ALEXANDRIA UNIVERSITY
FACULTY OF ENGINEERING
COMPUTER AND SYSTEMS ENGINEERING DEPARTMENT

Taxonomy for Email Classification and Summarization Techniques

Ahmed El-Sharkasy, Ahmed Kotb, Amr Nabil, Mohammad Kotb, Moustafa Mahmoud

Supervisors: Prof. Dr. Mohamed Abou-gabal, Dr. Mustafa Elnainany

Contents

1 Abstract

In this document we present a survey and taxonomy on recent research topics related to email classification and summarization. This document summarizes and organizes recent research results in the novel way that integrates and adds understanding to work in the field of email classification and summarization. It emphasizes the classification of the existing literature, developing a perspective on the area, and evaluating different trends.

Keywords Email, Classification, Summarization, Machine Learning.

2 Introduction

Email has been an efficient and popular communication mechanism as the number of Internet users increases. Therefore, email management has become an important and growing problem for individuals and organizations because it is prone to misuse. One of the problems that are most paramount is disordered email message, congested and unstructured emails in mail boxes. It may be very hard to find archived email message, search for previous emails with specified contents or features when the mails are not well structured and organized.

Many machine learning approaches have been applied in this field, the most State-of-the-Art algorithms in email classification include: support vector machines, neural network, nave bayes classifiers and entropy-based approach.

Email summarization is another important and challenging problem. We can think of automatic summarization as a type of information compression. To achieve such compression, better modelling and understanding of document structures and internal relations is required.

In this document we present a survey and taxonomy on recent research topics related to email classification and summarization.

3 Email Classification Taxonomy

The following table classifies some recent research papers in the field of email classification according to the different learning algorithms used in different papers

Learning Algorithm					
SVM	Nave Bayes	Neural Networks	Max. Entropy / Winnow	Nnge / Hoeffing Trees	Graph Mining
An Innovative Analyser for email classification Based on Grey List Analysis	Email Classification with Co-training	Email Classification: Solution with Back Propagation Technique	Automatic Categorization of Emails into Folders	Using GNUsmail to compare Data Stream Mining Methods for On-line Email Classification	A graph Based Approach for Multi-Folder Email Classification
Email Classification with Co-training	Automatic Categorization of Emails into Folders	Email Classification Using Semantic Feature Space			
Automatic Categorization of Emails into Folders					

4 Email Summarization Taxonomy

5 Papers Summary

5.1 Email Classification

5.1.1 Automatic Categorization of Email into Folders [1]

Year 2004

Citations 112

Introduction

- Users get alot of emails this days, not just spam but a large number of legitmate emails also that they need to process in a short time.
- The paper shows the results of an extensive benchmark on two large corpora (enron,sri) of 4 classification algorithms.
- The paper shows an enhancement to the exponential gradient method (winnow).

Related Work

- Clark and Niblet 1989: proposed a rule inductive algorithm CN2 and showed that it can outperform KNN.
- Cohen 1996: proposed the RIPPER classifier and showed that that it can outperfrom an tfidf classifier.
- Provost 1999: showed that Naive bayes can outperform RIPPER.
- Remmie 2000: achived a very high accuracy by classifying mails to 3 predefined folders .
- Kiritchenko and Malwin 2001: showed that SVM can outperfom Naive Bayes.

Algorithms Benchmarked

- Maximum Entropy.
- Naive Bayes.
- SVM.
- Winnow (enhanced version).

Challenges in mail classification

- Email users often create folders and let it fall out of use (small number of training data per folder).
- Folders dont necessarily correspond to simple semantic topics (unfinished todos, project groups, certain recipient).
- Differ drastically from one user to another.
- Email arrives in a stream over time which causes more difficulties, for example the topic of main folder can drift over time.

Data set pre-processing

- Removing non topical folders (Inbox, sent, trash, ...etc).
- Removing small folders (folders that has a small number of emails).

Training/test set splits

- The paper shows a new way to split training data into training set and test sit, the new method takes time factor into considerations.
- It works as follows:
  - sorting emails by time;
  - train the classifier for the first N emails;
  - test it on the following N emails;
  - train the classifier for the first 2N emails;
  - then test it for the following N emails;
  - and so on.

Features Extraction traditional bag of words representation.

Datasets

- Enron: http://www.cs.cmu.edu/ enron/
  - 150 users with more than 500,000 Emails;
  - applied to the following 7 employees folder only (the largest 7 folders : beck-s,farmer-d, kaminski-v, kitchen-l, lokay-m, sanders-r and williams-w3);
  - removed the non topical folders like "all documents", "calendar", "contacts", "deleted items", "discussion threads", "inbox", "notes inbox", "sent", "sent items" and "sent mail";
  - flatten all the folder hierarchies;
  - removed folders with less than 3 messages;
  - removed the X-Folder field from email messages. (The X-Folder field contains the class label).
- SRI : http://www.ai.sri.com/project/CALO
  - applied to the following 7 folders only: acheyer, bmark, disrael, mgervasio, mgondek, rperrault, and vchaudri;
  - removed the non topical folders (inbox, draft, sent, trash);

- flatten all the folder hierarchies;
- removed folders with less than 3 messages.

Critique

- They didnt use Stemming in their preprocessing to the dataset.
- Not including precision , recall and f1 score for accuracy measures.

Conclusion

- Naive Bayes is inferior to other algorithms.
- SVM achieved the highest accuracies in most of the tests.

Future Work

- Different sections of each email can be treated differently. For example, the system could create distinct features for words appearing in the header, body, signature, attachments, ...etc.
- Named entities may be highly relevant features. It would be desirable to incorporate a named entity extractor (such as MinorThird3, see, e.g., Cohen and Sarawagi (2004)) into the foldering system.

5.1.2 Email Classifications For Contact Centers [2]

Year 2003

Citations 14

Main Topic

- Proposing an automatic system to classify mail message for contact centers.
- Mails are categorized into 2 classes:
  - single messages: messages that dont require a response;
  - root messages: messages that require immediate response;
  - root messages can be sub divided into 3 classes:
    - root: the start of the communication (contains a problem or a question);

  * inner: communication on a certain problem;
  * leaf: marks the end of this interaction (eg. the problem was solved).

Tools used

- Rainbow: an implementation for naive bayess algorithm.
- SVMlight: an implementation for SVM algorithm.
- WordNet: used for parts of speach taging.
- Ltchunk: used to identify noun phrases and count number of sentences in email.

Dataset

- Pine-info discussion list web archive
  - http://www.washington.edu/pine/pine-info.

Pre-processing

- Removing reply blocks (blocks from previous emails in the current mail).
- Removing signature blocks.

Features (for SVM algorithm)

- Non-infected words
  - nouns, verbs, adjective, adverb;
  - using WordNet;
- Noun phrases
  - using Ltchunk;
- Verb phrases.
- Punctuation letters count.
- Length of email (number of sentences)
  - using Ltchunk;
- Dictionary
  - 2 dictionaries were made one for the most common words in single messages and the other for the most common words in root message.

Conclusion

- High accuracy was achieved on root vs leaf (92%) , root vs inner (87%) and root vs single(79%).

5.1.3 Using GNUsmail to Compare Data Stream Mining Methods for On-line Email [3]

Year 2011

Citations 0

Main Topic

- Introducing GNUsmail, an open source framework used for mail classification, focusing on online incremental learning.
- Proposing new techniques for testing other than holdout and cross-validation like prequential measure.

Evaluation methods

- Prequential measure.
- Sliding and fading windows.
- McNemar test.

Dataset

- Enron.
- A layer was added to feed the learning algorithm the new emails one by one, simulating new incoming emails.

Algorithms

- OzaBag over NNge, using DDM for concept drift detection.
- NNge.
- Hoeffding Trees.
- Majority class.

Tools

- GNUsmail: http://code.google.com/p/gnusmail/.

Result

- Improved GNUsmail by incorporating new different methods to evaluate data stream mining algorithms in the domain of email classification.

Future Work

- Current online learning algorithm implementations have an important limitation that affects the learning process: learning attributes have to be fixed before beginning the induction of the algorithm. They need to know all the attributes, values and classes before the learning itself, since it is not possible to start using a new attribute in the middle of the lifetime of a learning model. Future methods should support online addition of new features.

5.1.4 E-Classifier: A Bi-Lingual Email Classification System [4]

Year 2008

Citations 0

Problem

- Classifying Arabic and English emails.
- Implementing an outlook add-in "e-classifier".

Related Work

- English Email Classifiers
  - PopFile
    * http://popfile.sourceforge.net.
    * Uses naive bayes algorithm only.
  - SpamBayes
    * http://spambayes.sourceforge.net

    * Binary Classifier (Spam or not).

- Arabic Email Classifiers
  - There are no Email classification work on arabic language, the related work are on arabic documents not emails El-Kourdiet.

Dataset

- English: enron dataset.
- Arabic
  - Translated documents that have been converted to emails.
  - Documents obtained from http://www.comp.leeds.ac.uk/eric/latifa/research.html.

pre-processing

- English
  - Removing stop words.
  - Removing punctuation marks.
  - Converting all the letters to lowercase.
  - Porter stemmer.
- Arabic
  - No root extraction technique was used due to the lack of non commercial product.

Results

- 85% of English emails were classified correctly.
- 60% of Arabic emails were classified correctly.

Critique

- Used only overall accuracy measure which might not good indicator in case of skewed data.

5.1.5 An Object Oriented Email Clustering Model Using Weighted Similarities between Emails Attributes [5]

Year 2010

Citations 6

Description

- Proposing a new Object Oriented Email Clustering Model to categorize mail message into groups.

Algorithms

- K-means clustering algorithms.
- Text similarity techniques.
  - cosine Similarity.
  - Dice Similarity.
  - Blue Similarity.
  - TF-IDF Similairty (Term Frequency - Inverse Domain Frequency).
  - Jaccard Similairty.

Datasets

- Enron dataset.
- Inbox folder of base-e user mail box.

Dataset pre-processing

- Stemming.
- Parsing
  - To extract email attribuites (subject, body, ...etc).
- Storing in an object oriented representation.

Tools/programming languages used

- Java.
- Simmetric: used to calculate text similarities.
- Weka (Waikato Environment for Knowledge Analysis): used for stemming of emails.

Future work

- Thread summarization.
- Automatic email answering.

Conclusion

- Email can be represented as an object with attribute like subject, body, ...etc.
- Clustering of emails can be implemented in an object oriented way.

5.1.6 Content Based Email Classification System by applying Conceptual Maps [6]

Year 2009

Citations 0

Main Topic

- Proposing a Knowledge based System (KBS) to classify messages into folders.
- Using lexicon and conceptual graphs.

Major steps of processing on subject and body fields

- Word splitting.
- Word normalization (stemming).
- Detect abbreviation.
- Removing stop words.
- Word indexing.
- Identify noun-phrases by NLP techniques.
- Conversion of phrases into concepts.

Related Work

- C-Evolove.
- Titus.

5.1.7 A new approach to Email classification using Concept Vector Space Model [7]

Year 2008

Citations 3

Algorithms

- Used a classification algorithms based on pre-processing steps in the training phase to produce vector that identify the category or the new email.
- Based on WordNet, for describing a text Email by establishing concept vector space model, we can firstly extract the high-level information on categories during training process by replacing terms with synonymy sets in WordNet and considering hypernymy-hyponymy relation between synonymy sets.
- Used TF * IWF * IWF method to revise the weight of the concept vector.

Dataset

- Used documents of 20 news group (standard document set).
- Put these documents in 20 directory as 20 category, each category contains at least 1,000 article.
- Set of these articles are selected to be used as training set, another set as a test set.

Results

- Made two experiments on different conditions, comparing the concept VSM method with a traditional VSM method:
  - experiment 1:
    - selected 3 categories from the dataset, and chosen 300 email at random from each category as training set and 100 email from each category as test set;

    * observed that the F1-meausre of the concept VSM is always better than traditional VSM by at least factor of 0.1 (for more details check Tables 1,2 in the paper) with F1-measure for concept VSM in the 3 datasets 0.84, 0.90, 0.93 respectively.
  - experiment 2:
    * used the same categories in experiment one, but repeated experiment one but with different training set size, starting from 30 email;
    * observed that Concept VSM is always better than traditional VSM;
    * accuracy starts from 0.4 at 30 email training set for all categories and increases till it reach 0.9 for training set size as in experiment 1 (900 emails for all categories);
    * this means that Concept VSM is working fine with small training set size but it is better if it is increased;
    * for more details check Figure 2 in the paper.

Future work Use concept VSM to do level classification.

5.1.8 Ontology based classification and categorization of email [8]

Year 2008

Citations 4

Problem

- Making a user defined and user controllable spam filter to detect spam emails, the paper uses ontology for understanding the content of the email and Bayesian approach for making the classification.
- Categorizing mails based on their content.
- The complete process: classifying mails as hams or spams and further classification of ham emails to folders.

Algorithms

- Content based filtering: uses keywords in the mail for classification.

- Statistical based filtering: Assigns probability or score to each keyword and uses the overall probability or score to classify the new mail.
- Machine learning approach for filtering: Ontology is used as one of the learning tools for email classification.

Results

- 98% of the emails has been classified successfully to ham and spam.
- 95% of the ham has been successfully categorized into folders.

Conclusion

- User defined spam filter has better results than general spam filters for all user.

5.1.9 Enterprise Email Classification Based on Social Network Features [9]

Year 2011

Citations 0

Problem Managing the email services in Enterprises, so that business emails have priority over personal emails by classifying emails into official and private. The classification is made based on social features not on the email content for protecting the privacy of users' emails by building a social network analysis graph representing the senders and recipients as vertixes and the sending events as edges.

Algorithms

- Support vector machine (SVM).
- WEKA.

Results

- F-measure = 0.9

Related Work

- SNARF http://research.microsoft.com/en-us/projects/snarf/

Future work

- Combining some state-ofthe-art email prioritization algorithms with the proposed method to balance the loading of email server.

5.1.10 Email Categorization Using Multi-Stage Classification Technique [10]

Year 2007

Citations 5

Problem

- Email classification (spam) using a multi-stage classification technique collecting all the mails which is not TP or TN in a different mailbox for the user to give feedback about them. The classification of emails is done in multi stages where in each stage a new classifier is added to filter output.

Algorithms

- SVM.
- Naive Bayes.
- Boosting Algorithms.

Results

- Average FP is 0 and average FN is lower than that results from using any algorithm individually.
- Accuracy : 97.05%.

Data sets

- PUA.

Future work

- Analyse cost in complexity and speed.

5.1.11 Automatically tagging email by leveraging other users folders [11]

Year 2011

Citations 0

Problem

- Automatically associating semantic tags to emails other than creating folders. Beside providing a way to tag emails automatically, they started with predefined set of tags taken from a study on the folders and labels yahoo users are generating. The proposed technique took into consideration the performance and scalability. The technique learned how to tag by taking into account the habit of many users for making folders simultaneously.

Algorithms

- K-means.
- Naive bayes.
- Other proposed algorithms.

Data sets

- Emails from yahoo mail users (200 million emails).

Conclusion

- The paper presented a classification system for tagging emails suitable for a very large scale system up to millions of emails and reached a performance of 2 ms or less for classifying an email and with acceptable accuracy.

Future work

- Increasing the features extracted from the emails to include To: and Cc: fields, the length of a message, the number and names of file attachments,style (html/plain) signals, and more sophisticated subject tokenization techniques.

5.1.12 An Email Classification Model Based on Rough Set Theory [12]

Year 2005

Citations 19

Problem

- Reducing the error rate of classifying non-spam emails into spam by classifying the incoming emails into 3 categories instead of 2: spam, non-spam and suspicious using an algorithm based on rough set theory.

Related work

- Ripper Algorithm.
- Genetic Document Classifier.
- Smokey.
- Bayesian Junk Email Filter.
- Max. Entropy Model.

Data sets

- http://www.ics.uci.edu/mlearn/MLRepository.html

Results

- Accuracy reached 97%.

Conclusion

- Rough set based model can reduce the error rate that classifies a non-spam email to spam.

5.1.13 eMailSift: Email Classification Based on Structure and Content [13]

Year 2005

Citations 16

Problem

- Extracting structures/patterns from pre-classified emails and using them for classification.

Challenges

- Manual classification of emails is based on personal preferences.
- Each users mailbox is different and is constantly evolving (temporal factor).
- The information content of emails vary significantly and not as rich as text documents.
- The characteristics of folders may vary from dense to relatively sparse. A classification system needs to perform reasonably well in both and degrades gracefully.
- Emails are typically classified into sub-folders within a folder.

Related Work

- Rule Based Classification: use rules to classify emails into folders
  - William Cohen: RIPPER learning algorithm
  - i-ems: Rule based classification system that learns rules based only on sender information and keywords
  - Ishmail: Rule-based classifier integrated with the Emacs mail program Rmail.
- Information Retrieval Based Classification:
  - Segal and Kephart: TF-IDF classifier for classification in SwiftFile
- Machine Learning Based Classification:
  - The iFile system by Rennie uses the naive Bayes approach

- Re:Agent by Boone first uses the TF-IDF measure.
- Mail Agent Interface (Magi) by Payne and Edwards uses the symbolic rule induction system CN2.

Relevant Work in Graph Mining

- Subdue Substructure Discovery System by Cook and Holder: The Subdue graph based mining algorithm accepts as input a forest of graphs and identifies the best subgraph that minimizes the input forest using the minimum description length (MDL) principle.

Algorithm Phases

1. Preprocessing:
   - Elimination of stop words.
   - Words are ranked based on their occurrence frequency across all emails in a folder and those whose frequencies account for more than f% of the sum of all frequencies are retained.
2. Graph Representation:
   - Choose a graph representation that is appropriate for the email domain and use it for representing the emails in a folder.
3. Substructure Extraction:
   - Graph mining techniques are used for extracting representative substructures.
4. Representative Substructure Pruning:
   - The output of the discovery process may contain a large number of substructures. The goal of pruning is to identify the subset needed for discriminating incoming emails during classification.
5. Representative Substructure Ranking:
   - Each representative substructure is ranked to indicate its representativeness and the associated rank is used in classifying incoming emails.

6. Classification:

    - The incoming email is compared with the representative substructures of a folder to determine if it matches any of the representative substructures. For multiple folder classification, in case of more than one match, it is classified into the folder with the highest ranked substructure match.

Results

- The performance of eMailSift is much better than naive Bayes and it is consistent in successfully classifying incoming emails.

Notes

- The eMailSift classifier works well on folders of all sizes. With an increase in folder size, leading to an increase in the heterogeneity of a folder, the classification accuracy remains good.
- New trend: To the best of the authors knowledge (in 2005), this is the first attempt to assess the applicability of graph mining for classification.

5.1.14 A Graph-Based Approach for Multi-Folder Email Classification [14]

Year 2010

Citations 1

Abstract

- This paper presents a supervised learning model that leverages graph mining techniques for multi-folder email classification. A ranking formula is presented for ordering the representative substructures generated from pre-classified emails. These ranked representative substructures are then used for categorizing incoming emails.

Problem

- Other existing techniques (e.g., SVM, TFIDF, n-gram) rely heavily on extracting high-frequency keywords, thus ignoring the inherent structural aspects of an email which can play a critical role in classification. Moreover, they fail to take into account the differences between an email and a normal text document and hence not utilize the characteristics of email for classification. They also fail to take advantage of the structural characteristics provided by an email.

Solution

- Data representation in the form of a graph preserves the structural information of the data which may otherwise be lost if it is translated into other representation schemes.

Challenges

- Classification of emails is based on personal preferences
- Each mailbox is different and is constantly evolving; folder contents vary from time to time.
- The information content of emails vary significantly, and other factors, such as the sender, group the email is addressed to, play an important role in classification
- The characteristics of folders may vary from dense (more number of emails) to relatively sparse
- Emails within a folder may not be cohesive i.e., the contents may be disparate and not have many common words or a theme
- Emails are typically classified into subfolders within a folder.

Related Work

- Binary classification of documents based on graph mining
- Usage of TF-IDF (Term Frequency - Inverse Documnent Frequency) for email classification
- Rule-based classification techniques
- Employing temporal features (e.g., day of the week, time of the day, etc.) in order to classify email messages into classes.

Pre-processing

- Stop-word Elimination
- Stemming
- Feature Selection: a technique commonly used in machine learning for selecting a subset of relevant features in order to build the learning model.

Results

- Graph Mining vs. Naive Bayes: performance comparison between the m-InfoSift approach and the Probabilistic Bayesian approach clearly show a significant improvement as compared to Bayesian. Accuracy improvement is 10

Future Work

- Investigating incremental generation of representative substructures as the folders change over a period of time.
- Investigating how representative substructures change over a period of time and whether that information can be used to develop heuristics/rules to describe the manual classification process.

5.1.15 Applying Machine learning Algorithms for Email Management [15]

Year 2008

Abstract This paper presents the design and implementation of a new system to:

- Predict whether an email received require a reply.
- Group emails
- Summarize email messages.

The system uses not only subjects and headers fields but also content of email messages to classify emails based on users activities and generate summaries of each incoming message with unsupervised learning approach.

Introduction

- In this paper, machine learning based techniques were developed to reduce email overload, solve email reply prediction, email groupings and email summarization.

Email Reply Prediction

- One novelty is: if the BCC or CC contains email addresses, it implies that emails copied to others. Such mails may require a reply.
- The second novelty is to check email message content as well as the subject field for special words (e.g MUST, MEET, URGENT etc.) this indicates that the email may require a reply
- The third novelty is to check if email message contains multiples of question marks (?) or single question mark, and if there is any, such a mail indicates a request and such a mail will require a personal attention.

Email Grouping

- Email grouping based on the users activities or based on the intent of the sender. Our approach analyzed the word taxonomy of email content. Taxonomy allows classification of content into categories and subcategories
- The suggest grouping works similar to vector space model method but with a new Idea
- The suggested grouping procedure can be divided into three stages:
    1. The email indexing where content bearing terms are extracted from the email content.
    2. The weighting of the indexed terms to enhance retrieval of email relevant to the user.
    3. Ranking the email with respect to the query according to a similarity measure.

Email Summarization

- This algorithm extracts important words in email messages so that the summarizer can generate a more useful summary from the message. The algorithm works logically based on the techniques as shown below:

- Input: N, M, Msg Output: Sentence list
  1. Identify N most frequent words in incoming email messages.
  2. Select M sentences from email containing most frequent words.
  3. Order the selected sentences according to their occurrence in the message.
  4. Output the ordered sentences as summary.

5.1.16 Co-training with a Single Natural Feature Set Applied to Email Classification [16]

Year 2004

Citations 23

Abstract

- Using co-training technique to help build more accurate classifiers.
- Co-training allows classifiers to learn with fewer labelled documents by taking advantage of the more abundant unclassified documents.
- Conventional co-training requires the dataset to be described by two disjoint and natural feature sets that are sufficiently redundant, which is not practical.
- This paper shows that when only a single natural feature set is used, the performance of co-training is beneficial in the application of email classification.

Problem

- Effective classifiers can be build but using a sufficiently large set of training examples. However, obtaining labelled Web pages or emails is very costly, because it usually requires a great deal of human effort to classify unlabelled documents.
- A new technique to overcome this problem, called cotraining. But one of the main requirements that were stated for co-training to be successful was that the dataset must be described by two disjoint sets of natural features that were redundantly sufficient.

The Co-training Algorithm

1. Input a document with 2 disjoint feature sets.

2. Cotraining employs two classifiers in a loop to label all the unlabelled examples. Each classifier takes turns to select the most confidently predicted examples and add these into the training set

3. Both classifiers then re-learn on the enlarged training.

4. The loop is then repeated for a number of iterations to maximize performance on a separate validation set.

Dataset

- Email classification tests were applied on the LingSpam1 corpus. This dataset consists of 2883 emails of which 479 are spam and 2404 are genuine emails.

Preprocessing and Classifiers Used

- Each email is broken up into two sections: the text found in the subject header of the email and the words found in the main body of the message.
- After applying a stop list, a word count of each word type was kept with a distinction made between the words that appeared in the subject header and those that appeared in the body.
- Upon inspection of the word lists, it was decided that the top 100 words was a suitable cut-off
- Each of the email documents was then represented using the term frequencies of the selected 100 features.

Experiments Investigated

1. Investigating the redundancy of the feature sets

2. Co-training with a random split of all features

Notes

- It was found that the performance of cotraining is sensitive to the learning algorithm used. In particular, co-training with nave Bayes (NB) worsens performance, while Support Vector Machines (SVM) improves it.
- This paper investigates the performance of co-training with only one natural feature set in comparison to the use of two natural feature sets. The main question that is addressed is: how useful is co-training with a single natural feature set?
- Three types of classifiers were tested: Decision Tree (DT), NB and SVM.
- Implementations of these classifiers were obtained from WEKA.

5.1.17 Email Classification: Solution with Back Propagation Technique [17]

Year 2009

Abstract

- Using neural network for email content classification with back propagation
- This paper proposes a new email classification model using a teaching process of multi-layer neural network to implement back propagation algorithm. Contributions are: the use of empirical analysis to select an optimum, novel collection of features of a users email message and a demonstration of the effectiveness of two equal sets of emails (training and testing data).

Related Work

- Yukun et al proposed a new email classification model using a linear neural network trained by Perception Learning algorithm (PLA) and a nonlinear neural network trained by Back Propagation Neural Network (BPNN). A Semantic Feature Space (SFS) method was also introduced in this classification model.

Solution Heuristics

- If the email is about: loss of life, vital incident, accident, etc, then our classifier should, then it should be categroized as 'critical'
- If the email is about: meeting deadlines, reminder of vital appointments, interview appointment, visa embassy appointment. In summary, if such a mail is about time and deadline, then it should be categorized as 'urgent'
- If the email is about: conference invites, paper presentations, reminder of events, meeting reminder, tasks to perform daily ... etc, then it should be categorized as 'very important'
- If there is not timing and deadline in such a mail, if it is not about loss of life, illness, reminder of meeting, messages from friend and family, and the likes, then it should be categorized as 'others'

Algorithm

- Neural network (NN) with back propagation techniques.
- They implemented search for collections of important words in email corpus from Enron, the refined problem then becomes the task of searching this corpus for email datasets that the query retrieval system considers relevant to what the mail user entered as the query.

Notes

- Sample categories for this paper are: Critical, Urgent, Very important, and Others
- Back propagation is a popular type of network that can be trained to recognize different patterns including images, signals, and text.
- The input of the NN is the word importance in email messages and the output is the importance.

5.2 Email Summarization

5.2.1 Detection of question-answer pairs in email conversations [18]

Year 2004

Citations 41

Problem

- The sentence extraction summarization method cant be applied in all types of documents.
- Using it in summarizing email threads is not efficient, as it is a very special type of documents, as sentences and words are written relative to previous emails, so using sentence extraction will not be useful in this case.
- This paper is trying to solve this problem by extracting pairs of questions and answers to summarize email threads.

Conclusion

- Good approach to extract question-answer pairs in the email conversation in case of interrogative questions.
- Declarative and rhetorical questions cant be detected such as "Please let me know ...", "I was wondering if ...", "If you could ..., that would be great".
- Future work is to investigate these types of questions.

5.2.2 Using Question-Answer Pairs in Extractive Summarization of Email Conversations [19]

Year 2007

Citations 12

Problem

- After 3 years from the previous paper, they thought for a new approach to make a hybrid solution by extractive summarization of email threads with automatically detected QA pairs.
- This approach is better than extracting QA pairs only, as due to some statistics they made on their dataset that:
  - 20% of emails are question-answer exchange;

  - 40% of all email threads involve question-answer exchange of some form.
- Sentence extraction may be very useful if augmented with email specific features as dialogic structure.

Algorithms

- Extractive summarization:
  - represent each sentence in the SEQA threadset with a feature vector along with its binary classification, which represents wether or not a sentence should be in a summary;
  - features used are length, position in the document, TF-IDF scores, ...etc.
- QA Pair detection:
  - train a classifer on QA detection on the data corpus.
- Integrating QA Pairs with Extractive summarization:
  - 3 different approaches:
    * SE+A: a sentence figures as an answer to a question asked earlier in the thread as an additional feature in our machine learning-based extractive summarization approach;
    * SE+QA: to add automatically detected answers to questions in extractive summaries and add detected questions to answers in extractive summaries not in the summaries;
    * QA+SE: start with automatically detected question-answer pair sentences which are then augmented with extractive sentences that do not appear already in the question-answer pair sentences.

Data sets

- Corpus contains 300 email thread, each thread contains on average 3.25 email message.
- Data set was prepared manually concerning these points:
  - write summaries of email threads of the corpus;
  - highlight and link QA pairs in the email thread

  * Highlight only the questions that seek information (wether it is interogative or declarative questions, with or without question mark, but not rhetorical questions).
  * Link question with its answer if it was found in the same thread.
- SEQA threadset is set of email threads containing QA pairs identified manually of size 44 email thread.

5.2.3 Summarizing email conversations with clue words [20]

Year 2007

Citations 48

Problem Proposing a new framework to summarize emails by capturing email conversations and giving weights to sentences. Their algorithm allows the user to specify the size of the summary

Related Work

- Multi-Document summarization method.
- Ripper classifier.
- And more.

Algorithms

- Clue word summarizer (CWS).
- Porters stemming algorithm.
- MEAD Summarizer.

Dataset Enron Dataset.

Conclusion This papers introduces the fragment quotation graph which represents the conversation structure of the emails.This graph includes hidden emails and can represent the conversation in more details than a simple threading structure. Based on the fragment quotation graph, a new summarization approach CWS has been developed to select important sentences from an email conversation

Future Work Improving the fragment quotation graph generation with more sophisticated linguistic analysis and also evaluating the algorithm with different data sets

6 Results

In this section we describe the results.

7 Conclusions

We worked hard, and achieved very little.

References

[1] Ron Bekkerman, Andrew McCallum, Gary Huang, *Automatic Categorization of Email into Folders: Benchmark Experiments on Enron and SRI Corpora*, 2004.

[2] Ani Nenkova, Amit Bagga, *Email Classification for Contact Centers*, 2003.

[3] Jose M. Carmona-Cejudo, Manuel Baena-Garcia, Jose del Campo-Avila, Rafael Morales-Bueno, Joao Gama, Albert Bifet, *Using GNUsmail to Compare Data Stream Mining Methods for On-line Email Classification*, 2011.

[4] Nouf Al Fe'ar, Einas Al Turki, Asma Al Zaid, Mashael Al Duwais, Mona Al Sheddi, Nora Al khamees, Nouf Al Drees, *E-Classifier: A Bi-Lingual Email Classification System*, 2008.

[5] Naresh Kumar Nagwani, Ashok Bhansali, *An Object Oriented Email Clustering Model Using Weighted Similarities between Emails Attributes*, 2010.

[6] S. Baskaran, *Content Based Email Classification System by applying Conceptual Maps*, 2009.

[7] Chao Zeng, Zhao Lu, Junzhong Gu, *A new approach to Email classification using Concept Vector Space Model*, 2009.

[8] M.Balakumar, V.Vaidehi, *Ontology based classification and categorization of email*, 2008.

[9] Min-Feng Wang, Sie-Long Jheng, Meng-Feng Tsai, Cheng-Hsien Tang, *Enterprise Email Classification Based on Social Network Features*, 2011.

[10] Md Rafiqul Islam, Wanlei Zhou, *Email Categorization Using Multi-Stage Classification Technique*, 2007.

[11] Yehuda Koren, Edo Liberty, Yoelle Maarek, Roman Sandler, *Automatically Tagging Email by Leveraging Other Users' Folders*, 2011.

[12] Wenqing Zhao, Zili Zhang, *An Email Classification Model Based on Rough Set Theory*, 2005.

[13] Lokesh Shrestha, Kathleen McKeown, *eMailSift: Email Classification Based on Structure and Content*, 2004.

[14] Sharma Chakravarthy, Aravind Venkatachalam, Aditya Telang, *A Graph-Based Approach for Multi-Folder Email Classification*, 2010.

[15] Taiwo Ayodele, Shikun Zhou, *Applying Machine learning Algorithms for Email Management*, 2008.

[16] Jason Chan, Irena Koprinska, Josiah Poon, *Co-training with a Single Natural Feature Set Applied to Email Classification*, 2004.

[17] Jason Chan, Irena Koprinska, Josiah Poon, *Email Classification: Solution with Back Propagation Technique*, 2009.

[18] Lokesh Shrestha, Kathleen McKeown, *Detection of question-answer pairs in email conversations*, 2004.

[19] Kathleen McKeown, Lokesh Shrestha, Owen Rambow, *Using Question-Answer Pairs in Extractive Summarization of Email Conversations*, 2007.

[20] Giuseppe Carenini, Raymond T. Ng, Xiaodong Zhou, *Summarizing Email Conversations with Clue Words*, 2007.