Paper review: A Graph-Based Approach for Multi-Folder Email Classification

Problem statement

The paper proposes a new approach to effectively classify emails in the face of a large number of daily emails. Existing email classification approaches have limitations, such as relying heavily on high-frequency keywords and ignoring the structural aspects of an email. Another is that some models consider only the words without taking into consideration where in the structure these words appear together. The proposed approach leverages graph mining techniques to capture the structural aspects of an email, and a supervised learning model generates representative substructures from pre-classified emails. The ranked substructures are used to categorize incoming emails, leading to improved accuracy in email classification and a more efficient way of managing emails. Experimental validation of the proposed approach shows promising results.

Motivation

1. R. B. Segal and J. O. Kephart, “Swift-file: An intelligent assistant for organizing e-mail,” *Proceedings of AAAI 2000 Spring Symposium on Adaptive User Interfaces*, pp. 107–112, 2000.
2. E. Crawford, J. Kay, and E. McCreath, “Automatic induction of rules for e-mail classification,” *Proceedings of the Sixth Australasian Document Computing Symposium, Coffs Har- bour, Australia*, 2001.

These two referenced articles show us that traditional email classification uses different criteria that are extremely difficult to quantify, such as:

1. Evolving mailboxes: Each mailbox is different and is constantly evolving. Folder contents vary from time to time, and emails within a folder may not be cohesive.
2. Subfolder classification: Emails are typically classified into subfolders within a folder, adding an additional layer of complexity to the classification process.
3. Structural characteristics: Email classification requires consideration of the structural characteristics provided by an email, which are often ignored by conventional techniques.
4. Adaptive training: The email environment is constantly changing, and there is a need for adaptive and incremental re-training to ensure accurate classification.

At the same time, it also proves the necessity of using graph mining technology for multi-folder email classification.

key ideas, techniques, and contributions

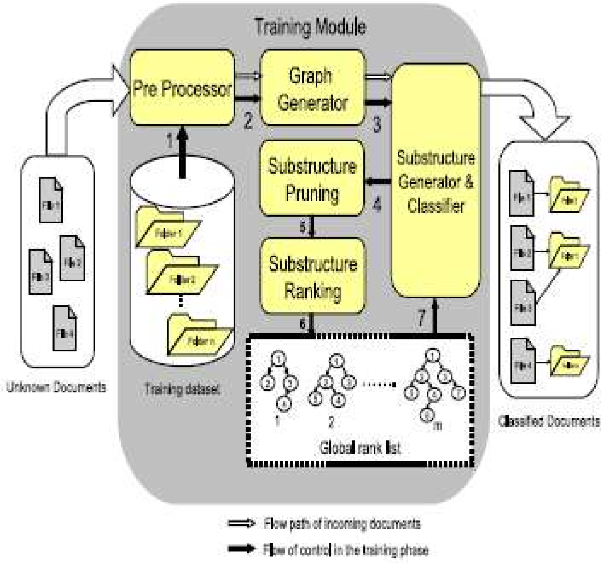


Figure 1. m-InfoSift System Overview

1. The m-InfoSift Framework
2. Substructure extraction, pruning and inferring parameters
3. Substructure ranking and classification

The m-InfoSift framework is a unique graph-based approach for email classification that uses a two-phase approach of training and classification. It employs Subdue for substructure extraction from a forest of unconstrained graphs and ranks substructures based on their ability to compress the input graph using the Minimum Description Length (MDL) principle. The top-n substructures are selected and used as templates for classifying unknown emails in the classification phase. Inexact graph matching is used to compare the incoming document with the template substructures, and the incoming document is assigned to the class with the highest ranked substructure match. The framework allows for the effective classification of emails with similar characteristics and structure, and it has several practical applications in spam filtering, prioritization of emails, and information retrieval.

Experimental evaluation

Background

The article discusses the comparison of the authors' approach to email classification with the Naive-Bayesian approach, as most email classification systems are either rule-based or user-guided and not available in open-source. The authors elaborate on the parameters used in their approach, including training-test set split, seed, substructure discovery threshold, classification threshold, and beam. The article provides details on various parameters used in the experiments conducted to evaluate the proposed email classification framework. These parameters include the training-test set split (80:20 and 60:40), seed value (default of 100), substructure discovery threshold (0.1), classification threshold (0.05), and beam value (2, 4, 8, and 12). The dataset used for the experiments consists of various folders selected from public Listservs and personal emails, and experiments were also carried out on the Enron Email Dataset. The experiments were carried out on machines with 2GB memory, and a large number of classes with diverse characteristics were used to study the effect of parameters on the classification of unknown test documents in a multiple folder environment.

Result

1. The study compares the performance of two email classification approaches, m-InfoSift and Naive Bayes, using Figure 5 to show that m-InfoSift outperforms Naive Bayes with an accuracy improvement of 10% to 70%. The Naive Bayes approach assigns probabilities to word occurrences, leading to wrongly classified emails when terms are common to multiple folders and have a higher weight assignment in one folder. The study's approach identifies patterns of word occurrences and ranks them across all the folders, avoiding this problem. Additionally, Naive Bayes performs poorly due to the large number of false positives, whereas m-InfoSift's global rank scheme has fewer false positives, as shown in Figure 5. The study also shows that m-InfoSift consistently outperforms Naive Bayes across the Enron dataset, as shown in Figure 6, although the difference in performance is less distinguishable than for the Listserv dataset.
2. To summarize, the experiments showed that the performance of the email classification using graph mining is affected by the size of the feature-set. The authors tested three different values for retaining the frequent set of words: 60%, 80%, and 100%, and found that using 80% as the feature selection ratio consistently resulted in better classification accuracy compared to retaining all words (100%) or a lower number of words (60%). Additionally, the classification accuracy was observed to decrease with the increase in the number of folders, which was expected as the chances of email getting correctly classified to the right folder decreases with the increase in the number of substructures in the global rank list. The results of these experiments are shown in Figures 5 and 8.

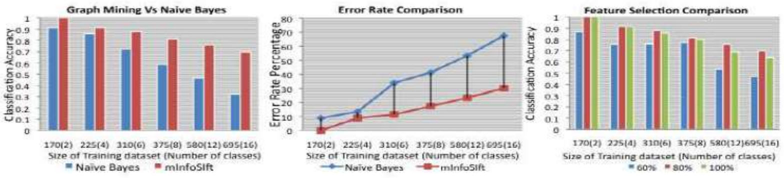


Figure 5. Naive Bayes vs. m-InfoSift - Listserv dataset

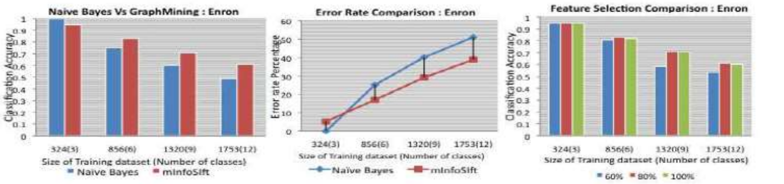


Figure 6. Naive Bayes vs. m-InfoSift - Enron dataset

3. Inexact graph matching allows for small variations in the substructures of the email graphs, which is important in situations where exact matches are difficult to find. This is especially true for emails, which do not always follow a set vocabulary and have sparse information content compared to text documents. The experiments conducted in Figures 7 and 8 demonstrate that inexact graph matching leads to better performance in classification accuracy compared to exact graph matching.

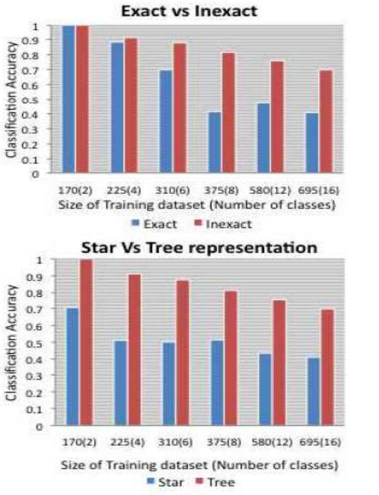


Figure 7. Substructure Discovery & Representations: Listserv dataset

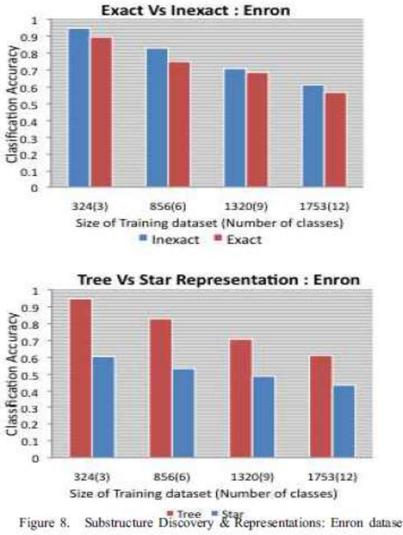
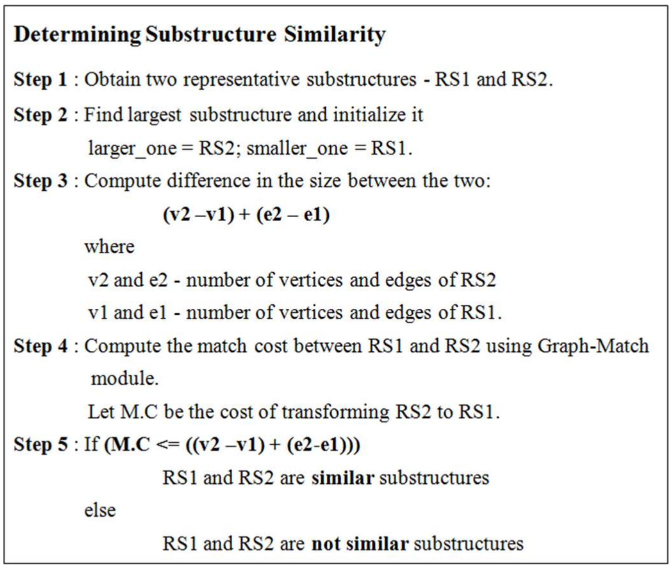


Figure 8. Substructure Discovery & Representations: Enron dataset

Conclusion

In conclusion, the paper presents a framework for email classification using graph mining techniques. The proposed ranking formula and adaptive characteristics adjust to the size and characteristics of the folder, and feature subset selection allows for classification with small amounts of data. The alternative graph representations improve the representation of the documents and incorporate relevant domain information. The experimental results show significant performance improvement over the Naive Bayesian approach for different domains.

Presentation perspective



```
Determining Substructure Similarity
Step 1 : Obtain two representative substructures - RS1 and RS2.
Step 2 : Find largest substructure and initialize it
         larger_one = RS2; smaller_one = RS1.
Step 3 : Compute difference in the size between the two:
                 $(v2 - v1) + (e2 - e1)$
         where
         v2 and e2 - number of vertices and edges of RS2
         v1 and e1 - number of vertices and edges of RS1.
Step 4 : Compute the match cost between RS1 and RS2 using Graph-Match
         module.
         Let M.C be the cost of transforming RS2 to RS1.
Step 5 : If (M.C <= ((v2 - v1) + (e2-e1)))
                 RS1 and RS2 are similar substructures
         else
                 RS1 and RS2 are not similar substructures
```

The paper focuses on the organization and logic flow of the proposed approach for email classification. The authors stress the importance of preserving the structural characteristics of emails using a graph representation, which helps achieve higher accuracy in classification. The paper includes visual aids such as figures and tables to illustrate the approach and experimental results, which I have shown in the part of **Result**. While the paper does not directly address the rigidity of proofs, it provides a detailed description of the methodology and evaluation process, which can help readers assess the soundness of the approach and replicate or build upon it in future research.

Evaluation based on my own criteria

strong points:

1. The performance of the system is consistent, but the classification accuracy reduces with an increase in the training and test data size, which suggests that using the latest emails in the folder for training can be advantageous.
2. Emails have a structure that can be exploited for classification purposes.
3. The performance of the m-InfoSift algorithm for multiple folders is significantly better than Naive Bayes and is consistent across different categories of emails.

weak points:

1. A structure that maps the document better intuitively is likely to improve the accuracy of classification, as demonstrated by the tree versus star representations. It can be considered to use in it.
2. It is crucial to derive all parameters for graph mining and classification from the folder itself, which ensures adaptability of the approach to arbitrary folders. It can be shown in this paper.
3. I think that larger thresholds are required for documents with relatively small content and thus fewer vertices in the input graph representation.

Extended Discussion

If you are the authors, what will you plan to do to address the weaknesses or to improve/extend any part of the solution?

1. I would explore alternative graph representations that may better capture the structure of email documents. This could involve investigating more complex graph structures or experimenting with different feature selection techniques.
2. I would investigate approaches to automating the parameter selection process. This could involve using machine learning techniques to learn optimal parameter settings based on the characteristics of the email dataset.
3. I would investigate approaches to adaptively adjusting the threshold based on the size of the document. This could involve using machine learning techniques to learn the optimal threshold setting for different document sizes.

If you are to continue working on this paper, what is the next topic you would like to pursue to enrich this research?

I will investigate incremental generation of representative substructures and how they change over time, and whether that information can be used to develop heuristics for the manual classification process

What other new research topics, or applications, will this proposed solution impact?

The proposed solution of using graph mining techniques for email classification has the potential to impact a variety of other research topics and applications. One such example is in the field of social media analysis. With the explosive growth of social media platforms, there is a need for automated techniques to analyze and classify large volumes of data generated by users. The approach proposed in this paper can be extended to social media analysis by using graph mining techniques to extract representative substructures from social media posts and messages, and then classifying them into different categories such as sentiment analysis or topic modeling.