A feature-centric spam email detection model using diverse supervised machine learning algorithms

Ammara Zamir, Hikmat Ullah Khan, Waqar Mehmood, Tassawar Iqbal and Abubakker Usman Akram
Department of Computer Science, COMSATS University Islamabad, Wah Campus, Wah Cantt, Pakistan

Received 31 July 2019
Revised 11 November 2019
19 April 2020
25 May 2020
Accepted 7 June 2020

Abstract

Purpose – This research study proposes a feature-centric spam email detection model (FSEDM) based on content, sentiment, semantic, user and spam-lexicon features set. The purpose of this study is to exploit the role of sentiment features along with other proposed features to evaluate the classification accuracy of machine learning algorithms for spam email detection.

Design/methodology/approach – Existing studies primarily exploits content-based feature engineering approach; however, a limited number of features is considered. In this regard, this research study proposed a feature-centric framework (FSEDM) based on existing and novel features of email data set, which are extracted after pre-processing. Afterwards, diverse supervised learning techniques are applied on the proposed features in conjunction with feature selection techniques such as information gain, gain ratio and Relief-F to rank most prominent features and classify the emails into spam or ham (not spam).

Findings – Analysis and experimental results indicated that the proposed model with sentiment analysis is competitive approach for spam email detection. Using the proposed model, deep neural network applied with sentiment features outperformed other classifiers in terms of classification accuracy up to 97.2%.

Originality/value – This research is novel in this regard that no previous research focuses on sentiment analysis in conjunction with other email features for detection of spam emails.

Keywords Email, Feature selection, Machine learning, Deep neural networks, Feature sets, Email classification, Email spam

Paper type Research paper

1. Introduction

Spam is undesired electronic information spread by spammers with the intention to cause psychological and monetary harm to the victims. Spam may have numerous forms including Web spam, review spam, short message service (SMS) spam and email spam. Web spam deceives search engines into making the wrong decisions in the ranking of Web pages. In review spam, spammers often exploit reviews by giving false positive (FP) reviews (Akram *et al.*, 2018). SMS spam (Popovac *et al.*, 2018) is delivered to customers over text messaging; it not only annoys customers but may also cause financial loss to service providers. Email spam contains an advertisement or irrelevant text, sent by spammers having no relationship with the recipient (Cormack and Lynam,

0264-0473
DOI 10.1108/EL-07-2019-0181

2005). Email spam is sent in many ways, such as by using an insecure server, using automated generated accounts, newsgroup postings and using malware to get user addresses. Email spam causes several threats, such as cyber-attacks, fake e-marketing and loss of legal emails (Cormack, 2008). For instance, Aramco, the largest oil company, became a victim of email spam, resulting in the destruction of 35,000 hard disks by opening spam email containing a harmful link (Hijawi *et al.*, 2017).

For email spam detection, a number of machine learning-based approaches, such as content-based supervised learning, rule-based learning, semi-supervised learning and unsupervised learning, have been proposed. To detect email spam, content-based approaches focus on the email content features (Mirza *et al.*, 2017). The rule-based approaches are simple and efficient because of the model-based approach (Thomas *et al.*, 2011) which classifies the emails based on a set of rules, also known as a *decision list*. The black list and white list are the two main approaches to rule-based learning. Semi-supervised learning deals with labelled as well as unlabelled data. Use of a small amount of labelled data with a large amount of unlabelled data results in a better learning accuracy of classifiers (Li *et al.*, 2014); whereas, the unsupervised learning methods classify emails from an unlabelled data set (Alsmadi and Alhami, 2015).

In recent research studies, email spam classification exploits a content-based features set; however, a limited number of features are considered, which alone rarely fulfils the requirements of classification. To deal with the abovementioned issue, this paper presents a new feature-centric spam email detection model (FSEDM) based on content, user, spam lexicon, semantic and sentiment features to classify spam emails. The main research contributions are as follows:

- A feature sets-based classification model is proposed which exploits content, semantic, sentiment, spam lexicon and user-based features to classify spam emails.
- The model exploits the role of sentiment features to classify spam emails.
- Popular feature selection algorithms are applied to find the important features among proposed features of a selected data set.
- It exploits the role of proposed features in performance of diverse machine learning algorithms.
- Diverse feature sets of selected data sets are used to train and test classifiers to assess their classification capability.

The rest of the paper is organized as follows: Section 2 reviews the earlier research studies related to feature extraction and machine learning techniques to classify spam email. Section 3 presents the research methodology. In Section 4, the experimental setup is discussed, while Section 5 describes the feature analysis and experimental results. Section 6 presents the conclusion of the proposed research study.

2. Related work

This section presents the previous studies related to supervised machine learning techniques to extract features from an email.

2.1 Features-based research studies

Features represent properties used to assess the activity within a user's email message. This section describes the features used for email spam classification shown in Table 1. Email features are extracted to reduce the classification time and increase accuracy. The relevant literature describes the following features which are widely used for spam email classification:

Email header and body features:

- stylometric features;
- behavioural features;
- network features; and
- SpamAssassin features.

Email header and body features along with other features are used in numerous research studies for spam email detection (Alqatawna *et al.*, 2015; Balakumar and Vaidehi, 2008; Islam and Xiang, 2010; Li *et al.*, 2014; Pérez-Díaz *et al.*, 2012; Sohn and Chung, 2015). A study by Li *et al.* (2014) considered both email header and email body features. The email header features contain: from, to, cc and characters count in the subject of the email, and word count in the mail-subject field (Alqatawna *et al.*, 2015). As the email body consists of the main contents of the email, the email body features contain word count, number of function words (e.g. information and click), subject-length, message size, attachment count, attachment size and embedded message count. Rayan *et al.* (2017) used stylometric features which define the writing style of an author. The writing style of an author includes the number of function words, unique word count, new lines, character count, attachment count and so on (Sohn and Chung, 2015).

In the relevant literature, the use of email behavioural features is also very common. In this regard, Bhat *et al.* (2011) used behavioural features. These behavioural features include HTML, images, scripts, hyperlinks, attachments of MIME type file, text/binary documents, file attachment, sent emails count, unique email count and unique sender addresses count. The authors used a vocabulary list to check the terms of an incoming email. Each term is defined by synonyms, specialization and generalization (Bhat *et al.*, 2011). Rayan *et al.* (2017) exploited network-based features which include packet size, transmission control protocol/internet protocol headers, name length of the sender and frequency of receiving email (Sohn and Chung, 2015). Islam and Xiang (2010) considered SpamAssassin features of email along with network-based features. The SpamAssassin includes features such as header text, body phrases text, name, system block list, whitelist/blacklist and character set (Islam and Xiang, 2010).

Email spam features	References
Header + body	Li *et al.* (2014)
Header + body + network	Alqatawna *et al.* (2015), Li *et al.* (2014); Sohn and Chung (2015)
Header + body + SpamAssassin + network	Islam and Xiang (2010)
Header + body + SpamAssassin	Méndez *et al.* (2012); Moh and Lee (2011)
Header + body + SpamAssassin + behavioural	Islam and Xiang (2010)
Header + body + behavioural	Alqatawna *et al.* (2015)
Header + body + term-based	Pérez-Díaz *et al.* (2012)
Header + body + behavioural + term-based	Balakumar and Vaidehi (2008), Carmona-Cejudo *et al.* (2011)
Header + body + stylometric	Sohn and Chung (2015)

Table 1. Features used for email spam classification

George and Vinod (2018) proposed an approach by applying natural language processing to extract composite email features. These composite features (including character-based, word-based, tag-based and structure-based features) are extracted from the Enron data set. Features are ranked using dimensionality reduction algorithms. Experiments were carried out on individual and combined features too (George and Vinod, 2018).

2.2 Supervised machine learning approaches

Machine learning techniques suitable for diverse classification problems are shown in Table 2, including subjectivity analysis (Khan and Daud, 2017) and sentiment classification for forum posts (Khan, 2017). Moreover, supervised machine learning techniques are applicable on the labelled data.

Faris *et al.* (2019) introduced a system based on genetic algorithm and random weighted network to deal with email spam. The researchers engrafted the automatic identification feature in the proposed system and introduced the automatic identification features capable of extracting the relevant features of an email during the classification process. The proposed system is tested on three data sets. Results showed that the proposed system can achieve remarkable results in terms of performance evaluation measures, and the proposed feature can extract the most relevant features of an email during processing (Faris *et al.*, 2019).

Méndez *et al.* (2019) proposed a new feature selection model which converts groups of words into topics using a semantic ontology approach. This study applied nine machine learning approaches in conjunction with information gain (IG), latent Dirichlet allocation and a statistical model which describes why some features of a data set are similar. The proposed model also used semantic-based feature selection techniques. Experimental results showed that the approach is effective for developing spam filters using the topic-driven approach (Méndez *et al.*, 2019).

Singh (2019) proposed a swarm-based water drops algorithm to filter email spam. The proposed algorithm is used in conjunction with a naïve Bayes (NB) classifier. The water drops algorithm is used to make a subset of features of an email data set. Then, NB is applied to categorize email into spam and ham. NB outperformed other classifiers (Singh, 2019).

Mohammed *et al.* (2018) proposed a new agent-based anti-spam model. The proposed model takes visual information and texts of an email in a filtering process. The proposed model is implemented using Java Environment. The proposed model was applied on the email data set and gave better results than other classifiers in terms of accuracy (Mohammed *et al.*, 2018).

The role of existing tools generating functional regular expressions using any input from an email data set is problematic. These tools are difficult to configure and low in

Machine learning techniques	References
NB	Esmaeili *et al.* (2017), Feng *et al.* (2016)
SVM	Renuka and Visalakshi (2014), Song (2013)
kNN	Firte *et al.* (2010)
Decision tree	Zhang *et al.* (2014), Zhuang *et al.* (2017)
RF	Gaikwad and Halkarnikar (2014)
Neural network	Barushka and Hájek (2016)
Artificial neural network	Idris *et al.* (2014), Idris and Selamat (2014)
NB	Esmaeili *et al.* (2017), Feng *et al.* (2016)

Table 2.
Algorithms for email classification

performance. To address this issue, Ruano-Ordás *et al.* (2018) introduced Regex, a novel automatic spam filtering tool. The proposed tool avoids FP errors. The computational time of the proposed tool is less as compared to other tools. The proposed tool outperformed other techniques in automatic pattern recognition of email spam (Yakovlev, 2018).

Barushka and Hájek (2018) proposed a spam filter based on N-grams feature selection, distribution-based balancing algorithm and a deep multi-layer perceptron with rectified linear units. The proposed system is applied on the four benchmarks data sets (Enron, SpamAssassin, SMS Spam Collection and social networking). The proposed model is capable of capturing complex features from high-dimensional data. It outperformed other classifiers in terms of accuracy and classified major and minor classes of spam (Barushka and Hájek, 2018).

Esmaeili *et al.* (2017) carried out spam detection using text classification. The proposed method uses Bayesian and principle component analysis to classify the emails from users' mailboxes. The proposed method extracts all the tokens, and divides and selects the best token with the help of feature selection methods. Afterwards, top selected tokens are used to classify the given emails into spam or ham. The proposed method applies the NB algorithm which does not consider the interdependence between features and token in this regard, thus optimal accuracy is not achieved (Esmaeili *et al.*, 2017).

Renuka and Visalakshi (2014) introduced latent semantic indexing, in addition to the feature selection method to increase the accuracy of the support vector machine (SVM) classifier. The experiments are carried out on the Ling-Spam data set, and the results are verified using performance evaluation measures (Renuka and Visalakshi, 2014). Another approach of creating filters for spam using *k*-nearest neighbour (kNN) was introduced by Firte *et al.* (2010). The proposed filter updates the data and the list of words used in an email. The proposed filter is an offline tool which uses kNN and a pre-classified email data set for the learning process (Firte *et al.*, 2010).

Zhuang *et al.* (2017) proposed a new method of dynamic features bundling for decision trees. The objective of the method is to perform collective judgment in the splitting phase, learn more knowledge from features and embed feature transformation into the induction phase. The proposed method reduces the extra pre-processing step for the transformation of static features. According to the results, up to 2–9% of area under the curve (AUC) improvement is recorded for the imbalance data set (Zhuang *et al.*, 2017).

In another study, Varghese and Dhanya (2017) focused on finding the best features of an email data set. For this purpose, features, such as bag-of-words, bigram bag-of-words, part of speech (PoS), tag and bigram PoS tag are considered. For feature selection, IG is used. Singular value decomposition is used as a matrix factorization. Afterwards, AdaBoost, random forest (RF) and SVM are used for a model generation. Experiments are carried out on a single feature model (Varghese and Dhanya, 2017).

2.3 Analysis on reviewed techniques

In reference to the above reviewed related work, different techniques are used to detect spam email classification. These techniques include frameworks based on email features and supervised machine learning techniques. Features-based frameworks include header, body, network, term-based, behavioural and stylometric features. These feature sets are applied with machine learning techniques to detect spam emails. While sentiment features are not taken into consideration for spam email detection. Hence, sentiment features based on semantics can be considered to detect spam emails.

3. Research methodology

This section discusses the proposed framework, the feature sets and the algorithm.

3.1 Proposed framework (feature-centric spam email detection model)

The proposed framework for email spam detection is shown in Figure. 1. Firstly, R language is used to perform pre-processing on the selected data set, and then Python is used to extract different features, including content, sentiment, semantic, spam lexicon and user-based, from the cleaned data set. To compute sentiment score, VADER is used. VADER is free code provided by Natural Language Toolkit. Thirdly, extracted features are normalized to feed in to the classifiers. Afterwards, selection of top-ranked features is carried out using IG, gain ratio (GR) and Relief-F algorithms. K-fold (k = 10) cross validation and holdout (70–30% split) are applied along with supervised machine learning algorithms including SVM, RF, AdaBoost, bagging, MLP, deep neural network (DNN) and J48. Default parameter settings are used except for DNN. For DNN, 300 iterations with 400 hidden layers and adaptive learning rate are used. Results are evaluated using standard performance evaluation measures: accuracy, *f*-measure, recall and precision. Results are computed on a Core i7 8th Generation computer with 8 GB RAM and 256 SSD.

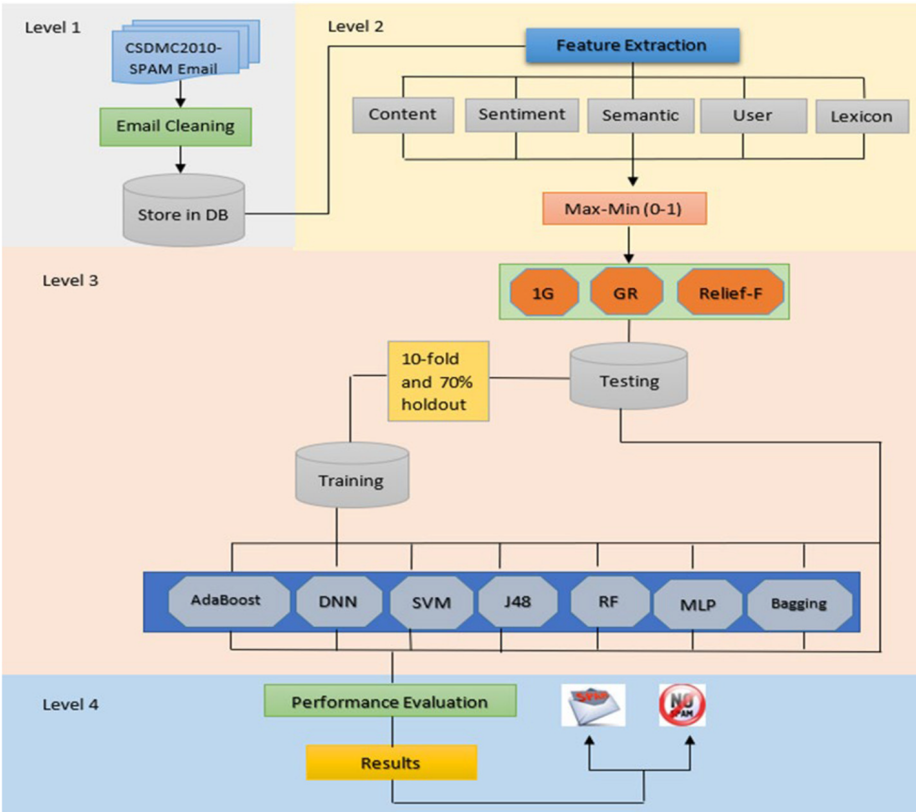

Figure 1.
A proposed framework for the proposed model (FSEDM)

3.2 Feature engineering

The feature sets are classified into five categories, namely, content-based, lexical-based, semantic-based, sentiment-based and user-based, as shown in Table 3. The content-based features are computed from both the header and the body of the email. The lexical-based features include words such as *luxury*, *auction* and *entertainment*, which are used by spammers to mislead email users. The sentiment-based features are computed using the VADER toolkit based on semantics. Sentiment features exploit positive words, negative words and emotions. The sentiment is a complex and multi-dimensional concept. An email is considered to be spam if it contains more negative and positive sentiments or emotional symbols. User-based features include features from the user's profile including name and are calculated using equation (1):

$$\begin{aligned} &If\,(\text{Profile Name}): \quad S_p = 1, \\ &\text{Else}\; S_p = 0 \end{aligned} \tag{1}$$

Semantic similarity is based on the similarity of two concepts, the title and the content of the email. Semantic similarity is computed using equation (2), where T is the title of an email and C is the content of an email (Jiang and Conrath, 1997).

$$cos\theta = \frac{\sum_{i=1}^{n} T_i C_i}{\sqrt{\sum_{i=1}^{n} T_i^2}\sqrt{\sum_{i=1}^{n} C_i^2}} \tag{2}$$

Algorithm 1: The proposed algorithm

```
Input: Data of emails
Output: email detection as spam or ham.
 1. Initialize-Variables
```

$N_{URL}, N_{RW}, N_{UW}, N_{QW}, N_A, N_{CO}, N_{CW}, N_{POS}, F_{LX}, N_{Q}, _{NW}, S_P, N_S, N_P, N_N, N_{EM}, F_S$

Features type	Features symbol	Description
Content-based	N_W	Number of words in the email
	N_{URL}	Number of URLs
	N_{RW}	Number of repetitive words
	N_{UW}	Number of unique words
	N_{QW}	Number of quoted words
	N_A	Number of attachments
	N_{CO}	Number of co-occurring words
	N_{CW}	Number of capitalized words
	N_{POS}	Number of nouns and pronouns
	N_{EM}	Contains emotion symbols
	S_Q	Number of question marks
Lexical-based	F_{LX}	Number of spam words in the lexicon
User-based	S_P	Features based on the user's profile name
Sentiment-based	s_p	Sentiment score of positive words
	S_N	Sentiment score of negative words
	N_{EM}	Emotional symbols
	N_S	Combined sentiment score
Semantic-based	S_{CS}	Similarity score between the title and the content of an email

Table 3. List of proposed features

2. For each, email $e \in E$, by a user $u \in U$.
3. $N_W = CountWords(e)$

▷Computation of content Feature Set (F_C)

4. $N_{URL} = CountURLs\ (e)$
5. $N_{RW} = CountRepetitiveowrds\ (e)$
6. $N_{UW} = CountUniquewords\ (e)$
7. $N_{QW} = CountQoutedwords\ (e)$
8. $N_A = CountAttachments(e)$
9. $N_{URL} = CountURLs\ (e)$
10. $N_{CO} = CountCooccuringwords\ (e)$
11. $N_{CW} = CountCapitalizedwords\ (e)$
12. $N_{POS} = CountNouns/pronouns\ (e)$
13. $N_Q = CountQuestionmarks\ (e)$
14. $F_C = F_{C,}, N_{URL}, N_{RW}, N_{UW}, N_{QW}, N_A, N_{CO}, N_{CW}, N_{POS}, N_Q]$

▷Computation of User Feature Set (F_U)

15. *If* (Profile Name)
16. $S_p = 1$
    Else
17. $S_p = 0$
18. $F_U = [F_C;\ S_P]$
19. $F_{LX} = COMPUTESEMANTICSSOCR(e)$

▷Computation of Semantic Feature (F_{SC})

20. $S_{SC} = computesemanticsocre\ (e)$
21. $F_{SC} = [F_{SC}; S_{SC}]$

▷Computation of Sentiment Feature (F_S)

22. $S_S SumCombinedSentimentScore(e)$
23. $N_p\ Countpositivesentimentscore(e)$
24. $S_N = Sumnegativesentimentsscore\ e$
25. $N_{EM} = Comtemotionalsymbls(e)$
26. $N_S = |N_P - |N_N|$
27. $F_S = [F_{SB}; N_S, N_p, N_N, N_{EM}]$
28. Class = Classifier $[F_{SB}; N_S, N_p, N_N, N_{FM}]$
29. If Class = 1 then
30. $p' = e_s$
31. *Else*
32. $p' = e_h$
33. *end if*
34. STOP ▷ (END of Algorithm)

4. Experimental setup

Below is a discussion of the data set to be used and the performance evaluation measures to be applied.

4.1 Data set

The CSDMC2010_SPAM data set is the latest data set of emails. The CSDMC2010_SPAM data set contains 32% of spam ratio, which is equal to the spam rate of SpamAssassin, a famously used data set for spam detection. This data set is available freely for research purposes (https://github.com/erayon/Email-spam-filter-naive-bayes-classifier-scikit-learn-text-classification/tree/master/CSDMC2010_SPAM/CSDMC2010_SPAM, accessed 10

January 2019). This data set has been used in earlier research studies (Al-Shboul *et al.*, 2016; Hijawi *et al.*, 2017; Liu and Moh, 2016; Shams and Mercer, 2013, 2016). Characteristics of the data set are shown in Table 4.

4.2 Machine learning techniques

Here are the details of the techniques applied on the selected data set.

4.2.1 AdaBoost. AdaBoost is a boosting learner that is used to build a strong classifier as a linear combination. AdaBoost uses weak learners to make good predictions:

$$f(y) = \sum_{i=1}^{i} a_i h_i(y) \tag{3}$$

The weak classifier produced output is $h_i(y)$. So, each weak learner is assigned a_i. For each iteration of i, a weak learner and a_i is selected. Computational complexity of h_i is independent of y. AdaBoost is the simplest algorithm and fairly good in generalization.

4.2.2 Random forest. RF is an ensemble learner which combines weak classifiers to make a strong classifier. RF is used to increase the prediction accuracy. It avoids overfitting to produce better results. RF is used to model the non-linear class boundaries:

$$Regression: \frac{1}{I}\sum_{i=1}^{I} f_i(Q) \tag{4}$$

Here, in equation (4), Q is the training set and I are the responses, whereas f_i is a regression tree which predicts class from a given set.

4.2.3 J48. The J48 algorithm is an ensemble learner and calls the target variable of the new data set. J48 is the implementation of IDE3 and is used for data mining. The disorder of data is called *entropy* which is measured in bits. Entropy is also known as the measure of uncertainty in any random sample. Equation (5) is used to calculate entropy:

$$Entropy(x\ |y) = 1 + \frac{|Z_x|}{|z|} log \frac{|Z_x|}{|z|} \tag{5}$$

4.2.4 Support vector machine. SVM is known as the kernel method. SVM makes an N-dimensional hyperplane that splits the data into two categories. SVM [equation (6)] uses z as a test point for classification. SVM is a good learner because of regular optimization used to avoid overfitting, kernel function for expert knowledge and convex optimization:

$$\mathrm{f(z) = sign\,((p,\ \varnothing(z))) - q} \tag{6}$$

4.2.5 Bagging. Bagging is an ensemble meta-algorithm and stands for bootstrap aggregation. Bagging increases the accuracy of machine learning algorithms and decreases the classification rate. Bagging uses votes for prediction.

Total email messages	4,327
Spam	2,949
Ham	1,378

Table 4. Characteristics of the data set

Table 5.
Feature ranking by IG, GR and Relief-F

Sr. No.	IG Ranked features	IG Value	GR Ranked features	GR Value	Relief-F Ranked features	Relief-F Value
1	Co-occurring words	0.0016	Semantic score	0.0297	User features	0.0482
2	Spam lexicon	0.0018	Repetitive words	0.0154	Comp. sentiments	0.0394
3	Pos. sentiment	0.0045	Spam lexicon	0.0161	Neut. sentiment	0.0268
4	Number of nouns	0.0058	Length of email	0.0165	Pos. sentiment	0.0203
5	Neut. sentiment	0.0068	Comp. sentiments	0.0170	Neg. sentiment	0.0169
6	Repetitive words	0.0073	Number of nouns	0.0177	Semantic score	0.0157
7	User features	0.0080	Quoted words	0.0195	Spam lexicon	0.0091
8	Comp. sentiments	0.0082	Neut. sentiment	0.0212	Repetitive words	0.0073
9	Length of email	0.0101	Pos. sentiment	0.0851	Number of nouns	0.0058
10	Quoted words	0.0106	Neg. sentiment	0.0536	Quoted words	0.0039

Figure 2.
Content-based feature analysis

4.2.6 Deep learning. Deep learning is also known as DNN, founded on a feed-forward neural network. Deep learning is trained through stochastic gradient descent using the back-propagation method. The deep learning network contains multiple hidden layers. These layers are comprised of neurons with tanh, max out activation and rectifier. The multi-threading concept is used to compute the global model across the network. Data abstraction is achieved through the deep learning method. Deep learning is used to increase the classification accuracy and text analysis (Lecun *et al.*, 2015).

4.2.7 Multilayer perceptron. Multilayer perceptron (MLP) is a class of feed-forward neural network that uses three layers of the node. MLP does not need to store the sample. MLP implements non-linear classifiers. The activation function is calculated as shown in equation (7). Tanh value ranges from 1 to −1, while y_i is the output of the i^{th} neurone:

$$x(y_i) = \tanh(y_i) \tag{7}$$

4.3 Selection method

This section discusses the applied feature selection algorithms.

4.3.1 Information gain. IG calculates the information of an attribute given about a class. It is used to measure the reduction in entropy. IG is widely used to extract useful features from data (Uysal, 2016). It is computed using the equations given below, where v defines the class number and p_v defines the probability of any item:

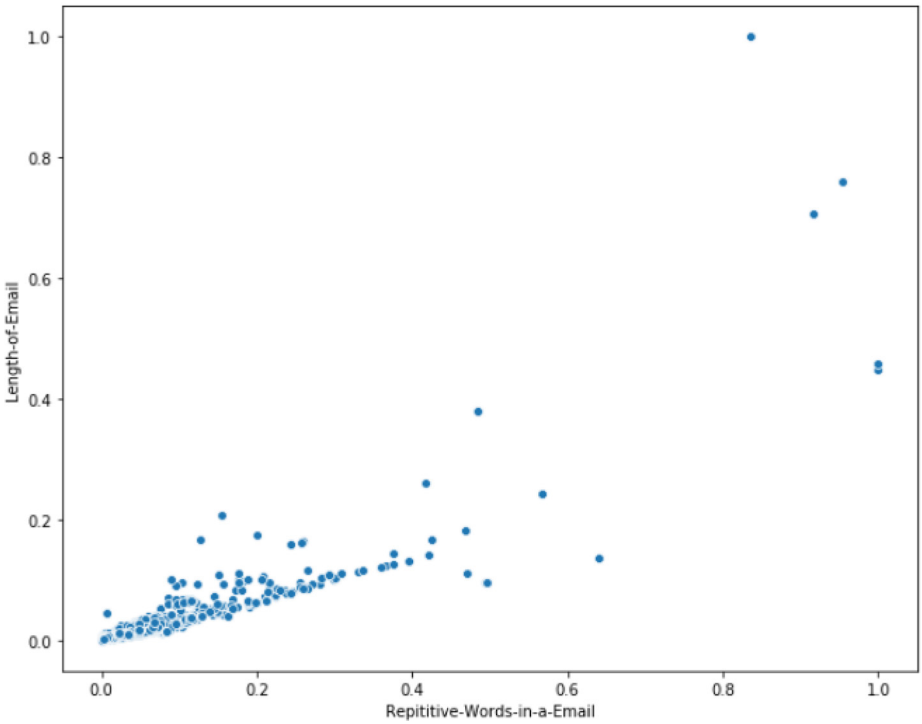

Figure 3.
Repetitive words

$$\mathrm{info}(K) = -\sum_{x=1}^{v}(P_v \log_2 P_v) \tag{8}$$

$$\mathrm{info}(K) = -\sum_{x=1}^{v}\frac{|K_v|}{|K|} * \mathrm{info}(K_v) \tag{9}$$

$$\mathrm{IG}(K) = \mathrm{info}(K) - \mathrm{infoA}(K) \tag{10}$$

4.3.2 Gain ratio. GR is the modification of IG and is used to reduce the biasness of IG. GR is used to normalize the value of IG by using intrinsic information. This intrinsic information is calculated using equation (11). GR is widely used for dimension reduction (Dai and Xu, 2013):

$$\mathrm{Info}(K) = -\sum_{x=1}^{v}\frac{|K_v|}{|K|} * \log_2\frac{|K_v|}{|K|} \tag{11}$$

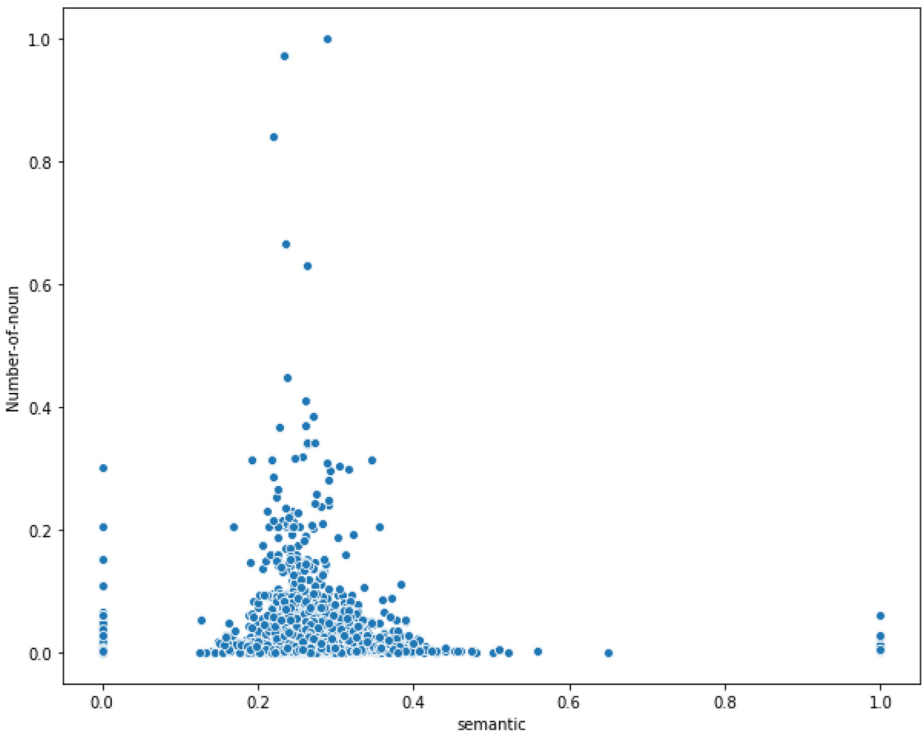

Figure 4.
Number of nouns

$$\mathrm{GR}\,(\mathrm{K}) = \frac{\mathrm{IG}}{\mathrm{Info}(\mathrm{K})} \quad (12)$$

4.3.3 Relief-F. Relief-F is used to evaluate the value of an attribute to the nearest instance of the same and different class. Relief-F works on both continuous and discrete data, and is also used for multi-label feature selection (Spolaôr *et al.*, 2013).

4.4 Performance evaluation measures

To check how accurately the classifiers classified the email spam training set, classification accuracy, precision and recall are taken as the measure. Accuracy, precision and recall are defined below, respectively.

4.4.1 Accuracy. Accuracy is used as a performance measure in the domains of information retrieval and data mining. It depicts the fraction of the results that have been successfully retrieved:

$$\mathrm{Accuracy} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \quad (13)$$

Here, FP, FN, TN and TP stand for false positive, false negative, true negative and true positive, respectively.

4.4.2 Precision. Precision is the performance evaluation measure that may be known as the ratio of retrieved documents that are related to the search:

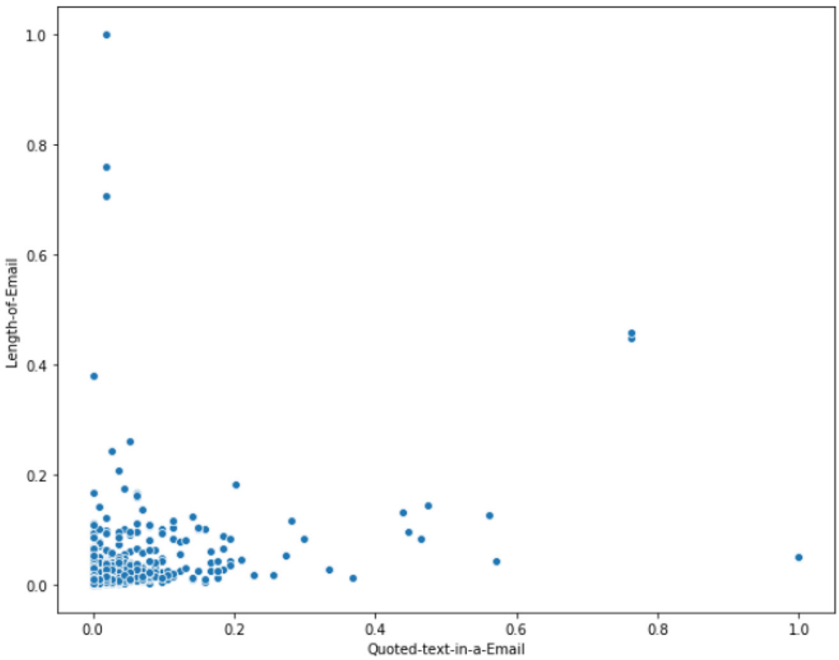

Figure 5.
Quoted words

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{14}$$

4.4.3 F-measure. The f-measure takes precision and accuracy. It may be considered as the weighted average of both values:

$$\text{F} = \frac{2\,\text{X precision X Recall}}{\text{precision} + \text{Recall}} \tag{15}$$

4.4.4 Recall. Recall, also known as *sensitivity*, is the ratio of related instances that have been retrieved over the total amount of retrieved instances:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{16}$$

5. Experimental results

In this section, empirical analysis is discussed which consists of feature selection using three state-of-the-art dimensionality reduction techniques and analysis of applied classifiers using selected feature sets. Moreover, comprehensive results analysis considering individual

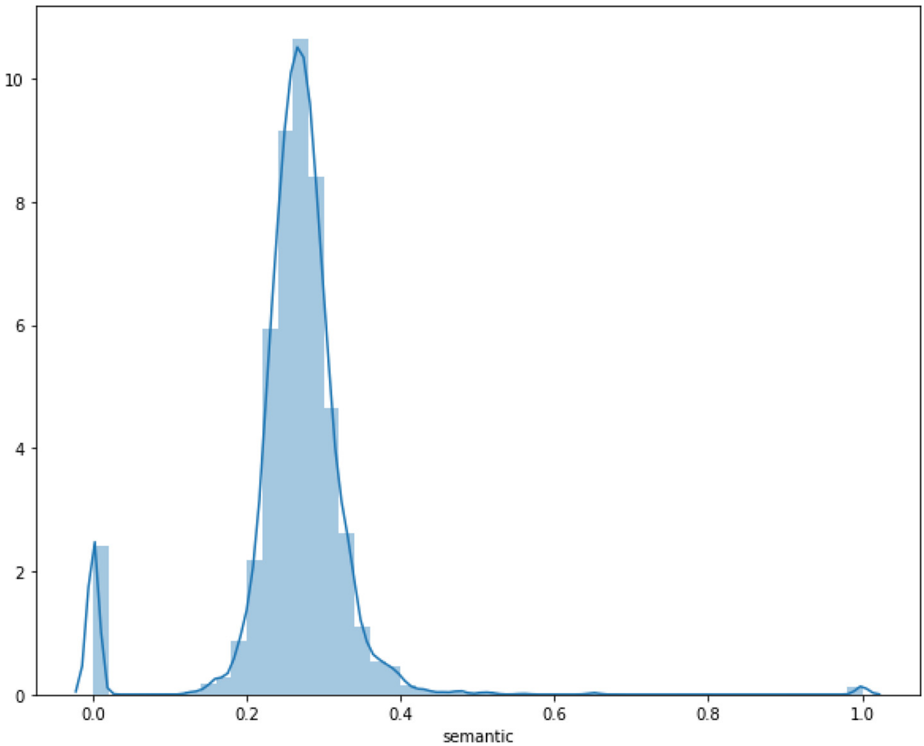

Figure 6.
Semantic feature

feature sets and their diverse combinations of features is described. The comparative analysis of the applied classifiers is also discussed.

5.1 Top-ranked features

A total of 18 features are extracted from each email in the data set which are classified into the following five categories: content, sentiment, semantic, user and spam lexicon. Later on, the feature selection techniques: IG, GR and Relief-F are applied to find the importance of different features among the set of 18 computed features. The top ten features are ranked by the applied feature selection techniques as shown in Table 5. Semantic score, positive sentiment, negative sentiment, neutral sentiment, compound sentiments, repetitive words, quoted text, number of nouns and user features are the more important among all the other features. All feature selection techniques have ranked these attributes and shown the significance of these attributes. Sentiment and semantic features show more impact among

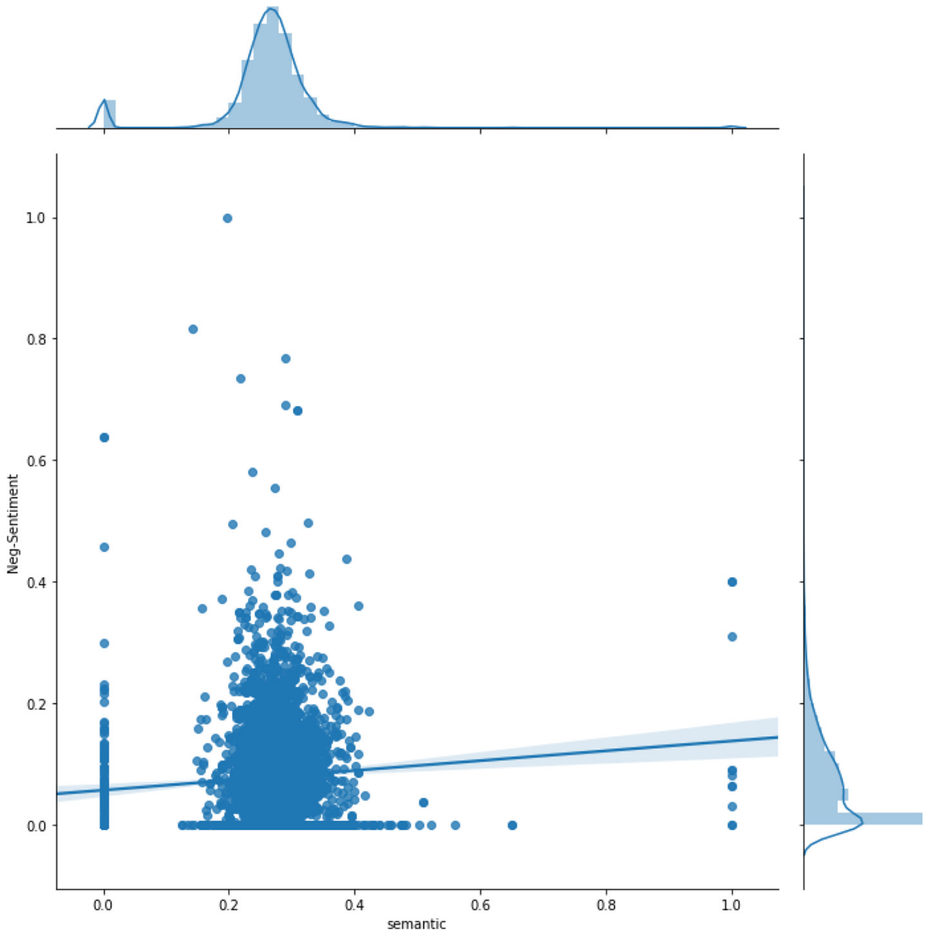

Figure 7.
Negative sentiment

the other feature categories, while the number of nouns, repetitive words and quoted text are more significant among the content-based features as shown in Table 5.

5.2 Proposed features analysis

This section provides the analysis on computed features of a select email data set. Length of an email is an important factor; spammers use either short or too long messages to deceive users. When content-based features are analysed, the repetitive words, the number of nouns and quoted text shows more significant values in conjunction with the length of email and semantic features as shown in Figure 2.

The values of these three features: repetitive words, number of nouns and quoted text in emails show increased values when the length of email is decreased or increased as shown in Figure 3-5, respectively. While the other content-based features have low or minor significance as shown in Figure 2.

Further, a graph of the semantic feature shown in Figure 6 shows a smooth distribution curve which demonstrates the importance of this feature.

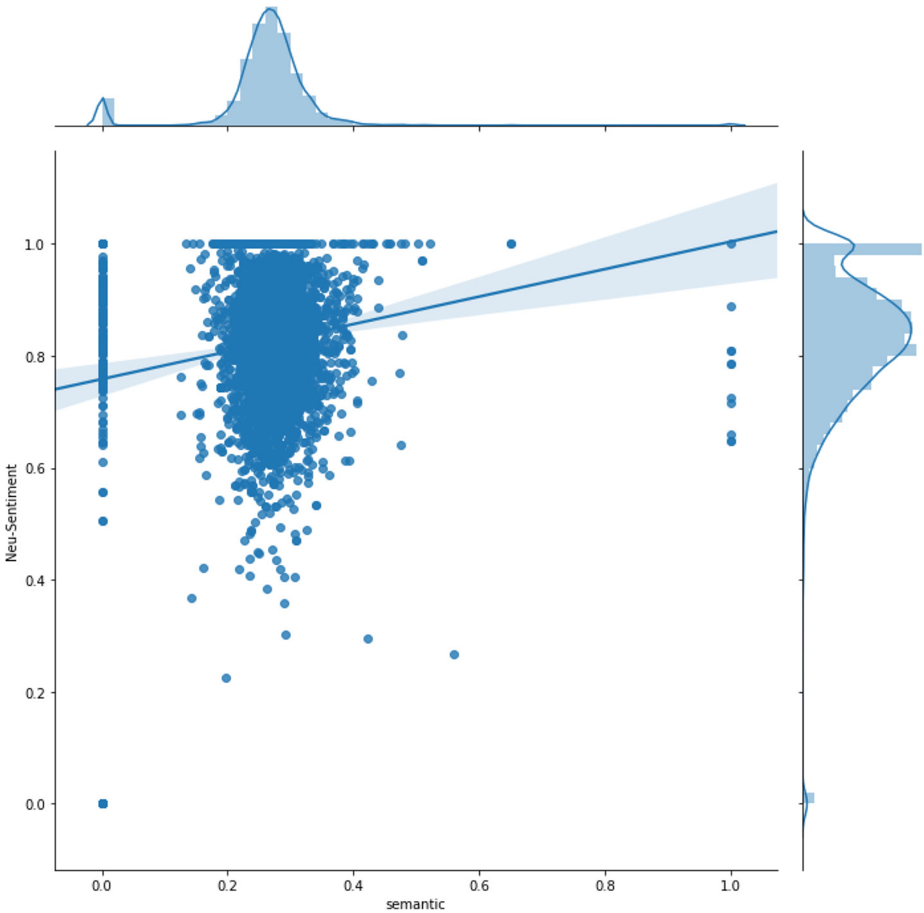

Figure 8.
Neutral sentiment

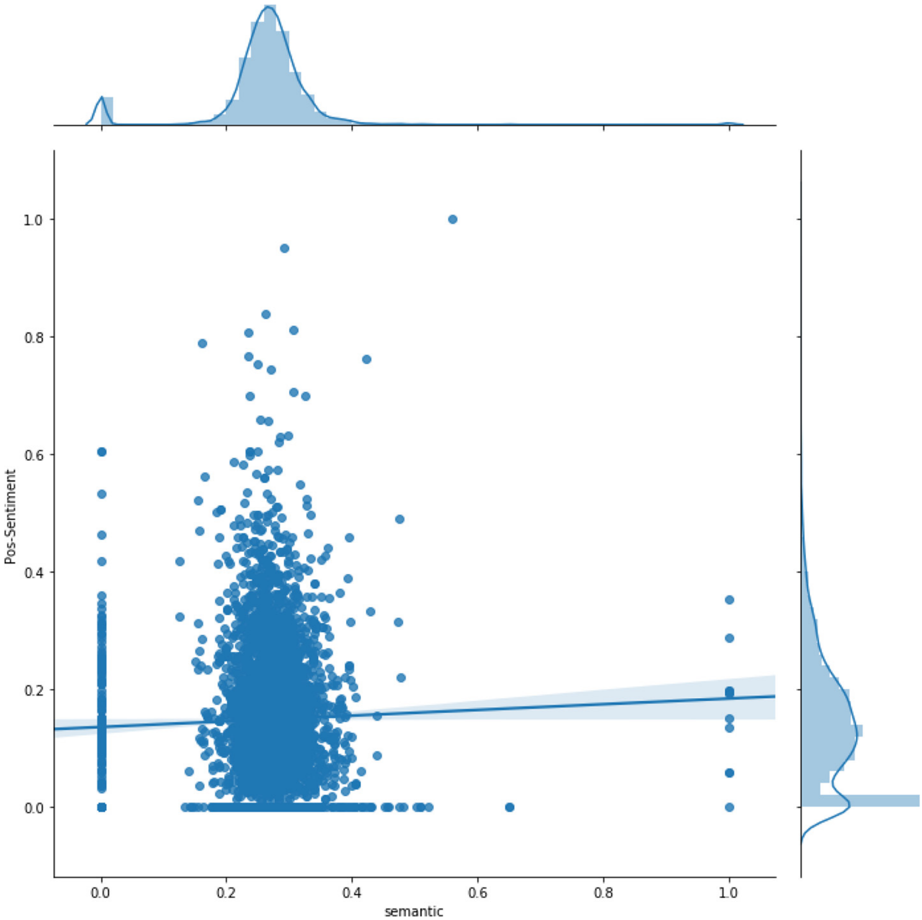

Figure 9.
Positive sentiment

When taking into consideration semantics, this feature gave improved results with sentiment feature sets. Negative sentiment, neutral sentiment, positive sentiment and compound sentiments show the increased values when the values of the semantic set are increased as shown in Figure 7-10, respectively.

Feature analysis shows that sentiment features based on semantics show significant impact among all other feature sets. All sentiment features are equally important and play a dominant role when combined with semantic features rather than other computed features.

5.3 Classifier results of different configurations

For splitting the data into training and testing, ten-fold cross validation and 70–30% holdout techniques are used. Machine learning classifiers including RF, SVM, MLP, DNN, bagging, J48 and AdaBoost are applied on the ranked features as shown in Tables 6 and 7, using *k*-fold cross-validation and hold-out settings. As evidenced from the results, DNN outperformed all other classifiers in terms of classification accuracy, as shown in Tables 6 and 7.

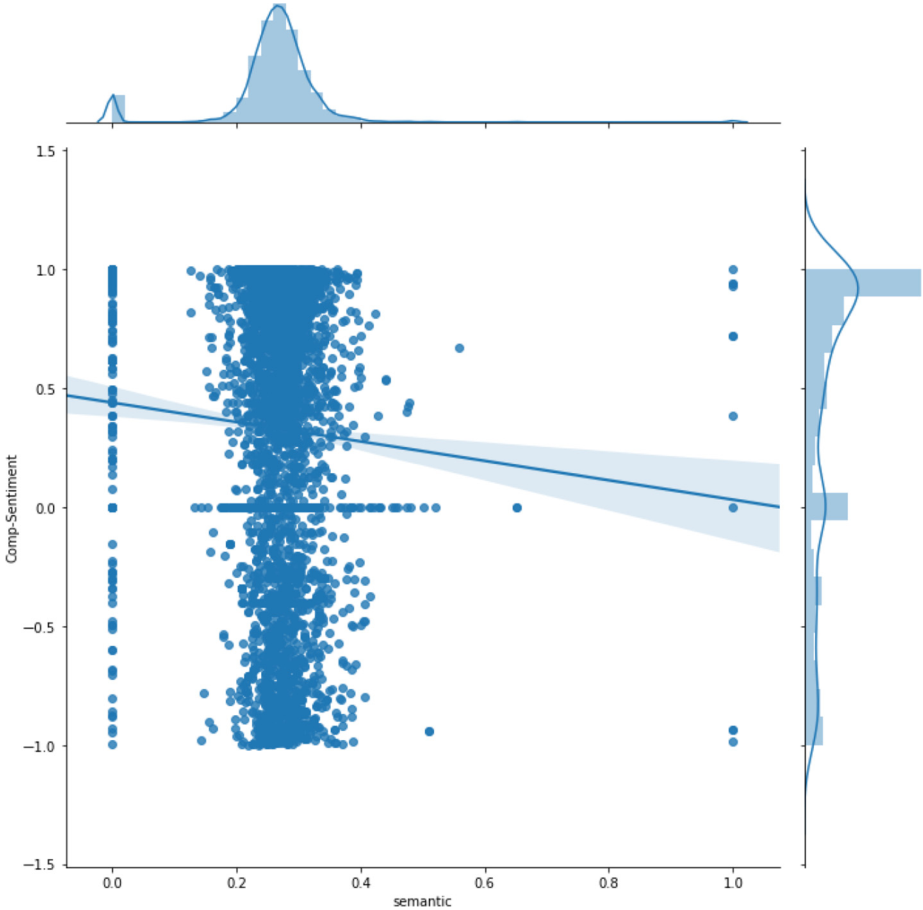

Figure 10.
Compound sentiments

Table 6.
Performance of classifiers with feature selection on ten-fold cross validation

	IG		GR		Relief-F	
	Acc (%)	F-measure	Acc (%)	F-measure	Acc (%)	F-measure
SVM	91.7	87.8	91.7	87.8	91.7	87.8
RF	93.9	93.1	94.1	93.5	93.0	91.6
J48	92.8	92.0	93.4	93.5	91.4	89.3
Bagging	93.1	91.8	93.4	93.0	92.3	90.1
MLP	91.7	87.8	91.7	87.8	91.7	87.8
DNN	96.7	93.5	95.3	93.9	93.7	92.0
AdaBoost	91.6	88.9	91.6	88.0	91.7	87.8

5.4 Feature-based classifiers results

Afterwards, machine learning algorithms are applied on computed features of the data set one by one. According to the results, the sentiment features results are more significant than all other features as shown in Table 8. Moreover, when DNN is applied with sentiment features to classify spam emails, DNN gave the highest accuracy of 97.2% as compared to the rest of other feature sets. Sentiment features also contributed in improving the performance of all features as shown in Figure 11.

5.5 Results of classifiers on a combination of different feature sets

To further validate the effectiveness of the proposed features, their accuracy in different combinations is carried out and classified. According to the results, when DNN is applied on all combinations, having sentiment features gave the highest accuracy than other combinations as shown in Figure 12.

5.6 Evaluation of classifiers performance based on receiver operating characteristics

The average execution time is also computed for the selected data set as shown in Figure 13. The results reveal that conventional classifiers including SVM, RF, J48 and bagging are efficient in terms of computational time. However, their accuracies are lower than DNN. DNN have the highest accuracy with the highest computation time too.

The receiver operating characteristics (ROC) curve shows the classification ability of the classifier. ROC is measured by the TP and FP rate. ROC and AUC are computed to evaluate the performance of applied classifiers as shown in Figure 14 and Table 9. DNN is the best classifier when applied with sentiment features to classify spam emails.

5.7 Comparison with other research papers

The proposed method is compared with existing techniques that used the CSDMC2010_SPAM data set to classify spam emails. The proposed feature sets-based classification model (FSEDM) with sentiment features achieved the best classification results, up to 97.2%, as shown in Table 10.

6. Conclusion

In this research study, a new feature sets-based classification model (FSEDM) is presented. The proposed feature set includes content, sentiment, semantic, user and spam lexicon. The proposed features are computed from the CSDMC2010_SPAM data set. Detailed

	IG Acc (%)	F-measure	GR Acc (%)	F-measure	Relief-F Acc (%)	F-measure
SVM	92.3	88.7	92.3	88.7	92.3	88.7
RF	93.6	92.7	93.6	92.8	93.6	92.1
J48	92.2	91.4	92.7	91.0	91.4	89.6
Bagging	92.3	90.7	92.8	91.9	92.8	90.4
MLP	92.3	88.7	92.3	88.7	92.8	88.7
DNN	96.6	95.2	94.0	93.2	96.6	94.8
AdaBoost	91.4	89.2	92.3	90.8	92.3	88.7

Table 7. Performance of classifiers with feature selection on 70–30% holdout

Feature set	Algorithms	Precision	Recall	Acc (%)	F-measure
All features	SVM	84.2	91.8	91.7	87.8
	RF	93.1	93.9	93.8	92.9
	J48	92.4	93.1	93.1	92.7
	Bagging	84.2	91.8	91.7	87.8
	MLP	84.2	91.8	91.7	87.8
	DNN	93.7	94.2	96.6	93.9
	AdaBoost	84.2	91.8	91.7	87.8
Content	SVM	91.8	91.8	91.7	91.8
	RF	92.6	93.8	94.0	93.0
	J48	92.2	93.1	93.0	92.4
	Bagging	91.5	92.9	92.8	91.5
	MLP	91.8	91.8	91.6	91.8
	DNN	95.2	93.8	95.1	94.4
	AdaBoost	91.8	91.8	91.7	91.8
Sentiment	SVM	91.8	91.8	91.7	91.8
	RF	90.3	90.3	92.0	90.3
	J48	91.8	91.8	91.6	91.8
	Bagging	90.8	91.9	91.9	88.4
	MLP	84.2	91.7	91.6	87.8
	DNN	94.8	95.7	97.2	95.0
	AdaBoost	91.8	91.8	91.6	91.8
Semantic	SVM	91.8	91.8	91.7	91.8
	RF	87.3	90.6	90.6	88.5
	J48	91.8	91.8	91.7	91.8
	Bagging	87.7	91.6	91.6	88.2
	MLP	91.8	91.8	91.7	91.8
	DNN	91.8	91.8	93.9	91.8
	AdaBoost	91.8	91.8	91.7	91.8
Spam lexicon	SVM	91.8	91.8	91.7	91.8
	RF	84.9	85.5	85.2	88.5
	J48	91.8	91.8	91.7	91.8
	Bagging	87.7	91.6	91.6	88.2
	MLP	91.8	91.8	91.7	91.8
	DNN	91.0	92.9	93.0	92.0
	AdaBoost	91.8	91.8	91.7	91.8
User	SVM	91.8	91.8	91.7	91.8
	RF	85.0	84.7	84.6	84.8
	J48	91.8	91.8	91.7	91.8
	Bagging	91.8	91.8	91.7	91.8
	MLP	91.8	91.8	91.7	91.8
	DNN	91.8	93.8	92.5	92.7
	AdaBoost	91.8	91.8	91.7	91.8

Table 8.
The results of machine learning algorithms on the features set

experiments are carried out to validate the proposed model. The results of performance evaluation measures verify that sentiment features played an important role in classification of spam emails. DNN outperformed all other classifiers and classified spam emails up to 97.2% when applied with sentiment features. Sentiment features contributed individually and also in combination with content-based features to improve the accuracy of classifiers. Thus, the role of sentiment-based features is more effective in classification of email into spam and ham.

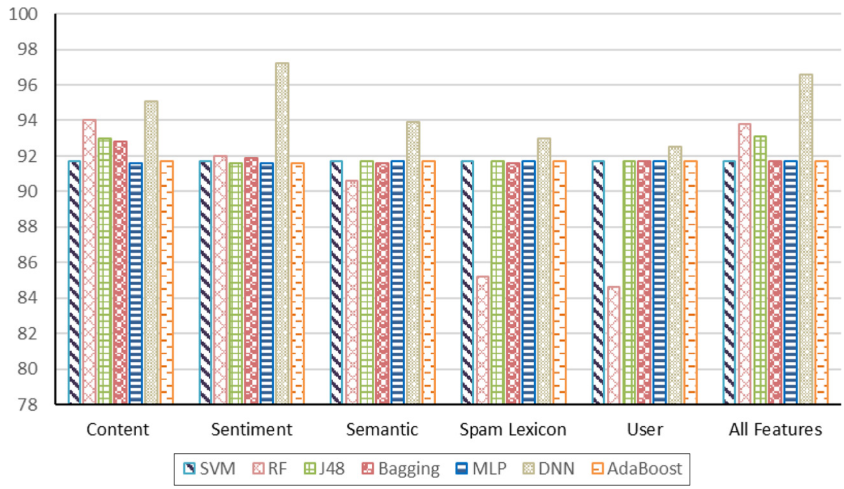

Figure 11.
Best classification accuracy of classifiers on each feature set

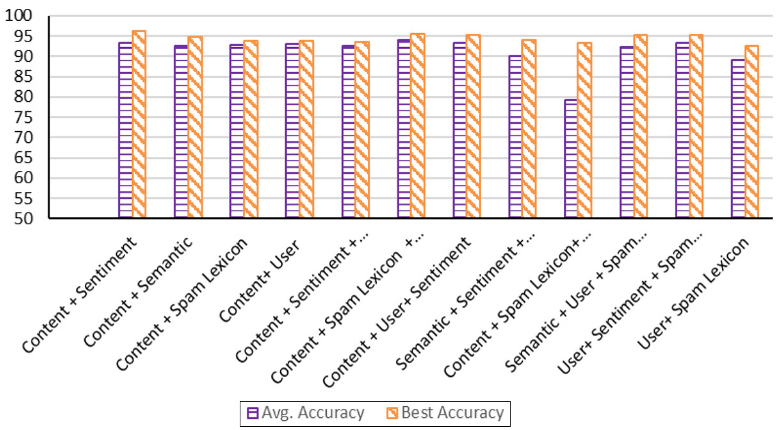

Figure 12.
Best and average accuracy on combinations of proposed features

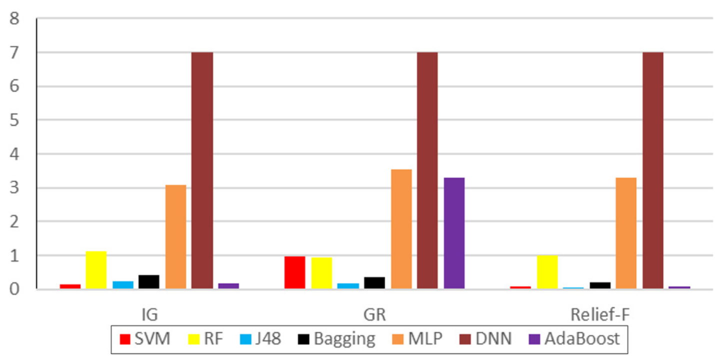

Figure 13.
Execution time of classifiers

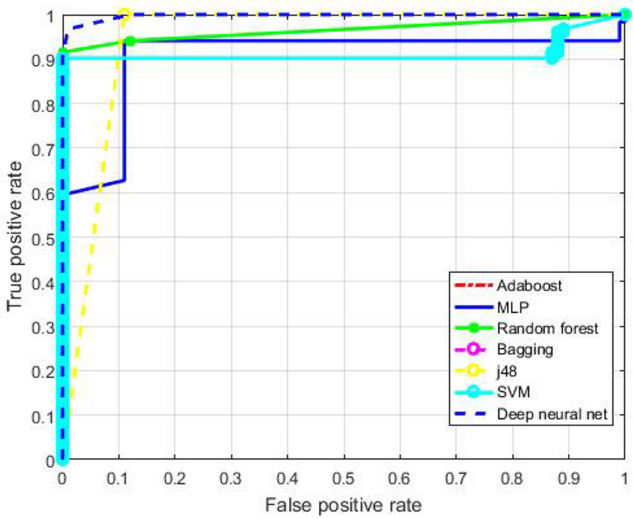

Figure 14.
ROC curve of the classifiers

Table 9.
Comparative analysis using AUC

Classifiers	AUC
DNN	98.0
MLP	95.0
SVM	94.0
RF	93.0
AdaBoost	92.0
J48	91.0
Bagging	88.0

Table 10.
Comparison with existing techniques

Technique	Highest accuracy (%)
Classifying spam emails using text and readability features (Shams and Mercer, 2013)	95.19
Content-based spam email filtering (Liu and Moh, 2016)	79.1
Voting-based classification for email spam detection (Al-Shboul *et al.*, 2016)	96.6
Proposed FSEDM	97.2

References

Akram, A.U., Khan, H.U., Iqbal, S., Iqbal, T., Munir, E.U. and Shafi, M. (2018), "Finding rotten eggs: a review spam detection model using diverse feature sets", *KSII Transactions on Internet and Information Systems*, Vol. 12 No. 10, pp. 5120-5142.

Al-Shboul, B.A., Hakh, H., Faris, H., Aljarah, I. and Alsawalqah, H. (2016), "Voting-based classification for e-mail spam detection", *Journal of ICT Research and Applications*, Vol. 10 No. 1, pp. 29-42.

Alqatawna, J., Faris, H., Jaradat, K., Al-Zewairi, M. and Adwan, O. (2015), "Improving knowledge based spam detection methods: the effect of malicious related features in imbalance data distribution", *International Journal of Communications, Network and System Sciences*, Vol. 8 No. 5, p. 118.

Alsmadi, I. and Alhami, I. (2015), "Clustering and classification of email contents", *Journal of King Saud University - Computer and Information Sciences*, Vol. 27 No. 1, pp. 46-57.

Balakumar, M. and Vaidehi, V. (2008), "Ontology based classification and categorization of email", *International Conference on Signal Processing, Communications and Networking (ICSCN '08)*, IEEE, pp. 199-202.

Barushka, A. and Hájek, P. (2016), "Spam filtering using regularized neural networks with rectified linear units", *Conference of the Italian Association for Artificial Intelligence*, Springer, pp. 65-75.

Barushka, A. and Hájek, P. (2018), "Spam filtering using integrated distribution-based balancing approach and regularized deep neural networks", *Applied Intelligence*, Vol. 48 No. 10, pp. 3538-3556.

Bhat, V.H., Malkani, V.R., Shenoy, P.D., Venugopal, K. and Patnaik, L. (2011), "Classification of email using beaks: behavior and keyword stemming", *TENCON IEEE Region 10 Conference*, IEEE, pp. 1139-1143.

Carmona-Cejudo, J.M., Baena-García, M., Del Campo-Avila, J. and Morales-Bueno, R. (2011), ""Feature extraction for multi-label learning in the domain of email classification", *IEEE Symposium on Computational Intelligence and Data Mining (CIDM '11)*, IEEE, pp. 30-36.

Cormack, G.V. (2008), "Email spam filtering: a systematic review", *Foundations and Trends® in Information Retrieval*, Vol. 1 No. 4, pp. 335-455.

Cormack, G.V. and Lynam, T.R. (2005), "TREC 2005 spam track overview", *TREC '05*, pp. 500-274.

Dai, J. and Xu, Q. (2013), "Attribute selection based on information gain ratio in fuzzy rough set theory with application to tumor classification", *Applied Soft Computing*, Vol. 13 No. 1, pp. 211-221.

Esmaeili, M., Arjomandzadeh, A., Shams, R. and Zahedi, M. (2017), "An anti-spam system using naïve Bayes method and feature selection methods", *International Journal of Computer Applications*, Vol. 165 No. 4, pp. 1-5.

Faris, H., Al-Zoubi, A.M., Heidari, A.A., Aljarah, I., Mafarja, M., Hassonah, M.A. and Fujita, H. (2019), "An intelligent system for spam detection and identification of the most relevant features based on evolutionary random weight networks", *Information Fusion*, Vol. 48, pp. 67-83.

Feng, W., Sun, J., Zhang, L., Cao, C. and Yang, Q. (2016), "A support vector machine based naïve Bayes algorithm for spam filtering", *IEEE 35th International Performance Computing and Communications Conference (IPCCC '16)*, IEEE, pp. 1-8.

Firte, L., Lemnaru, C. and Potolea, R. (2010), "Spam detection filter using KNN algorithm and resampling", *IEEE International Conference on Intelligent Computer Communication and Processing (ICCP '10)*, IEEE, pp. 27-33.

Gaikwad, B.U. and Halkarnikar, P. (2014), "Random Forest technique for r-mail classification", *International Journal of Scientific and Engineering Research*, Vol. 5 No. 3, pp. 145-153.

George, P. and Vinod, P. (2018), "Composite email features for spam identification", *Cyber Security: Proceedings of CSI '15*, Springer, pp. 281-289.

Hijawi, W., Faris, H., Alqatawna, J., Al-Zoubi, A.M., I. and Aljarah, I. (2017), "Improving email spam detection using content based feature engineering approach", *IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT '17)*, Aqaba, pp. 1-6, doi: 10.1109/AEECT.2017.8257764.

Idris, I. and Selamat, A. (2014), "Improved email spam detection model with negative selection algorithm and particle swarm optimization", *Applied Soft Computing*, Vol. 22, pp. 11-27.

Idris, I., Selamat, A. and Omatu, S. (2014), "Hybrid email spam detection model with negative selection algorithm and differential evolution", *Engineering Applications of Artificial Intelligence*, Vol. 28, pp. 97-110.

Islam, R. and Xiang, Y. (2010), "Email classification using data reduction method", *5th International ICST Conference on Communications and Networking in China (CHINACOM)*, IEEE, pp. 1-5.

Jiang, J.J. and Conrath, D.W. (1997), "Semantic similarity based on corpus statistics and lexical taxonomy", arXiv preprint cmp-lg/9709008.

Khan, H.U. (2017), "Mixed-sentiment classification of web forum posts using lexical and non-lexical features", *Journal of Web Engineering*, Vol. 16 Nos. 1/2, pp. 161-176.

Khan, H.U. and Daud, A. (2017), "Using machine learning techniques for subjectivity analysis based on lexical and nonlexical features", *International Arab Journal of Information Technology*, Vol. 14 No. 4, pp. 481-487.

Lecun, Y., Bengio, Y. and Hinton, G. (2015), "Deep learning", *Nature*, Vol. 521 No. 7553, pp. 436.

Li, W., Meng, W., Tan, Z. and Xiang, Y. (2014), "Towards designing an email classification system using multi-view based semi-supervised learning", *IEEE 13th International Conference on, Trust, Security and Privacy in Computing and Communications (TrustCom)*, IEEE, pp. 174-181.

Liu, P. and Moh, T.-S. (2016), "Content based spam e-mail filtering", *International Conference on Collaboration Technologies and Systems (CTS '16)*, IEEE, pp. 218-224.

Méndez, J.R., Cotos-Yañez, T.R. and Ruano-Ordás, D. (2019), "A new semantic-based feature selection method for spam filtering", *Applied Soft Computing*, Vol. 76, pp. 89-104.

Méndez, J.R., Reboiro-Jato, M., Díaz, F., Díaz, E. and Fdez-Riverola, F. (2012), "Grindstone4Spam: an optimization toolkit for boosting e-mail classification", *Journal of Systems and Software*, Vol. 85 No. 12, pp. 2909-2920.

Mirza, N., Patil, B., Mirza, T. and Auti, R. (2017), "Evaluating efficiency of classifier for email spam detector using hybrid feature selection approaches", *International Conference on Intelligent Computing and Control Systems (ICICCS '17)*, IEEE, pp. 735-740.

Moh, T.-S. and Lee, N. (2011), "Reducing classification times for email spam using incremental multiple instance classifiers", *International Conference on Information Intelligence, Systems, Technology and Management*, Springer, pp. 189-197.

Mohammed, M.A., Gunasekaran, S.S., Mostafa, S.A., Mustafa, A. and Ghani, M.K.A. (2018), ""Implementing an agent-based multi-natural language anti-spam model", *International Symposium on Agent, Multi-Agent Systems and Robotics (ISAMSR '18)*, IEEE, pp. 1-5.

Pérez-Díaz, N., Ruano-Ordas, D., Méndez, J.R., Galvez, J.F. and Fdez-Riverola, F. (2012), "Rough sets for spam filtering: Selecting appropriate decision rules for boundary e-mail classification", *Applied Soft Computing*, Vol. 12 No. 11, pp. 3671-3682.

Popovac, M., Karanovic, M., Sladojevic, S., Arsenovic, M. and Anderla, A. (2018), "Convolutional neural network based SMS spam detection", *26th Telecommunications Forum (TELFOR '18)*, IEEE, pp. 1-4.

Rayan, A., Nirmal, N., Sohn, K.A. and Chung, T.S. (2017), "A graph model based feature set selection from short texts with application to document novelty detection", *Intelligent Data Analysis*, Vol. 21 No. 5, pp. 1117-1139.

Renuka, K.D. and Visalakshi, P. (2014), "Latent semantic indexing based SVM model for email spam classification", *Journal of Scientific and Industrial Research*, Vol. 73 No. 7, pp. 437-442.

Ruano-Ordás, D., Fdez-Riverola, F. and Méndez, J.R. (2018), "Using evolutionary computation for discovering spam patterns from e-mail samples", *Information Processing and Management*, Vol. 54 No. 2, pp. 303-317.

Shams, R. and Mercer, R.E. (2013), "Classifying spam emails using text and readability features", *IEEE 13th International Conference on Data Mining (ICDM '13)*, IEEE, pp. 657-666.

Shams, R. and Mercer, R.E. (2016), "Supervised classification of spam emails with natural language stylometry", *Neural Computing and Applications*, Vol. 27 No. 8, pp. 2315-2331.

Singh, M. (2019), “Classification of spam email using intelligent water drops algorithm with naïve Bayes classifier”, *Progress in Advanced Computing and Intelligent Engineering*, Springer.

Sohn, K.A. and Chung, T.-S. (2015), “A graph model based author attribution technique for single-class e-mail classification”, *IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS ‘15)*, IEEE, pp. 191-196.

Song, M.H. (2013), “E-mail classification based learning algorithm using support vector machine”, *Applied Mechanics and Materials*, Vols 268/270, pp. 1844-1848, Trans Tech Publications Ltd.

Spolaôr, N., Cherman, E.A., Monard, M.C. and Lee, H.D. (2013), “Relief-F for multi-label feature selection”, *Brazilian Conference on Intelligent Systems (BRACIS ‘13)*, IEEE, pp. 6-11.

Thomas, K., Grier, C., Ma, J., Paxson, V. and Song, D. (2011), “Design and evaluation of a real-time URL spam filtering service”, in *The IEEE Symposium on Security and Privacy (SP ‘11)*, IEEE, pp. 447-462.

Uysal, A.K. (2016), “An improved global feature selection scheme for text classification”, *Expert Systems with Applications*, Vol. 43, pp. 82-92.

Varghese, R. and Dhanya, K. (2017), “Efficient feature set for spam email filtering”, *IEEE 7th International Advance Computing Conference (IACC ‘17)*, IEEE, pp. 732-737.

Yakovlev, E. (2018), “Spam indication through machine learning structure study”, НАУЧНО-ПРАКТИЧЕСКИЕ ИССЛЕДОВАНИЯ, p. 42.

Zhang, Y., Wang, S., Phillips, P. and Ji, G. (2014), “Binary PSO with mutation operator for feature selection using decision tree applied to spam detection”, *Knowledge-Based Systems*, Vol. 64, pp. 22-31.

Zhuang, X., Zhu, Y., Chang, C.-C. and Peng, Q. (2017), “Feature bundling in decision tree algorithm”, *Intelligent Data Analysis*, Vol. 21 No. 2, pp. 371-383.

Corresponding author
Hikmat Ullah Khan can be contacted at: hikmat.ullah@ciitwah.edu.pk