Wireless Personal Communications (2022) 123:1913–1923
https://doi.org/10.1007/s11277-021-09221-5

A Collaborative Abstraction Based Email Spam Filtering with Fingerprints

P. Rajendran[1] · A. Tamilarasi[2] · R. Mynavathi[3]

Accepted: 15 October 2021 / Published online: 2 November 2021

Abstract
Spam detection in emails tends to be an endless research interest among many researchers and academicians. Even though email communication has become a major role in day to day activities, the increasing volumes of threats towards spam emails has paved the way for numerous email spam detection techniques. Many spam filtering methods including data mining and machine learning techniques are adopted by researchers; yet a complete accurate filtering model is an expected solution to cope up with the intentional spam attacks. This paper proposes one such model that uses a hybrid approach towards efficient spam detection. A collaborative spam filtering framework using abstraction of the entire email layout and the fingerprints of the layout is proposed to match and catch the sprouting nature of spam. Collaborative framework uses recommendations from other users to create spam database. Any incoming mail is checked against the spam database for spam or ham classification using near duplicate similarity matching scheme. To reduce false positive and false negative ratio in spam classification, we calculate cumulative weights from both email layouts and fingerprints. Fingerprint signatures of newly classified spam are progressively updated to the spam database for up-to-date spam detection. The system is evaluated with Spam Assassin dataset and the results are proven for a comparatively better performance.

Keywords Collaborative spam · Near duplicate · Email abstraction · Reputation · Spam filtering

✉ P. Rajendran
prnavya@gmail.com

A. Tamilarasi
drtamil@kongu.ac.in

R. Mynavathi
rpgmyna@gmail.com

[1] Department of Computer Applications, Velalar College of Engineering and Technology, Thindal, Erode, Tamilnadu, India

[2] Department of Computer Applications, Kongu Engineering College, Perundurai, Erode, Tamilnadu, India

[3] Department of Information Technology, Velalar College of Engineering and Technology, Thindal, Erode, Tamilnadu, India

1 Introduction

Spam is generally an unsolicited electronic junk mail that includes text messages, images or videos and is sent without the consent of the recipients. Spam messages are broadcasted to large number of email users occupying larger bandwidth. It is not only obstructing the network traffic but also forms a base for email viruses and denial of service attacks. Moreover, spam messages contain mostly offensive and fraudulent texts that are unpleasant to the recipients. Email users are drowned with nearly 50% of spam messages with new content and new addresses in their inbox daily. Spam messages may destroy email servers with potentially harmful information and the users need to spend certain amount of time to identify and analyze spam messages in their inbox and delete them. Several spam filtering techniques are proposed to identify solicited and unsolicited messages; however, email spammers use dynamic new structures to thwart all the techniques and conceal email content. The main problem that arises with spam is that spammers devise new ways to attack the spam filters and thereby benefit from sending large amount of spams. The primary challenge is to develop a system that can deal with newly arising spams. Existing spam filtering techniques are either list-based filters or content-based filters or a collaborative response system which generally identify duplicate contents, fraudulent texts or the disreputable servers. Though these filters provide better accuracy rate in spam classification, they are prone to erroneous misclassification of hams as spams. This paper proposes a collaborative Spam detection system that uses email layouts and fingerprints to identify spam messages. Collaborative approach collects the feedback from the users regarding what mails are spams and consequently develops a model against it. The incoming emails are mapped to a known spam database using near duplicate matching scheme. Overall three key processes are involved in this spam detection approach. First, a layout abstraction set is generated from the HTML content of the ENTIRE email. Secondly, from the abstraction set, fingerprints are generated and weights are calculated for each fingerprint. Finally, the near duplicate matching process is carried out with the generated fingerprints and the detection of spam is carried out.

2 Related Work

Email spam filtering is an exploring area due to the increasing nature of spam emails. Various technologies [1, 2] have been proposed to classify legitimate mails and spam messages. Depending upon how the techniques work, previous spam detection mechanisms can be categorized under three divisions.

(i) List Based Filtering
(ii) Content Based Filtering
(iii) Other Filtering methods.

List based filtering methods blocks or allows the email messages by categorizing email users as spammers or trusted users. Content based filtering works by evaluating words and sentences extracted from the email to classify under ham or spam category [3–6]. Such techniques include word-based filters, rule-based filters and probability-based filters. Naive Bayes and SVM classification methods falls under content-based filtering. Naïve Bayes

[7–12] trains the filtering model using classified emails with probability value for each suspicious word. Support Vector Machine based models [10, 13, 14] are supervised learning models. Various other content-based classification techniques include Markov random field model [15–17], Regression models [18, 19], and Neural network models [20, 21].

Other filtering methods such as Collaborative Spam filtering [22–29] make use of users' collective feedback reports to check for spam messages. It takes input from millions of email users. Every incoming mail is flagged as spam or ham and is reported to a central spam database. For every new spam encountered, it should be reported to the spam database by some users. Subsequent users receiving emails can query with the spam database to decide whether the message is already marked as spam. P2P-based architecture [30, 31], centralized server-based system [27, 32] are generally representatives of collaborative filtering methods.

In [22, 29, 30], digest technique generates a 32-byte code to represent the distribution of word trigrams in e-mail. Multiple digests produced from strings of fixed length sampled at random positions are discussed in [23, 28, 33]. A fingerprint-based feature vector obtained from set of checksums over a substring is proposed in [34, 35]. Collaborative method of spam filtering is given in [25]. Most of the methods generate email abstraction based on text content [36–41]. However, if a random paragraph is inserted, these methods will fail to capture such intentional spam mails. This raises the need to devise a better approach that withstands the cunning nature of spam messages.

3 Proposed Methodology

The proposed methodology introduces a new process of generating an abstraction from the email. Instead of generating the tag sequence from the message content as discussed in [25], the aim is to generate the abstraction for the entire email layout using the HTML tags. Both the < head > and < body > tags of the email content are processed to generate the abstract. Individual tags are processed to a new tag. The generated new tags are then used to generate the fingerprints. Fingerprint generation serves two purposes.

(i) To uniquely abstract the email tags and minimize the storage space.
(ii) For efficient duplication detection.

The fingerprints of the generated abstracts are manipulated for near duplicate matching process. The classification of Spam and Ham is finally done in the Spam Detection process. The proposed model of the system is illustrated in Fig. 1.

3.1 Email Layout Abstraction

Each email is represented by its layout. Since all emails follow MIME format, HTML tags are available in MIME text/html format and thus adequate information related to structure of email can be obtained. In Email layout abstraction, each paragraph is represented as a sequence of tags. Duplicate emails are identified by comparing the HTML tag sequence of the two emails. As the initial process, tag preprocessing is done to identify and remove the tags that are in common. It also prevents spam insertion. This process eliminates mismatched non-empty tags and also tags with no corresponding end tags. The procedure for generating layout abstraction is as follows.

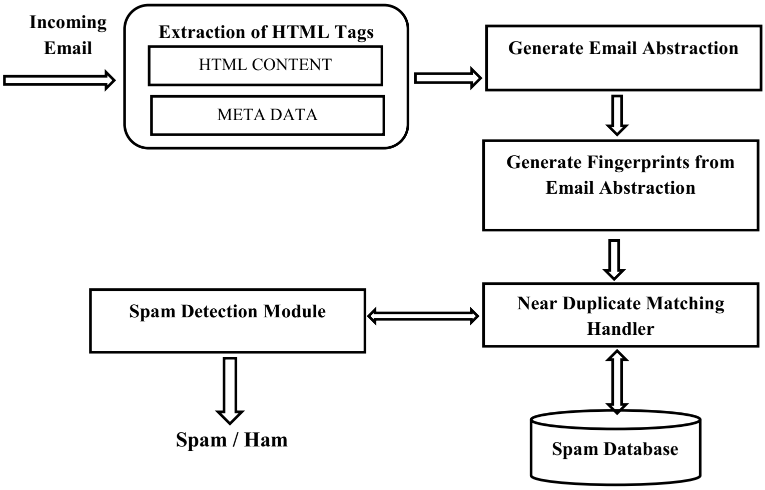


Fig. 1 Abstraction based fingerprint model for Spam detection

```
Algorithm 1: Generation of LayoutAbstractionSet
Step 1: Extract the name of each HTML tag separately.
Step 2: Eliminate tag attributes and their values.
Step 3: Convert the user text message within each paragraph to a newly defined tag called <texttag/>
Step 4: Segregate all the anchor tags and retain the anchor tag values separately.
Step 5: Combine set of all tags generated from step1 and step 3 as LayoutAbstractionSet(LAS).
Step 6: Combine anchor tags under AnchorTagSet(ATS).
Step 7: Preprocess the tag sequence in LAS.
Step 8: for each tag in LAS:
Step 9:      assign new tag position
Step 10:     rearrange the tags in accordance with new position
Step 11: Concatenate all tags in the new position
Step 12: Append LAS with ATS
Step 13: Return new LAS
```

Mainly the emails fall under three categories. The first type of email is collected as feedback from the users and is named as reported spams. The second category emails are the one that are matched against the reported spams. These emails are called as testing emails. Finally, misclassified hams are also to be monitored and hence Email layout abstraction is done for all the three categories of emails.

3.2 Fingerprint and Weightage calculation

Fingerprints are technologies that are mainly used for biometric authorization. When used in data and information retrieval area, the fingerprints provide a precise short way of representing large information content. They are in fact tags of shorter length for a longer text. It has the advantage of near duplicate matching with minute variations. In spam detection, fingerprints are used as a digest value to represent large spam contents. After the generation of email layouts, fingerprints are generated over the segregated layout abstraction set. Fingerprints are generated functions denoted as $f(x) = k(i) \rightarrow \{0{,}1\}^{len}$, where $k(i)$ denotes the tags of interest and len represents the length of a tag sequence. Rabin Fingerprinting scheme is used to generate the fingerprint from the abstraction. It works with an irreducible polynomial p of certain degree d.Having p, the fingerprint for a LAS will be computed with a function $fun(LAS) = LAS(t)modp(t)$.

The algorithmic steps for fingerprint generation are shown in Algorithm 2.

Algorithm 2: Fingerprint generation for LayoutAbstractionSet

```
Step 1: for each division of LayoutAbstractionSet
Step 2:     Generate fingerprint using Rabin Algorithm
Step 3:     Compute a threshold value for each fingerprint
Step 4: Calculate the compound weight of all fingerprints
```

An email message is partitioned into various components (em) with individual weights (w). For each component, em, fingerprint is generated (fp(em)). The final weight (fw)for the entire email message (em) is computed as

$$fw(em) = \sum_{i=1}^{n} w^{i} \sum_{j=1}^{m} fp(em)^{i}$$

3.3 Near duplicate Matching Process

Matching process compares each testing email with a known spam database. An indicator score for each email is calculated and spam mails are identified with a threshold value.

$$Spam(em) = \left\{ \begin{array}{l} Spam,\ if\, fw(em) > t \\ Ham,\ if\, fw(em) \leq t \end{array} \right\}$$

The fingerprints generated from the email components are maintained in a tree data structure. Here, we define a tree to be an ordered data structure that stores the fingerprints as dynamic sets with leaf nodes as strings. Initially the tree will be empty. After the extraction of fingerprints, insertion happens if the fingerprints are not already in the tree. The tree is built in the training phase with the known spam database fingerprints. Fingerprints of each abstraction is collected and put into the leaf nodes to the root nodes of the tree (ie) from low level of the tree to the higher level. A traversal from the root node to a leaf node demonstrates one single abstraction fingerprint. Matching of the email abstraction fingerprint is done from the root node to the leaf node. Matching process is also checked with the cumulative weights of the fingerprints compared with a threshold value.

3.4 Reputation Evaluation

The main goal of collaborative email spam detection is to build a known spam database collecting feedback from different users. This known spam database is used to match the incoming mails and further block the near duplicate spams. To validate the feedback of the users regarding the truthfulness of spams reputation evaluation is carried out for each reporter.

- A score is assigned to each reporter when he reports a spam fingerprint for the first time.
- For each reporter, current reputation score is calculated with respect to the last score and current feedback multiplied with a weight.
- If the feedback is found to be true, the reputation score is incremented by a small value.
- If the feedback if found to be false, the reputation score is decremented by half of its value.

Considering the multiplication factor as β, the reputation is calculated as follows

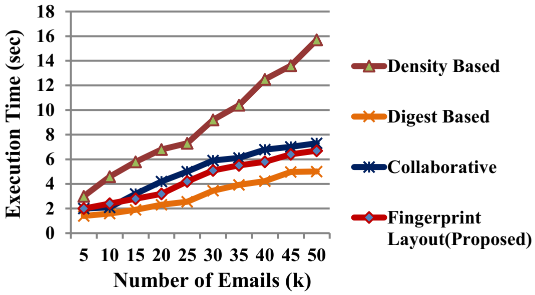
$$Repute(em) = Repute(em - 1) + \beta \times feedback(em)$$

To avoid errors in reputation evaluation, the score is incremented in a minimum value and for false positive errors it is dropped drastically.

4 Experiments and Results

The proposed system model is experimented with a well-known spam system, SpamAssassin. It is assumed that there are about 45,000 spams per day and 5,000 legitimate mails in the data set. The model is compared with density based, digest based and collaborative methods. The authors adopt the email representation method with different parts of the email. Time of execution of the model and the accuracy in detecting spams

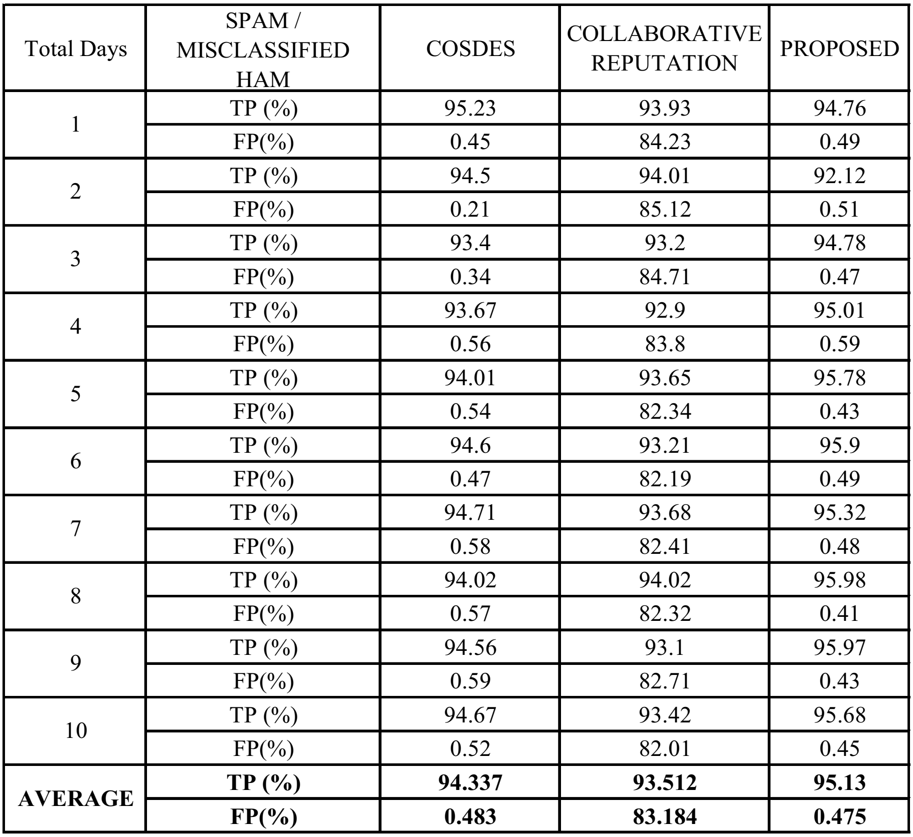
Fig. 2 Execution Time analysis with varied number of emails



are evaluated. Execution time of generating the email abstraction layout is compared with the other methods and the results are shown in Fig. 2.

The detection accuracy of spam and non spam is also experimented and is compared with the other two approaches. The experiment is set to check whether the proposed spam detection system is withstanding intended malicious attacks. In collaborative approach a set of already reported spams are inserted in to the database and the near duplicate matching scheme is processed. The model is evaluated for 10 days spam and the True Positive (which is a real spam) rate and False Positive (a misclassified ham as spam) rate is observed. The table values in Fig. 3 lists the TP and FP rates for 10 days.

As seen in Fig. 3, the proposed model reports on average 95.13% of True Positive rate and 0.511% of False Positive rate. This shows an efficient performance when compared to the other two schemes. Though Collaborative Reputation method has reported highest True Positive rate, its highest False Positive rate of 83.184% is highly unacceptable. When compared with COSDES, the proposed model achieves better performance both in True Positive rate and False Positive rate.



Total Days	SPAM / MISCLASSIFIED HAM	COSDES	COLLABORATIVE REPUTATION	PROPOSED
1	TP (%)	95.23	93.93	94.76
	FP(%)	0.45	84.23	0.49
2	TP (%)	94.5	94.01	92.12
	FP(%)	0.21	85.12	0.51
3	TP (%)	93.4	93.2	94.78
	FP(%)	0.34	84.71	0.47
4	TP (%)	93.67	92.9	95.01
	FP(%)	0.56	83.8	0.59
5	TP (%)	94.01	93.65	95.78
	FP(%)	0.54	82.34	0.43
6	TP (%)	94.6	93.21	95.9
	FP(%)	0.47	82.19	0.49
7	TP (%)	94.71	93.68	95.32
	FP(%)	0.58	82.41	0.48
8	TP (%)	94.02	94.02	95.98
	FP(%)	0.57	82.32	0.41
9	TP (%)	94.56	93.1	95.97
	FP(%)	0.59	82.71	0.43
10	TP (%)	94.67	93.42	95.68
	FP(%)	0.52	82.01	0.45
AVERAGE	**TP (%)**	**94.337**	**93.512**	**95.13**
	FP(%)	**0.483**	**83.184**	**0.475**

Fig. 3 Accuracy Evaluation of Spam detection

5 Conclusion

The field of collaborative spam detection represents reported spams and the near duplicate matching process for efficient spam detection and filtering. Email content alone should not be taken into consideration for near duplicate matching as the evolving nature of spams varies with respect to email layouts. This paper explores the enhanced version of email layout abstraction method combined with fingerprint generation for near duplicate matching. This enhanced feature can more effectively capture the cunning spams. The spam database is also updated with the newly evolving spams and is kept up-to-date for blocking subsequent spams. Experimental results prove the effectiveness of the proposed method recommending this approach for efficient spam detection.

Authors' Contribution All the authors made substantial contribution to the conception of the work.

Funding The authors did not receive support from any organization for the submitted work.

Data Availability The data that support the findings of this study are available from the public corpus of spamassassin.apache.org.

Declarations

Conflict of interest The authors have no conflicts of interest to declare that are relevant to the content of this article. The authors did not receive support from any organization for the submitted work.

Code availability (software application or custom code).

The algorithm of the proposed work is included in this article itself.

References

1. Dada, E. G., Bassi, J. S., Chiroma, H., Abdulhamid, S. M., Adetunmbi, A. O., & Ajibuwa, O. M. (2019). Machine learning for email spam filtering: Review, approaches and open research problems. *Heliyon*. https://doi.org/10.1016/j.heliyon.2019.e01802.
2. Radovanovic, D., & Krstajic, B. (2018). Review spam detection using machine learning. In *23rd International Scientific-Professional Conference on Information Technology (IT)*, 1–4, https://doi.org/10.1109/SPIT.2018.8350457.
3. Liu, P., & Moh, T. (2016). Content based spam e-mail filtering. In *International Conference on Collaboration Technologies and Systems (CTS)*, 218–224, https://doi.org/10.1109/CTS.2016.0052.
4. Sokolov, M., Olufowobi, K., and Herndon, N. (2020). Visual spoofing in content-based spam detection. In *13th International Conference on Security of Information and Networks (SIN 2020)*. Association for Computing Machinery, 1–5. https://doi.org/10.1145/3433174.3433605.
5. Shyry, P., & Jinila, B. (2021). Detection and prevention of spam mail with semantics-based text classification of collaborative and content filtering. *Journal of Physics: Conference Series., 1770*, 012031. https://doi.org/10.1088/1742-6596/1770/1/012031
6. Wang, S., Zhang, X., Cheng, Y., Jiang, F., Yu, W., & Peng, J. (2018). A fast content- based spam filtering algorithm with fuzzy-SVM and K-means. *IEEE International Conference on Big Data and Smart Computing (BigComp)*. https://doi.org/10.1109/BigComp.2018.00051.
7. Anitha, P. U. & Rao, C. V. G. & Babu, S. (2017). Email spam classification using neighbor probability based Naïve Bayes algorithm. In *7th International Conference on Communication Systems and Network Technologies (CSNT)*, 350–355.https://doi.org/10.1109/CSNT.2017.8418565

8. Ma, T.M., Yamamori, K., & Thida, A. (2020). A comparative approach to Naïve Bayes classifier and support vector machine for email spam classification. In *IEEE 9th Global Conference on Consumer Electronics (GCCE)*, 324–326, https://doi.org/10.1109/GCCE50665.2020.9291921
9. Peng, W., Huang, L., Jia, J., & Ingram, E. (2018). Enhancing the Naive Bayes spam filter through intelligent text modification detection. In *17th IEEE International Conference on Trust, Security and Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*. 849–854, https://doi.org/10.1109/TrustCom/BigDataSE.2018.00122.
10. Gupta, P., Dubey, R. K., Dr. Mishra, S. (2019). Detecting Spam emails/sms using naive bayes and support vector machine. International Journal of Scientific & Technology Research, 8(11)
11. Samsudin, N., Foozy, M., Feresa, C., Alias, N., Shamala, P., Othman, N., Din, W., & Sofiah, W. I. (2019). Youtube spam detection framework using naïve bayes and logistic regression. *Indonesian Journal of Electrical Engineering and Computer Science., 14*, 1508–1517.
12. Santoshi, K.U., Bhavya,S.S., Sri, Y.B., & Venkateswarlu, B. (2021). Twitter spam detection using naïve bayes classifier. In *6th International Conference on Inventive Computation Technologies (ICICT)*, 773–777. https://doi.org/10.1109/ICICT50816.2021.9358579.
13. Ahmad, S. B. S., Rafie, M., & Ghorabie, S. M. (2021). Spam detection on Twitter using a support vector machine and users' features by identifying their interactions. *Multimedia Tools and Applications, 80*, 11583–11605. https://doi.org/10.1007/s11042-020-10405-7.
14. Mishra, S., & Malathi, D. (2017). Behaviour analysis of SVM based spam filtering using various parameter values and accuracy comparison. *International Conference on Computing Methodologies and Communication (ICCMC), 2017*, 27–31. https://doi.org/10.1109/ICCMC.2017.8282698
15. Mahdi, W., Aziz, Q., Manel, M., & Florence, S. (2017). A topic-based hidden Markov model for real-time spam tweets filtering. *Procedia Computer Science, 112*, 833–843. https://doi.org/10.1016/j.procs.2017.08.075
16. El-Mawass, N., Honeine, P., & Vercouter, L. (2020). SimilCatch: Enhanced social spammers detection on Twitter using Markov Random Fields. *Information Processing & Management*. https://doi.org/10.1016/j.ipm.2020.102317
17. Wang, Z., Hu, R., Chen, Q., Gao, P., & Xu, X. (2020). ColluEagle: Collusive review spammer detection using Markov random fields. *Data Mining and Knowledge Discovery., 34*, 1621–1641. https://doi.org/10.1007/s10618-020-00693-w
18. Dedeturk, B. K., & Akay, B. (2020). Spam filtering using a logistic regression model trained by an artificial bee colony algorithm. *Applied Soft Computing*. https://doi.org/10.1016/j.asoc.2020.106229
19. Wijaya, A., & Bisri, A. (2016). Hybrid decision tree and logistic regression classifier for email spam detection. In *8th International Conference on Information Technology and Electrical Engineering (ICITEE)*, 1–4. https://doi.org/10.1109/ICITEED.2016.7863267
20. Madisetty, S., & Desarkar, M. S. (2018). A neural network-based ensemble approach for spam detection in twitter. *IEEE Transactions on Computational Social Systems, 5*(4), 973–984. https://doi.org/10.1109/TCSS.2018.2878852
21. Sharmin, T., Di Troia, F., Potika, K., & Stamp, M. (2020). Convolutional neural networks for image spam detection. *Information Security Journal: A Global Perspective, 29*(3), 103–117. https://doi.org/10.1080/19393555.2020.1722867
22. AlMahmoud, A., Damiani, E., Otrok, H., & Al-Hammadi, Y. (2019). Spamdoop: A privacy-preserving big data platform for collaborative spam detection. *IEEE Transactions on Big Data, 5*(3), 293–304. https://doi.org/10.1109/TBDATA.2017.2716409.
23. Azad, M. A., Bag, S., Tabassum, S., & Hao, F. (2020). Privy: Privacy preserving collaboration across multiple service providers to combat telecom spams. *IEEE Transactions on Emerging Topics in Computing, 8*(2), 313–327.
24. Balika, J., & Chelliah., Anand, Sasidharan., Dharmesh, Kumar, Singh., & Nilesh, Dangi. (2021). Collaborative and early detection of email spam using multitask learning. *International Journal of Performability Engineering, 17*(6), 528–535.
25. Chen, M., Sung, P., & Tseng, C. (2011). Cosdes: A collaborative spam detection system with a novel E-Mail abstraction scheme. *IEEE Transactions on Knowledge & Data Engineering, 23*(5), 669–682. https://doi.org/10.1109/TKDE.2010.147
26. Guo, Z., Shen, Yu., Bashir, A., Imran, M., Kumar, N., Zhang, Di., & Yu, K. (2020). Robust spammer detection using collaborative neural network in internet of thing applications. *IEEE Internet of Things Journal, 8*(12), 9549–9558. https://doi.org/10.1109/JIOT.2020.3003802
27. Shi, W., & Xie, M. (2013). A reputation-based collaborative approach for spam filtering. *AASRI Procedia, 5*, 220–227. https://doi.org/10.1016/j.aasri.2013.10.082

28. Sousa, P., Machado, A., Rocha, M., Cortez, P., & Rio, M. (2010). A collaborative approach for spam detection. 2nd international conference on evolving internet, 92–97, https://doi.org/10.1109/INTERNET.2010.25
29. Hau, X., Pham, L., Nam-Hee, J. J., & Sadeghi-Niaraki, A. (2011). Collaborative spam filtering based on incremental ontology learning. *Telecommunication Systems - TELSYS*. https://doi.org/10.1007/s11235-011-9513-5
30. Damiani, E., Vimercati, S., Paraboschi, S., & Samarati, P. (2004). P2P-based collaborative spam detection and filtering. In *4th International Conference on Peer-to-Peer Computing*, 176–183. https://doi.org/10.1109/PTP.2004.1334945
31. Koggalahewa, D. N., Xu, Y., & Ernest, F. (2020). Spam detection in social networks based on peer acceptance. In *Proceedings of the Australasian Computer Science Week Multiconference (ACSW '20)*. Association for Computing Machinery, 1–7. https://doi.org/10.1145/3373017.3373025
32. Pera, M., & Ng, Y.-K. (2007). Using word similarity to eradicate junk emails. *International Conference on Information and Knowledge Management*. https://doi.org/10.1145/1321440.1321581
33. Moniza, P., & Asha, P. (2012). An assortment of spam detection system. In *International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*, 860–867, https://doi.org/10.1109/ICCEET.2012.6203823
34. Ho, P.-T., & Kim, S.-R. (2014). Fingerprint-based near-duplicate document detection with applications to SNS spam detection. *International Journal of Distributed Sensor Networks*. https://doi.org/10.1155/2014/612970
35. Jaiswal, S., Patel, S., Singh, & Ravi. (2016). Privacy preserving spam email filtering based on somewhat homomorphic using functional encryption. https://doi.org/10.1007/978-81-322-2695-6_49.
36. Gopi, S., & Ketan, K. (2019). Incremental personalized E-mail spam filter using novel TFDCR feature selection with dynamic feature update, Expert Systems with Applications.
37. Henke, M., Santos, E., Souto, E., & Santin, A. O. (2021). Spam detection based on feature evolution to deal with concept drift. *JUCS - Journal of Universal Computer Science, 27*(4), 364–386. https://doi.org/10.3897/jucs.66284.
38. Luo, GuangJun, Shah, N., Khan, H. U., & Haq, A. U. (2020). Spam detection approach for secure mobile message communication using machine learning algorithms. *Security and Communication Networks*. https://doi.org/10.1155/2020/8873639.
39. Ma, J., Zhang, Y., Liu, J., Yu, K., & Wang, X. (2016). Intelligent SMS spam filtering using topic model. *International Conference on Intelligent Networking and Collaborative Systems (INCoS)*. https://doi.org/10.1109/INCoS.2016.47
40. El Kouari, O., Benaboud, H., & Lazaar, S. (2020). Using machine learning to deal with Phishing and spam detection: An overview. In *Proceedings of the 3rd International Conference on Networking, Information Systems & Security (NISS2020)*. Association for Computing Machinery, 1–7. https://doi.org/10.1145/3386723.3387891
41. Yeganeh & Mehdi (2012). A Model for fuzzy logic based machine learning approach for spam filtering. *IOSR Journal of Computer Engineering*. https://doi.org/10.9790/0661-0450710.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

P. Rajendran is working as Associate Professor in the Department of Computer Applications, Velalar College of Engineering and Technology. He has completed M.C.A., M.Phil and currently doing Ph.D (Part time) in Anna University, Chennai. His research area includes Privacy Preserving Data Mining, Big Data Analytics, Data Mining and Data Warehousing. He has published many articles in national and international journals and presented papers in national and international conferences.

Dr. A. Tamilarasi has completed M.Sc., M.Phil., M.Techand Ph.D and is presently working as Professor in the Department of Computer Applications at Kongu Engineering College, Tamilnadu, India. Her area of specialization is Theoretical Computer Science. She has published 168 articles in International journals and has presented more than 70 papers in national and international conferences. She has authored 9 books. She has received funds from UGC and ICMR. She has guided more than 30 Post Graduate projects. She is a recognized supervisor in Anna University, Chennai.

Dr. R. Mynavathi has completed her PhD degree in Anna University, Chennai. She is presently working as Professor in the Department of Information Technology, Velalar College of Engineering and Technology, Tamilnadu, India. Her area of specialization includes Data Mining and Data Warehousing. She is also interested in Big Data Analytics and Data Science. She has presented more than 10 articles in International Journals and has presented more than 25 papers in National and International Conferences.