Labelling with Recurrent Neural Networks; Studies in Computational Intelligence; Springer: Berlin/Heidelberg, Germany, 2012; pp. 37–45.

19 D. Bhuriya, G. Kaushal, A. Sharma, and U. Singh, "Stock market predication using a linear regression," in *Proceedings of the International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017*, 2017, vol. 2017-Janua, doi: 10.1109/ICECA.2017.8212716.

20 V. Prakaulya, R. Sharma, U. Singh, and R. Itare, "Railway passenger forecasting using time series decomposition model," in *Proceedings of the International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017*, 2017, vol. 2017-Janua, doi: 10.1109/ICECA.2017.8212725.

21 R. Yadav, A. Choorasiya, U. Singh, P. Khare, and P. Pahade, "A Recommendation System for E-Commerce Base on Client Profile," in *Proceedings of the 2nd International Conference on Trends in Electronics and Informatics, ICOEI 2018*, 2018, doi: 10.1109/ICOEI.2018.8553930.

22 V. S. Tomar, N. Gupta, and U. Singh, "Expressions recognition based on human face," in Proceedings *of the 3rd International Conference on Computing Methodologies and Communication, ICCMC 2019*, 2019, doi: 10.1109/ICCMC.2019.8819714.

23 R. Verma, P. Choure, and U. Singh, "Neural networks through stock market data prediction," in *Proceedings of the International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017*, 2017, vol. 2017-Janua, doi: 10.1109/ICECA.2017.8212717

24 P. Kewat, R. Sharma, U. Singh, and R. Itare, "Support vector machines through financial time series forecasting," in *Proceedings of the International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017*, 2017, vol. 2017-Janua, doi: 10.1109/ICECA.2017.8212859.

25 A. Sharma, D. Bhuriya, and U. Singh, "Survey of stock market prediction using machine learning approach," in *Proceedings of the International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017*, 2017, vol. 2017-Janua, doi: 10.1109/ICECA.2017.8212715.

26 S. Sable, A. Porwal, and U. Singh, "Stock price prediction using genetic algorithms and evolution strategies," in *Proceedings of the International Conference on Electronics, Communication and Aerospace Technology, ICECA 2017*, 2017, vol. 2017-Janua, doi: 10.1109/ICECA.2017.8212724.

27 A. Roshan, A. Vyas, and U. Singh, "Credit Card Fraud Detection Using Choice Tree Technology," in *Proceedings of the 2nd International Conference on Electronics, Communication and Aerospace Technology, ICECA 2018*, 2018, doi: 10.1109/ICECA.2018.8474734.

28 H. Soni, A. Vyas, and U. Singh, "Identify Rare Disease Patients from Electronic Health Records through Machine Learning Approach," in *Proceedings of the International Conference on Inventive Research in Computing Applications, ICIRCA 2018*, 2018, doi: 10.1109/ICIRCA.2018.8597203.

29 A. Saxena, A. Vyas, L. Parashar and U. Singh, "A Glaucoma Detection using Convolutional Neural Network," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2020, pp. 815-820, doi: 10.1109/ICESC48915.2020.9155930.

30 B. Bamne, N. Shrivastava, L. Parashar and U. Singh, "Transfer learning-based Object Detection by using Convolutional Neural Networks," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2020, pp. 328-332, doi: 10.1109/ICESC48915.2020.9156060.

31 Gupta, P., Shukla, M., Arya, N., Singh, U., Mishra, K. (2022). Let the Blind See: An AIIoT-Based Device for Real-Time Object Recognition with the Voice Conversion. In: Al-Turjman, F., Nayyar, A. (eds) Machine Learning for Critical Internet of Medical Things. Springer, Cham. https://doi.org/10.1007/978-3-030-80928-7_8

32 Nguyen, B.P.; Tay, W.L.; Chui, C.K. Robust biometric recognition from palm depth images for gloved hands. IEEE Trans. Hum. Mach. Syst. 2015, 45, 799–804.