Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

MultiSURF: Optimal Feature Selection Technique for Spam Mail Detection and Classification

B. Aruna Kumari[1], C. Nagaraju[2]

[1]Research Scholar, YSR Engineering College of Yogi Vemana University, Computer Science & Engineering, Proddatur, arunakumarib1421@gmail.com
[2]Professor, YSR Engineering College of Yogi Vemana University, Computer Science & Engineering, Proddatur, nagaraju.c@yvu.edu.in

Unsolicited emails sent in bulk for malevolent purposes. These emails can clutter user inboxes, posing risks like attacks, theft, and malware infections. Detecting and blocking spam mails help to protect users from these security risks. Spam emails consume valuable network bandwidth and computational resources. Distinguishing spam from legitimate mails is difficult because most of the spam mail features are alike legitimate mail features. The impact of irrelevant features in spam mail is significantly increases time consumption rate and reduces the accuracy rate for classification. For this purpose, a new technique MultiSURF is proposed in order to eliminate irrelevant features by integrating with the random forest. This technique automatically removes irrelevant features from the dataset and improves the effectiveness of spam mail detection and classification. The evaluation results represent the proposed method performs better than ReliefF and its variants methods.

Keywords: ReliefF, SURF, SURFStar, MultiSURF, RandomForest classifier.

1. Introduction

The rise in spam emails presents major challenges for both email users and service providers. These emails not only flood inboxes but also create security threats like phishing attacks, fraud and malware distribution. Consequently, there is a growing demand for effective spam mail detection mechanisms to mitigate these threats [1]. Spam mail detection is a critical aspect of email communication due to several reasons, encompassing both technical and societal implications. Phishing emails aim to trick recipients into disclosing sensitive details, such as credit card numbers, passwords by pretending to be trustworthy organizations. Malware-laden spam can infect systems with viruses, ransomware, or spyware, compromising user data and system integrity and resource drain used by spam mails like network bandwidth, storage capacity and processing power. Large volumes of spam can overwhelm email servers, leading

to performance degradation and potential downtime. Spam emails inundate users' inboxes, cluttering their email experience and making it challenging to identify legitimate mailss. Excessive spam can lead to user frustration, reduced productivity, and decreased trust in email communication.

2. Literature Survey

In [2], a study was conducted to detect spam emails using various machine learning algorithms including ID3, c-SVC (c-Support Vector Classification), RndTree, Naïve Bayes and C4.5. To enhance the spam detection success rate, different feature selection methods (Backward Elimination, ReliefF, Forward Selection, and Fisher Filtering) and data transformation techniques were applied. The Results indicated that the c-SVC algorithm achieved the highest correct recognition rate, while the RndTree had the best ROC analysis result. The Naïve Bayes algorithm was the most efficient in terms of processing time. Both feature selection and data transformation positively impacted classification accuracy. The sub-feature set derived from the forward selection method yielded similar success rates to the original feature set, but with a reduced processing time. Data transformation improved classification accuracy by 2.46%. The overall classification success rate using single classifiers and data transformation was 93.13%. In [3], a new method for identifying e-mail spam has been proposed utilizing a hybrid bagging approach based on machine learning. This approach combines two machine learning methods: random forest and decision tree, to classify emails as either ham or spam. During preprocessing, the database is divided into sets, and various techniques are applied. CFS (Correlation Feature Selection) is employed to select relevant features. The method's effectiveness is evaluated based on accuracy, precision, recall and other metrics, achieving 98% accuracy. In the future, the researcher anticipates that more advanced techniques such as evolutionary algorithms and dataset procedures will be widely adopted to improve effectiveness. In [4], A machine learning method is employed to identify spam emails by categorizing them into two groups: spam and ham, using Sequential Minimal Optimization (SMO). The process begins by extracting features from the text of every email. A hybrid feature selection method is then applied to select the most significant features for the detection process. These chosen features are subsequently fed into the SMO algorithm to determine the final classification. This approach offers an efficient method for spam control, resulting in a simplified model with reduced computational cost. In [5], The study focused on using data mining methods to distinguish spam emails by utilizing the UCI spam base dataset. It assessed the effectiveness of various machine learning tools, feature selection methods, and ensemble learning methods. The study also compared the classification accuracy of different classifiers (such as Naïve Bayes, ensemble boosting, decision tree and ensemble hybrid boosting classifiers) using cross-validation, and the confusion matrix to demonstrate performance and accuracy results. The study found that the Ensemble learning methods, especially Bagging with Random subspace classifier, after applying feature selection methods, gave better classification accuracy results. Moreover, the hybrid technique also improved the classifiers' results, and it gave good values of precision and F-measure. The study suggests that further work is required to obtain highly accurate and interpretable classification accuracy. In [6], a new method is being proposed based on an innovative Relevance Feature Discovery (RFD) model. This method will scan email contents, categorizing patterns as general, positive, or

negative. It will then analyze whether the emails are spam or not based on these patterns and process them accordingly. Additionally, this approach will synchronize with the email server user's emails. Attached images will be detected and classified as either spam or ham. Unlike the current method, which does not include general patterns, RFD introduces general patterns to help users decide if an email is spam, thereby preventing the loss of important emails. Spam images will be detected using Histogram, File Properties, and Hough Transform techniques. The proposed system is for English language emails, but there is potential to design the system for multiple languages in the future. In [7], the process of filtering spam emails is identified by various factors like number of features, type of classifier and sample representation. To evaluate the effectiveness of this process, twelve feature selection techniques were analyzed and implemented: Term Frequency Document Frequency, Point-wise Mutual Information, Mutual Information, Normalized Mutual Information, CDM, Weighted Mutual Information, Chi-square, NGL, GSS (Galavotti Sebastiani Simi), CPD, Fisher Score, and LTC. Features from the header, body, and subject of emails were used, with both boolean and frequency-based feature vector table (FVT) representations. The findings showed that using the frequency FVT representation with the RF classifier, the header features produced the best results. When using the boolean FVT representation, features extracted from the email body performed better with the SVM classifier. Conversely, using the email subject was ineffective for identifying spam. The best feature selection approach involved using Weighted Mutual Information (WMI) and LTC to select prominent features with shorter lengths from a high-dimensional feature space. This approach resulted in an overall F1-measure of 0.978. Based on these limitations, a MultiSURF technique is implemented.

3. Motivation

Advanced feature selection methods aim to enhance machine learning model performance and efficiency by identifying the most relevant features from the dataset. Feature selection offers several benefits such as improving the model performance by focusing on the most relevant features, reducing overfitting, dimensionality reduction, enhancing interpretability by simplifying the model. To reduce computational burden, enhancing model stability and filtering out noisy features from the robust models. By incorporating domain-specific knowledge for better feature relevance, here we are proposing a MultiSURF feature selection technique integrating Random Forest classifier.

4. Existing Feature Selection Methods

4.1 ReliefF Method

ReliefF (Relief Feature Selection) is an efficient feature selection algorithm. It assesses the significance of features by their capacity to differentiate between similar and dissimilar instances [8]. ReliefF calculates a weight for each feature by comparing its values among the nearest instances of the same and different classes. Features with higher weights are deemed most significant and are chosen for incorporation into the ultimate feature subset. In the context of spam mail detection, ReliefF can be utilized to identify the most discriminative features, such as words or attributes, that differentiate between spam and non-spam emails. The

mathematical equation of ReliefF method is as follows:

$$w[A] = w[A] - \frac{1}{m-k}\sum_{j=1}^{k}(\text{diff}(A, x_i, \text{Hit}_j)) + \frac{1}{m-k}\sum_{j=1}^{k}\left(\frac{P(C) - \text{diff}(A, x_i, \text{Miss}_j, c)}{1-P(C)}\right) \quad \text{Eq (1)}$$

In Eq(1), diff (A, x, Hit_j) represents distance function measuring the difference between the values of feature A with hit or miss operations, Hit_j is the j-th nearest hit, $\text{Miss}_{j,C}$ is the j-th nearest miss from class C, and $P(C)$ is the prior probability of class C.

4.2 SURF Method

In the context of feature selection, SURF(Speeded Up Robust Features) can be used to identify informative features or keypoints in high-dimensional datasets. The mathematical equation to update the weight W[A] for feature A is as follows:

$$w[A] = w[A] - \frac{1}{m-n_i}\sum_{x_j \in N_i}(\text{diff}(A, x_i, x_j) * (2I(y_i = y_j) - 1)) \quad \text{Eq (2)}$$

In Eq(2), N_i is the set of neighbors within distance d, I is an indicator function and its value is 1 if the argument is true and 0 otherwise. n_i is the number of neighbors.

4.3 SURFStar Method

The SURFstar algorithm is a commonly used feature selection method in machine learning. It is particularly useful in tasks such as spam mail detection, where the identification of relevant features that contribute to the classification process. SURFStar method is similar to SURF, but instead of a fixed threshold d, SURFStar dynamically adjusts the neighborhood size. The weight update formula also remains similar to SURF but with dynamically defined neighborhoods.

5. Proposed MultiSURF Technique

ReliefF detects feature interactions but is computationally intensive and sensitive to parameters. SURF improves efficiency but relies heavily on a radius parameter and does not detect feature interactions. SURFStar enhances efficiency and adapts dynamically but lacks interaction detection. To overcome these drawbacks, MultiSURF technique is proposed. In this paper, MultiSURF technique integrating with Random Forest technique is proposed to select most relevant features to classify spam or ham mails in high dimensional datasets with low computational costs. In Figure 1, the proposed architecture is represented.

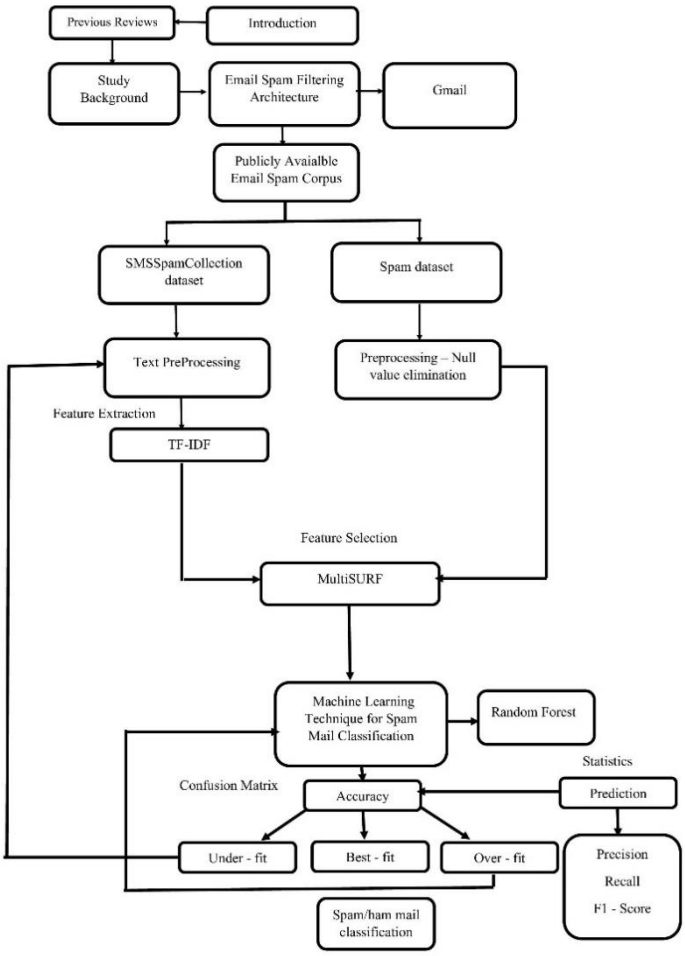


Figure1 Proposed Architecture

5.1 Feature Extraction

In this paper, for feature extraction TF-IDF is used. TF-IDF gives higher values to less frequent words and smaller values to high-frequency words. If both TF and IDF values are high, the word is rare in all documents but frequent in a single document. Calculating Term Frequency (TF) using a formula,

$$TF = \frac{(\text{Frequency of the word in the sentence})}{(\text{Total no,of words in the sentence})} \quad \text{Eq [3]}$$

Calculating IDF values from the formula,

$$IDF = \frac{\text{Total number of sentences}}{\text{Number of sentences containing that word}} \quad \text{Eq [4]}$$

5.2 Feature Selection

MultiSURF works by evaluating the relevance of each feature through a comparison of its value distribution across different classes in the dataset. MultiSURF improves feature selection with greater adaptability, robustness to noise, computational efficiency, and balanced sensitivity and specificity. The mathematical equation to update the weight W[A] is as follows:

$$w[A] = w[A] - \frac{1}{m-n_i}\sum_{x_j \in N_i}(\text{diff}(A, x_i, x_j) * (2I(y_i = y_j) - 1)) \qquad \text{Eq [5]}$$

In Eq [5], N_i indicates the nearest neighbors based on a stricter threshold.

5.3 Classification

Random Forest is integrated with MultiSURF technique to improve accurate classification over the individual Random Forest. Integrated Random Forest considers the predictions from each tree and aggregates them through majority voting to determine the final outcome. With an increased number of trees in the forest, accuracy rises while the risk of overfitting decreases. The IRF (Integrated Random Forest) algorithm is particularly useful for predicting outputs with high accuracy on large datasets and requires less training time.

6. Experimental Results

To test the effectiveness of proposed method, here we considered two datasets i.e., SMSSpamCollection and SPAM dataset. In this paper, two datasets SMSSpamCollection and SPAM dataset which are shown in figure2 and figure 3 are used for the experimentation. The SMSSpamCollection dataset comprises of 5574 records, with the first attribute being the "type of mail" which distinguishes between spam and ham emails. The second attribute is the email text which includes a variety of content. Our aim is to use this dataset to train machine learning models.

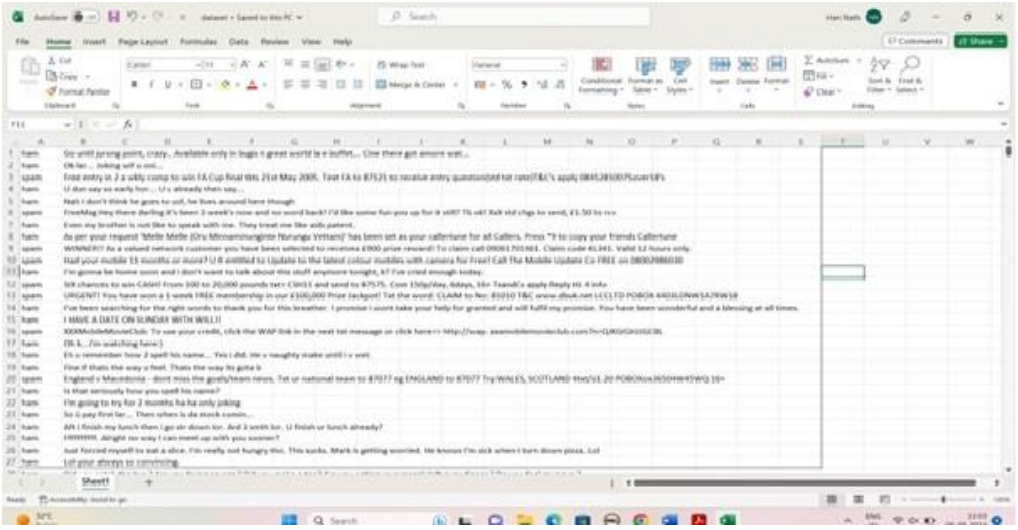
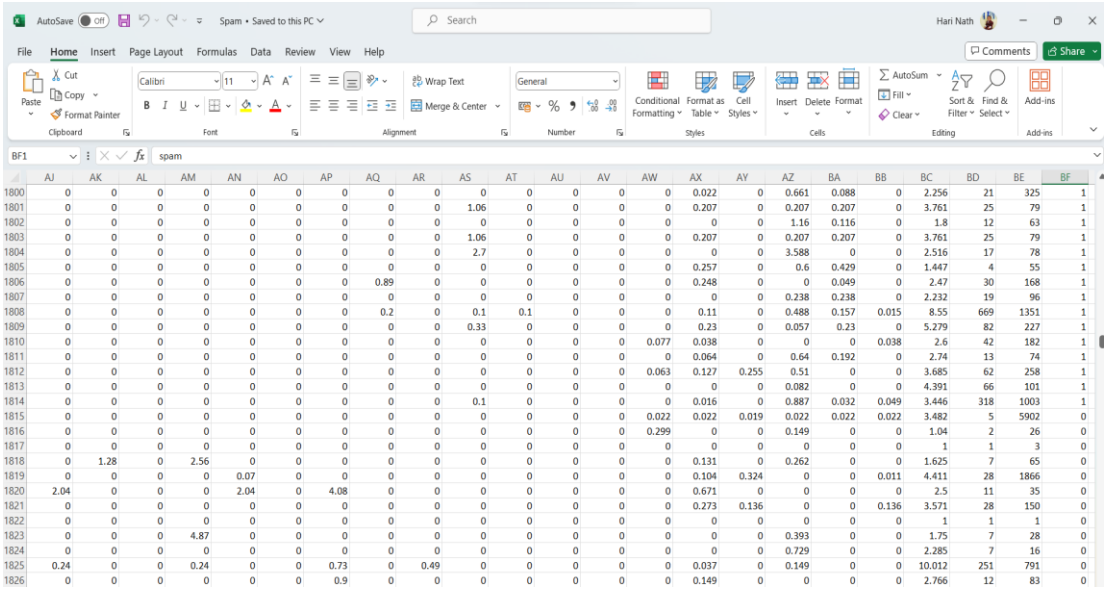


Figure 2. SMSSpamCollection Dataset

The second SPAM dataset is it consists of 1813 spam and 2788 non-spam emails. Among 4601 instances. It includes 57 attributes, which are shown in Figure 2 comprising words or characters that frequently occurred. The first 48 attributes are continuous real numbers, ranging from 0 to 100, and are labeled as "word_freq_WORD." This indicates the percentage of words present in the email, with "WORD" representing any alphanumeric character string. The next six attributes are also continuous real numbers, ranging from 0 to 100, and are labeled as "char_freq_CHAR." This indicates the percentage of characters present in the email, with "CHAR" representing any character in the email. The remaining attributes are a combination of continuous real numbers and continuous integers. The final column represents the class type, indicating whether the email is spam (denoted as "1") or not (denoted as "0").



	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU	AV	AW	AX	AY	AZ	BA	BB	BC	BD	BE	BF
1800	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.022	0	0.661	0.088	0	2.256	21	325	1
1801	0	0	0	0	0	0	0	0	0	1.06	0	0	0	0	0.207	0	0.207	0.207	0	3.761	25	79	1
1802	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1.16	0.116	0	1.8	12	63	1
1803	0	0	0	0	0	0	0	0	0	1.06	0	0	0	0	0.207	0	0.207	0.207	0	3.761	25	79	1
1804	0	0	0	0	0	0	0	0	0	2.7	0	0	0	0	0	0	3.588	0	0	2.516	17	78	1
1805	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.257	0	0.6	0.429	0	1.447	4	55	1
1806	0	0	0	0	0	0	0	0.89	0	0	0	0	0	0	0.248	0	0	0.049	0	2.47	30	168	1
1807	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.238	0.238	0	2.232	19	96	1
1808	0	0	0	0	0	0	0	0.2	0	0.1	0.1	0	0	0	0.11	0	0.488	0.157	0.015	8.55	669	1351	1
1809	0	0	0	0	0	0	0	0	0	0.33	0	0	0	0	0.23	0	0.057	0.23	0	5.279	82	227	1
1810	0	0	0	0	0	0	0	0	0	0	0	0	0	0.077	0.038	0	0	0	0.038	2.6	42	182	1
1811	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.064	0	0.64	0.192	0	2.74	13	74	1
1812	0	0	0	0	0	0	0	0	0	0	0	0	0	0.063	0.127	0.255	0.51	0	0	3.685	62	258	1
1813	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.082	0	0	4.391	66	101	1
1814	0	0	0	0	0	0	0	0	0	0.1	0	0	0	0	0.016	0	0.887	0.032	0.049	3.446	318	1003	1
1815	0	0	0	0	0	0	0	0	0	0	0	0	0	0.022	0.022	0.019	0.022	0.022	0.022	3.482	5	5902	0
1816	0	0	0	0	0	0	0	0	0	0	0	0	0	0.299	0	0	0.149	0	0	1.04	2	26	0
1817	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	3	0
1818	0	1.28	0	2.56	0	0	0	0	0	0	0	0	0	0	0.131	0	0.262	0	0	1.625	7	65	0
1819	0	0	0	0	0.07	0	0	0	0	0	0	0	0	0	0.104	0.324	0	0	0.011	4.411	28	1866	0
1820	2.04	0	0	0	2.04	0	4.08	0	0	0	0	0	0	0	0.671	0	0	0	0	2.5	11	35	0
1821	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.273	0.136	0	0	0.136	3.571	28	150	0
1822	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
1823	0	0	0	4.87	0	0	0	0	0	0	0	0	0	0	0	0	0.393	0	0	1.75	7	28	0
1824	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.729	0	0	2.285	7	16	0
1825	0.24	0	0	0.24	0	0	0.73	0	0.49	0	0	0	0	0	0.037	0	0.149	0	0	10.012	251	791	0
1826	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0.149	0	0	0	0	2.766	12	83	0

Figure 3. SPAM Dataset

The formulae to calculate accuracy, precison, recall and F1-score are as follows:

- Accuracy: How many predictions of the total number of values were accurate.

$$\text{Accuracy} = \frac{\text{truepositive}+\text{truenegative}}{(\text{truepositive}+\text{falsepositive}+\text{truenegative}+\text{falsenegative})} \quad \text{Eq [6]}$$

- Precision: Precision describes the number of accurately anticipated situations that really turn out to be positive.

$$\text{Precison} = \frac{\text{truepositive}}{(\text{truepositive}+\text{falsepositive})} \quad \text{Eq [7]}$$

- Recall: Used to retrieve true negative values

$$\text{Recall} = \frac{\text{truenegative}}{(\text{truepositive}+\text{falsenegative})} \quad \text{Eq [8]}$$

- F1 Score: It is used to classify dataset values as positive or negative

$$\text{F1-score} = \frac{2 * \text{Precision} * \text{Recall}}{(\text{Precision}+\text{Recall})} \quad \text{Eq [9]}$$

Table1. Results of SMSSpamCollection dataset

	ReliefF	SURF	SURFStar	MultiSURF
Accuracy	81.66	81.66	81.66	83.33
Precision	0.99	0.99	0.99	0.99
Recall	26.66	26.66	26.66	33.33
F1	42.1	42.1	42.1	50

Graph 1. Results of SMSSpamCollection dataset

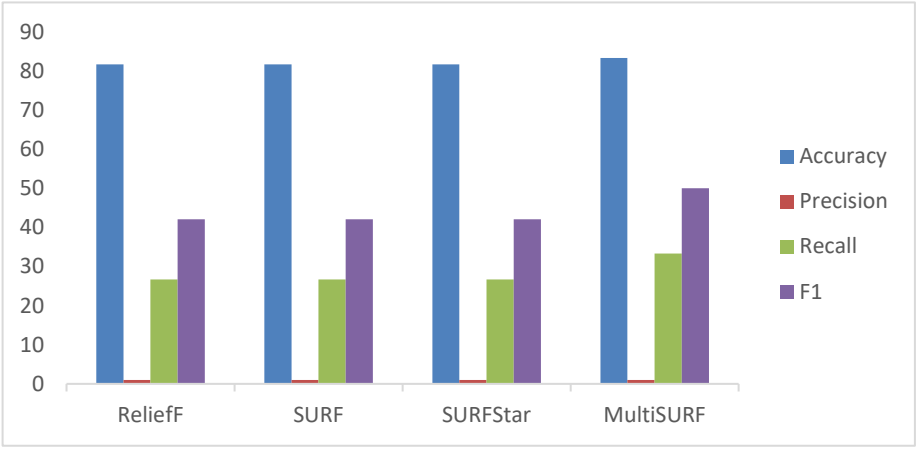
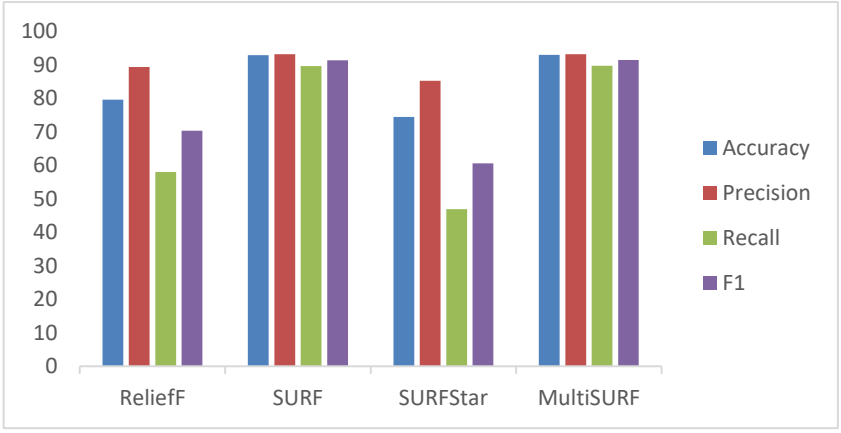


Table2. Results of SPAM dataset

	ReliefF	SURF	SURFStar	MultiSURF
Accuracy	79.58	92.90	74.43	92.97
Precision	89.33	93.15	85.22	93.16
Recall	58.05	89.60	46.96	89.77
F1	70.37	91.34	60.55	91.43

Graph2. Results of SPAM dataset



The table 1, graph 1 and table 2, graph 2 represents the accuracy, precison, recall and f1-score for the datasets SMSSpamCollection and SPAM dataset respectively. The precision represents accurate classification of samples among the positive samples and the recall specifies accurate classification rate of negative samples. Based on the graphs and tables, precision and recall got highest values for proposed method such that it impacts on increasing of accuracy rate. F1 – Score represents the combination of Precision and recall. All the parameters represent MultiSURF produced higher values.

7. Conclusions

This paper suggests an innovative approach for improving spam detection by utilizing sophisticated feature selection technique MultiSURF to identify and prioritize relevant features from email datasets, reducing dimensionality and enhancing classification accuracy. This algorithm builds on the principles of the original Relief algorithm, enhancing its ability to handle various challenges like noise, multiclass data, and high dimensionality. The subsequent phase involves utilizing Random Forest (RF) for classification, leveraging the chosen features. The MultiSURF method produced better accuracy over the ReliefF, SURF and SURFstar techniques. The MultiSURF technique produced 92.9% of accuracy rate. However, the accuracy rate can be improved by breaking down the weight formula and consider the possible modifications like reguralizing the weights, normalization of differences and adding the learning rate coefficient.

Funding Information

No funding is provided for this paper from any organization.

Conflicts of Interest

References

1. Muhammad Adnan, et.al., (2023), Improving Spam Email Classification Accuracy Using Ensemble Techniques: A Stacking Approach, International Journal of Information Security, Springer.
2. Hidayet Takei, Fatema Nusrat (2023), High Accurate Spam Detection with the Help of feature selection and Data transformation, The International Arab Journal of Information Technology, Vol. 20, No. 1, pp: 29-37.
3. Alanazi Rayan (2022), Analysis of e-Mail Spam Detection Using a Novel Machine Learning – Based Hybrid Bagging Technique, Hindawi, Computational Intelligence and Neuroscinece, pp: 1-12.
4. Ahmed Al – Ajeli, et.al., (2020), Improving Spam Email Detection Using Hybrid Feature Selection and Sequential Minimal Optimisation, Indonesian Journal on Electrical Engineering and Computer Science, Vol. 19, No. 1, pp: 535 - 542.
5. Doaa Mohammad Ablel-Rheem, et.al., (2020), Hybrid Feature Selection and Ensemble Learning Method for Spam Email Classification, International Journal of Advanced Trends in Computer Science and Engineering, Vol. 9, No. 1.4, pp: 217 – 223.
6. Sayarabanu B. Nadaf, Anil D. Gujar (2016), Spam Mail Detection Using Relevance Feature Discovery, International Journal of Science and Research, Vol. 5, Issue 7, pp: 768 – 770.

7. Josin Thomas, Vinod P, Nisha S Raj, (2014), Towards Spam Mail Detection Using Robust Feature Evaluated with Feature Selection Techniques, International Journal of Engineering and Technology, Vol. 6, No. 5, pp: 2144 – 2158.
8. Farhad Soleimanian Garehchopogh, Seyyed Keyvan Mousavi (2019), A New Feature Selection in Email Spam Detection by Particle Swarm Optimization and Fruit Fly Optimization Algorithms, Journal of Computer and Knowledge Engineering, Vol. 2, No.2, pp: 49 – 62.
9. Bhagyashri U. Galkwad, P. P. Halkarnikar (2013), Spam E-mail Detection by Random Forests Algorithm, International Journal of Advanced Computer Engineering and Communication Technology, Vol. 2, Issue 4, pp: 1 – 8.
10. B. Aruna Kumari, C. Nagaraju (2024), Efficient Genetic Algorithm for Spam Mail Detection and Classification, Global Journal of Engineering and Technology Advances, 18(02), pp:35 -41.
11. B. Aruna Kumari, C. Nagaraju (2024), A Generalized Two – Level Ensemble Method for Spam Mail Detection, Journal of Electrical Systems, Vol. 20, No. 2, pp: 1570 – 1579.