UNIVERITY OF TARTU

FACULTY OF MATHEMATHICS AND COMPUTER SCIENCE

Institute of Computer Science

Information Technology specialty

Martin Mäe

Classification of Email Messages

Bachelor's Thesis (6 EAP)

Supervisor: Tõnu Tamme

Author: …………………………………...   "…..“ June 2011

Supervisor: ……………………………….   "…..“ June 2011

Allowed to defence

Professor: ……………………...............   "…..“ June 2011

TARTU 2011

Contents

Introduction

Today email is one of the most widely used communication methods. It has been used for decades by now and is used daily by organizations as well as by individuals to forward and receive all kind of information. Considering this the amount of email messages sent and received has grown significantly and more than before we are seriously facing a message overload problem.

To make managing and finding messages easier it is reasonable to classify messages based on user needs. The specific way for classifying emails can be developed by every person just the way it is reasonable for the specific user.

An electronic message or in short email consists of two parts: the message body (email content) and the message header. By using information from there I will try to classify email messages to make it easier to find and manage both incoming and existing emails.

This thesis aims to give an overview of what classification is and introduce some common classification methods. Another aim is to briefly introduce email format and message overload problem and to take a look at the number of emails sent yearly. Last aim is to study different built-in features for widely used email programs to see if these features are useful for classifying emails to make finding information faster and easier.

This thesis is divided into 3 chapters. The first chapter gives an overview of email message format, the message overload problem, widely used email clients, and the amount of emails sent. In chapter two some classification methods, information extraction, categorization and classification are introduced. In chapter three some real life experiments are conducted to show how to use email clients to classify email messages.

1 Email

This chapter defines the message overload problem and email message format structure. In addition, it gives an overview of widely used email clients and the amount of emails sent in the 2009 and 2010.

1.1 Email message format

The email message format is defined in RFC (**R**equest **f**or **C**omments) 5322 (released in October 2008) and also some additional RFCs from 2045 to 2049. Collectively, these RFCs are called Multipurpose Internet Mail Extensions, or in short MIME.[16]

An email message consists of two major sections:

- **Header** contains information about the sender, receiver, subject, date, etc[16]
- **Body** is the message itself as text and is the same as the body of a regular letter[16]

Now, let's take a look at the main fields of an email header:

- Date – time of sending out the email message[16]
- From – usually the author of the email[16]
- To – one or many recipients of email[16]
- Cc – recipients who are not directly related to message but may be interested in the information containing in email[16]
- Bcc – recipients on that filed will remain invisible to other addressees[16]
- Subject – a short summary of the contents of email[16]

All these fields contain valuable information to classify an email message. In addition we can also use information in the email body and if an attachment is added, this is also a good piece of information to use. The email body can be written in plain text or in HTML.[16]

1.2 Message overload problem

Every day, more and more emails are sent and received by users as we depend increasingly on email communication. Messages are not only received from friends or colleagues but also from all kind of social networks and advertising companies. In 1996

Steve Whittaker and Candace Sidner pointed out that email is also used for document delivery, sending reminders, scheduling appointments which shows that email is used for a variety of purposes exceeding its original design as a simple communication application. Organizing this flow is far beyond filtering spam into the Junk folder by different spam filters and the amount of email messages in our Inbox keeps increasing. At some point when we realize that it is not necessary to delete any emails as the capacity of any email account is enough to store hundreds of thousands emails we face a problem where it is almost impossible to find information needed as the number of emails in our inbox keeps growing. In this situation we have created a huge and very chaotic list of email mainly sorted by the date received.[6]

Received emails often contain information which is not needed at the time of getting the email. In a situation like this the message is skipped and there is a real danger that this message will get “lost” or overlooked in the increasing amount of emails.

Whittaker and Sidner point out three different user criteria:[6]

- *Non filers* – these users do not use folders and rely on full text search. Most of their information in the inbox is old and 95% of emails are in inbox without being filed into different folders. To manage the information in the inbox they periodically delete large amounts of messages or move them to some kind of folder.[6]
- *Frequent filers* – these users made significant attempts to organize their inbox. They scanned their inbox to delete or move emails on a daily basis. Their inbox contained only 5% of the total number of their emails.[6]
- *Spring filers* – these users dealt with their inbox once in a period of 1-3 months. They use folders but the rate of success is 50-50. More than 40% of their inbox was out of date information.[6]

Ten years later in 2006 a Microsoft Research team published a paper of changes compared to Whittaker’s and Sidner’s paper. They said that the size of inbox is basically the same as in 1996 as the archives of emails have increased tenfold.[1] This shows that the amount of information stored in inboxes is extremely large and finding useful information from there is almost impossible or it takes a long time.

1.3 Amount of email sent

Royal Pingdom has put internet into numbers yearly. If we take a look at the numbers of the last two years we can see that the total number of emails sent has increased from 90 trillion in 2009 to 107 trillion in 2010 that means the increase of 16%. We can also see that the number of email users has increased from 1.4 billion to 1.88 billion. Simple calculation shows that every email user sent 64286 emails in 2009 and 56915 emails in 2010 that makes average of 176 and 155 emails per day.[10, 11]

1.4 Most popular email clients

In 2010 Outlook was the most used email client with its market share of 43%. On the 2nd and 3rd places there are Hotmail and Yahoo! Mail with market shares of 17% and 13%. Gmail, Apple Mail, iPhone, Thunderbird, Windows Live Mail (desktop application), AOL Mail and Lotus Notes also made it to the top ten with market share of 5% or less. All other clients combined together occupy a market share of 8%.[8]

2 Theoretical background

This chapter defines categorization, classification and gives a brief overview of information extraction. In addition, some classification methods are described such as decision tree learning, naive Bayes classifier and *k*-nearest neighbour algorithm.

2.1 Categorization

Categorization is generally an unsystematic and creative process of dividing different objects into groups called categories where the members of a category share some similar characteristics. Any object can be part of several categories as it has more than just one characteristic and different categories can have the same characteristics to define them. For example, a **van** can be a member of the **vehicles category** and more specifically it can be a member of **vehicles with automatic transmission category** if it has an automatic gearbox. We can also say that all objects are different (basically every single snowflake is unique) which means that we have as much categories as we have different objects but with that huge amount of information nobody can manage it or understand what we are talking about. Requirements for categories are not too strict and are often related to context or depending on a person's point of view. For example, if the air temperature is 10 degrees Celsius, it is quite chilly in summer but might be pretty pleasant in April or a person can say that all objects that are furry, have four legs and bark are dogs. From this point of view every person has its own way to categorize objects and re-categorize things. As we can see categorization can be a non-formal process of dividing world into categories.[2]

2.2 Classification

Classification is a systematic process of dividing objects into groups that this time are called classes. Each class has a unique name to identify it and they make up a widely known system. Classification varies from categorization because an object belongs to one class only. Classification is more accurate than categorization and it has a more specific approach to how objects can be divided between classes.[2]

A classification scheme is a hierarchical system of classes where every class has its own characteristics which are shared by all members of this class. The best known classification schemes are used in biology, libraries, medicine, and science. This list is not complete because we can classify objects almost everywhere.[2]

For example, in bibliography we have a class of books which are divided into subclasses like mathematics, languages, biology, and chemistry. The subclass mathematics has its own subclasses like algebra, geometry, functional analysis, complex analysis. That makes finding a useful and thematic book easier.

2.3 Information extraction

Information extraction (IE in short) is a process, which extracts structured information out of text for further investigation or use. Next, let's take a look at different steps and aspects of information extraction and find out what kind of information can be extracted from text.[3]

In every email body there is usually some kind of text which contains information. To get essential data out of it, we need to extract information in this text. Next, let us see how information found in text can be divided into different groups by meaning.

First we will try to find all names in the text and classify them. This task is generally called **named entity recognition.** This task focuses on finding names of people, companies, facilities and places that are mentioned in text. It is also important to understand and recognize names that refer to one certain entity.[3] For example, North Atlantic Treaty Organization and NATO refer to the same organization. It is also possible that the same name refers to two different entities as Washington can refer to a person or a location. **Ambiguity** is the term where a word can be understood differently.[3] The most common way to recognize a name in a text is a starting capital letter.

The next important thing is to find out how entities found are related to each other. This is a task of **relation detection and classification.** This task tries to detect relations such as those between family members, employment, a smaller part of something bigger, geographical relations.[3] For instance, Estonia is a country in Europe, a person works for a company.

Next, we need to detect the events where entities are participating at a certain time. This is a target of **event detection**. Here, we also need to figure out which parts of the text refer to the same events that are just differently phrased.[3] For example: "AirBaltic increases price of plane tickets" and "Ryanair goes along with it" - then we are talking about one event that is the increase of the price of plane tickets.

Furthermore, we need to find out, when these events are taking place. This is a task for **temporal analysis** after a process of **temporal expression detection**. Fully qualified temporal expressions contain date, month and year in some widespread form. Temporal expressions are also days of week, time and also indirect expressions like *day after tomorrow* or *next year.* It is important to remember that the duration from one temporal point to another is also a temporal expression. To specify the relative time in calendar with specific date the release date of an article for example or an email message is called the **temporal anchor**.[3] So if we receive a meeting call on 01.01.2011 and it says that the meeting takes place in two weeks then the date of the meeting is 15.01.2011.

After temporal expressions have been detected, the **temporal normalization** is the process that maps found expressions to a specific point in time. This process also maps the start and end points of event.[3]

2.4 Keyword and key phrase

Keywords are important words in a text that give a good overview and description of the content.[7]

A **key phrase** is a set of keywords combined together to a search phrase.[14]

Keyword extraction is a process of identifying important words in a text that best bring out the content of the text. Key phrases can be formed from keywords for more specific search. In different types of texts, keywords can refer to something that is in an important place in the text. **Key-player** for instance refers to an athlete or an organization. **Key-location** is some kind of important place. **Key-verb** best describes the main activity in the text. **Key-noun** is the best word for describing events, locations or persons.[4]

Key phrase extraction is a language processing task to collect the main topics of a document into a list of phrases. After extracting these key phrases and most meaningful words we have a short and synoptic overview of the text. Key phrase extraction is the basic way to extract most important words from the text but it is often a good input for other, more complex text analysis methods.[5]

In statistical key phrase extraction there are different ways to find out how frequently a term is represented in text:

- Relative frequencies – how often a word appears with regard to the total amount of words in text

- Collection frequency – the total number of occurrences of a word in a collection.[9]
- Term frequency–inverse document frequency (tf-idy) – often used in text mining; tf-idf is a statistical measure used to rate how important a word is in a document collection.[18]

2.5 Likey: unsupervised language-independent key phrase extraction

Likey is a statistical approach for key phrase extraction. It is unsupervised and untrained. The only language-dependent component is a reference corpus (this is seen as a sample of language) with which the texts that will be analyzed are compared. Likey selects the words and phrases from the text that best summarize the content by comparing ranks in text and in the selected corpus. The *Likey Ratio* for every phrase is defined as

$$L(d,p) = \frac{rank_d(p)}{rank_r(p)}.$$

In this formula $rank_d(p)$ is the rank value of phrase p in text and $rank_r(p)$ is the rank value of phrase p in the selected reference corpus. These values are calculated according to the frequencies of phrases of the same length. After the calculation a list of results is created where phrases with the lowest ratio are the best candidates for being a key phrase. If phrase p is not in the corpus, the value of maximum rank for this phrase is used where the length is the same as the phrase found in text: $rank_r(p) = max\_rank_r(n) + 1$. As a post-process all words longer than one character face a pre-removal process. If any word in a phrase has a low rank compared to the corpus rank value these words will be removed. This process removes words like “the” and “of”.[5]

2.6 Decision tree learning

In data mining **decision tree learning** is a commonly used classification method. Algorithms used to construct a tree usually work top-down by choosing different variables at each step to split the set of items. This algorithm is used to predict the value of target variable based on different input variables.[15]

To “learn” a tree the source set is split into subsets based on attribute values. The process of splitting is repeated until it has no value to the prediction or when a node has the value of a target variable.[15]

There are two main types of decision trees:

- **Classification tree** – outcome is the class to which the data belongs
- **Regression tree** – this is a process where the predicted outcome is real number

2.7 *k*-nearest neighbour algorithm

***k*-nearest neighbour** algorithm is a lazy and one of the simplest and most fundamental machine learning algorithms for classifying. This should be the first choice if there is weak prior knowledge about classifying data.[12]

The idea of *k*-nearest neighbour classifier is to detect *k* nearest neighbours and compare unclassified object to them. First, there are defined classes and conditions under which an object is assigned to a specific class. When a new object needs to be classified, it is compared to its neighbours. If there is only one neighbour the object will be assigned to same class. If there is more than one neighbour in the specified area the object is assigned to the most frequently occurring class. When the chosen value for *k* is very large it reduces noise but also makes boundaries between classes more unclear.[12]

2.8 Naive Bayes classifier

In simple words **naive Bayes classifier** is a probabilistic classifier (based on Bayes' theorem) that assumes that the presence of some particular feature of a class is not related to the presence of any other feature. The same applies to the absence of any feature in class. As it is a really simple classifier it often outperforms more complicated classification methods.[17]

To classify an object into a class in a set of classes first we need to calculate the probabilities of existing objects to belong to a specific class. If a new object needs to be classified we first find out in which classes are the nearest objects. The nearest objects are objects in a circle drawn around the unclassified object. It is more likely that unclassified object belongs to a class whose representatives have the majority. If we take a look at a situation where there are 40 green and 20 red objects classified and near the unclassified object we have three red and one green object this method classifies new object to red class as there are more red objects near although there is more green objects in total compared to red.[13]

3 Automatic classification offered by email programs

All major email clients offer their own built-in tools for classifying email messages. In this chapter I make real experiments using built in tools of Microsoft office and Mozilla Thunderbird to classify and organize existing email messages and received messages in my Inbox.

3.1 Microsoft Outlook

The Microsoft Outlook 2007 (version 12.0.6557.5001) categorization tool can be found under Tools menu or when making a right click on any email message in the Inbox list.

To explain how this automatic classification feature works in Outlook, I created a connection between my Gmail account and Outlook 2007 installed to my computer. As seen on Figure 1 I have many unread messages.

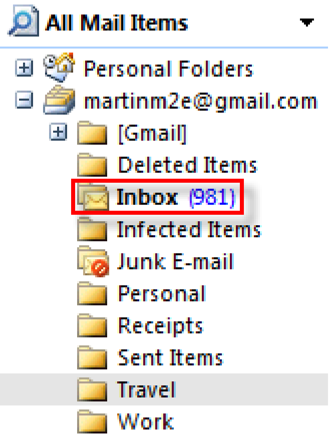


Figure 1. Total amount of messages in Inbox

It takes a lot of time to read all these messages if they are all located in my Inbox and maybe some urgent messages are discovered too late or will be missed at all. To organize the emails the user can create folders and name them as desired to identify emails that can be found in this folder.

To organize my Inbox I created some folders for different emails as seen on Figure 2:

- geni - emails received from website www.geni.com
- UT - emails sent from lecturer in university or email related to some university subject

By using these two folders, I will show the usage of Outlook rule creator main features and ways to organize my Inbox.

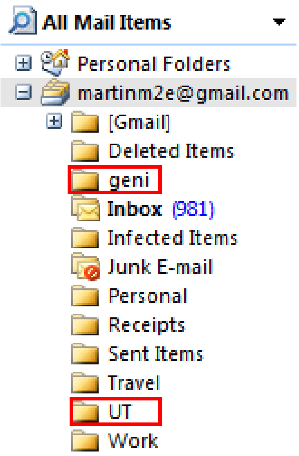


Figure 2. Two created folders for emails

First I am going to move all email messages from www.geni.com to folder "geni". Let us choose an email from proper sender, in my case from Geni. Right click on message and let's choose *Create rule...* as shown in Figure 3.

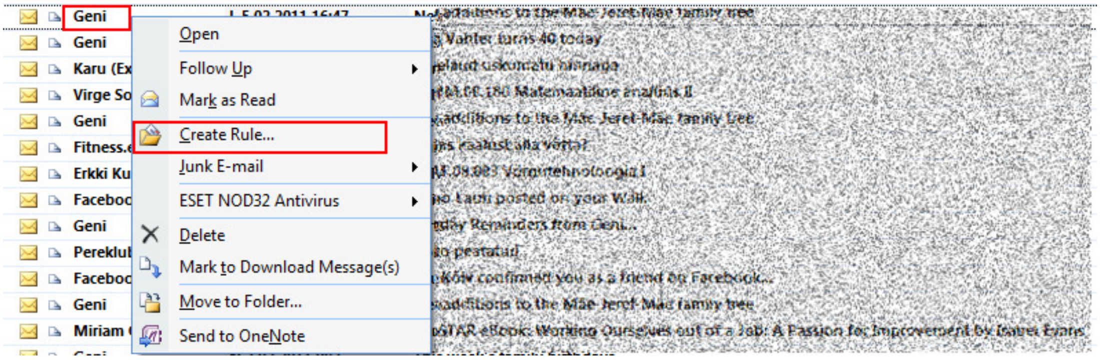
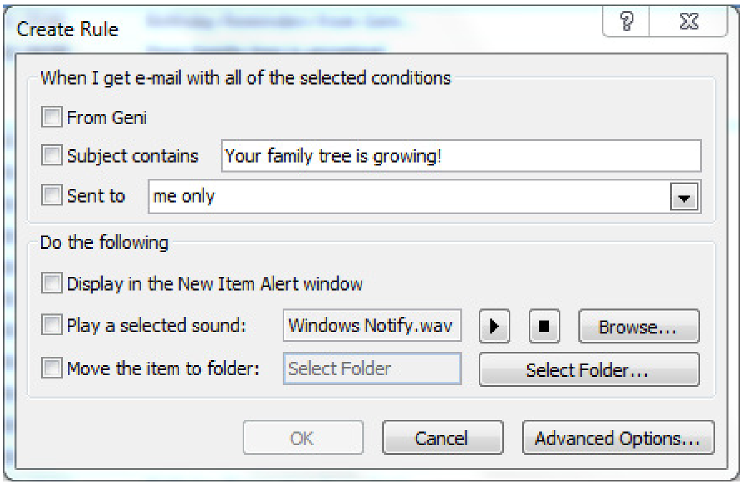


Figure 3. Menu item to start rule creation

After clicking on this menu item a pop-up window appears where the user can choose between different options. This pop-up window is shown on Figure 4. This window is divided into two main parts where in the first part user can choose which emails will be selected with this rule and the other one lets choose the action applied on the message. I will give a brief overview of all options in both sections:

- From - this field is by default filled by the sender of message on which right click was made. This field is prefilled with the sender information from the From field
- Subject contains - this field is by default filled by the subject of the message on which right click was made. This field is prefilled with the information from the Subject field

- Sent to - a drop down list of all addressees in the message on which right click was made
- Display in the New Item Alert window – user can select if the alert appears when the message matching the rule is received
- Play a selected sound - user can select a specific sound which is played when message matching the rule is received
- Move the item to folder - user can choose a folder where the message is moved


Create Rule

When I get e-mail with all of the selected conditions

□ From Geni

□ Subject contains: Your family tree is growing!

□ Sent to: me only

Do the following

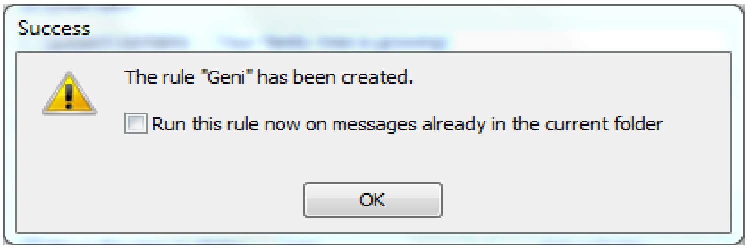
□ Display in the New Item Alert window

□ Play a selected sound: Windows Notify.wav

□ Move the item to folder: Select Folder

Figure 4. Options for creating a rule

As all emails received from www.geni.com contain sender Geni in the From field let us check From Geni field and move these messages to folder “geni”. After selecting Move the item to folder another pop-up appears where user can select proper folder. Let’s choose folder “geni”. When the selections are made, click *OK*. After this another pop-up appears which notifies user that the rule is created and asks if the user wants to run this rule now on messages in current folder (in our case Inbox). This pop-up is shown on Figure 5. If the checkbox stays unmarked created rule will only apply messages received from this point forward.

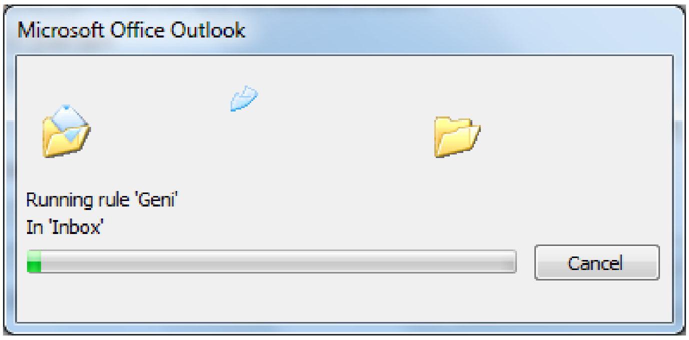

Success

The rule "Geni" has been created.

□ Run this rule now on messages already in the current folder

Figure 5. Confirmation if created rule will apply to existing messages or only to new messages

Let us select the check box and click *OK*. After that the rule starts running and selecting proper messages to move to selected folders. This may take several minutes if Inbox contains a lot of messages. The process can be followed as displayed on Figure 6.



Running rule 'Geni'
In 'Inbox'

Figure 6. Rule running process

After this we can see from the left panel that the number of unread messages in Inbox has decreased and there are unread messages in folder “geni” Figure 7.

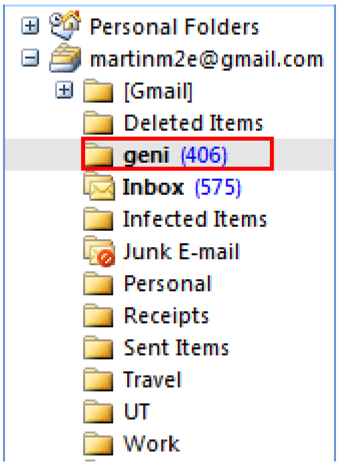
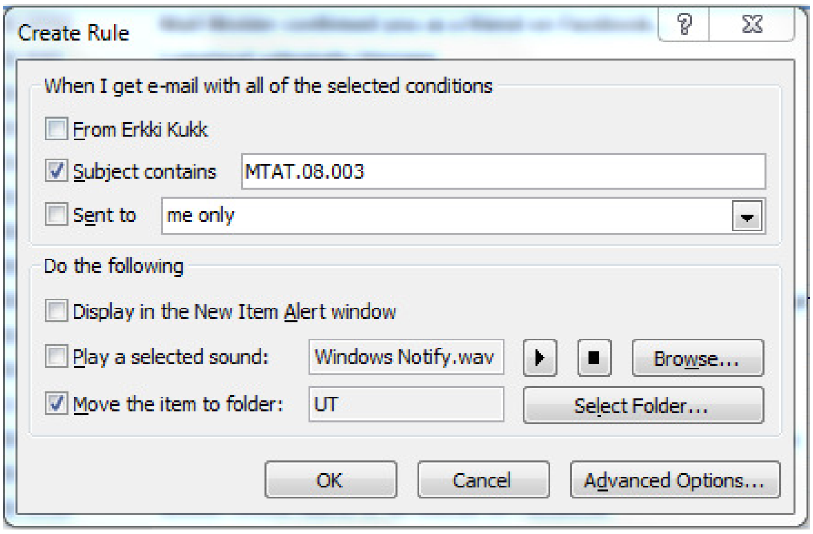


Figure 7. Classified Geni messages in specified folder

Next, let us classify emails related to university to folder “UT”. As in previous case let us find a proper email message and make a right click on it. In the appearing pop-up, this time let us choose different options to create the rule. As lecturers usually include the code of subject to the beginning of email subject, let us use this information to create this rule. Let us insert this code to Subject contains box and choose Move the item to folder value as UT Figure 8.



Create Rule

When I get e-mail with all of the selected conditions

☐ From Erkki Kukk

☑ Subject contains: MTAT.08.003

☐ Sent to: me only

Do the following

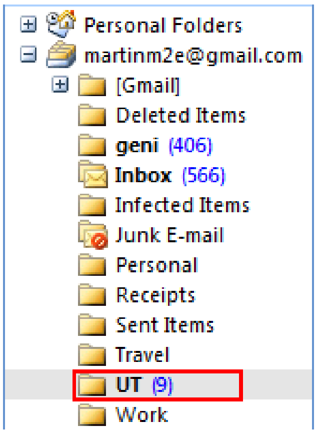
☐ Display in the New Item Alert window

☐ Play a selected sound: Windows Notify.wav

☑ Move the item to folder: UT

Figure 8. Rule creation for UT messages

The following steps are the same as described in previous case and the result is as in Figure 9.



- Personal Folders
- martinm2e@gmail.com
  - [Gmail]
  - Deleted Items
  - geni (406)
  - Inbox (566)
  - Infected Items
  - Junk E-mail
  - Personal
  - Receipts
  - Sent Items
  - Travel
  - UT (9)
  - Work

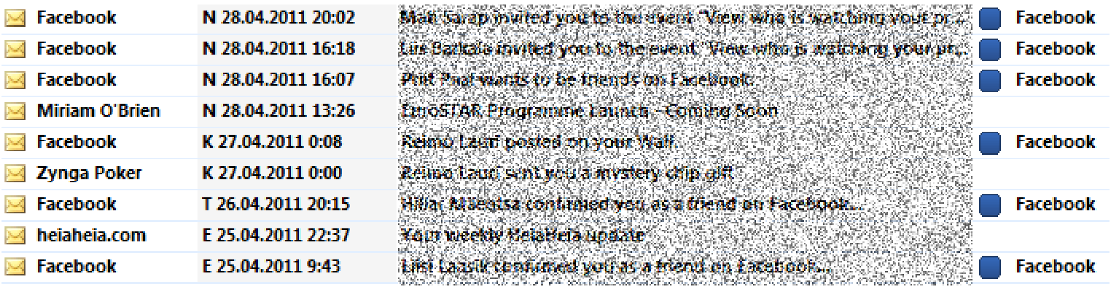
Figure 9. Classified UT messages in specified folder

This is not the only way to create rules to classify messages but this is probably the easiest and fastest way to do this. The time to run the rule the first time might be long but it strongly depends on the amount of messages it is applied on. The more there are messages, the longer it takes.

Different rules can move emails to the specified folder, flag them or delete them as it is specified by the user. It is also possible to create rule combinations that move one email to several folders. For instance - I created the third rule called “Birthday reminders from Geni ...” which moves messages that subject contains sentence mentioned to folder called “geni BD reminder”. As a result, all messages from Geni that contain the text “Birthday reminders from Geni ...” on the subject line will be moved to the specified folder and they are also available in the Geni folder next to all other messages received from www.geni.com. This kind of rule combination can classify emails with some specific characteristic.

Continuing like that all email accounts that are connected to Outlook can be organized and categorized to make finding specific emails and following incoming mail easier. All new messages are analyzed and if they match rule conditions they will be moved to specific folder.

In addition to moving messages to specific folders it is also possible to add flags to emails or mark with to-do remarks. For example I added blue flag named Facebook to all messages received from Facebook. The process of creating this rule was mostly as easy as described before. The user specifies the conditions and selects the actions performed if a message meets the rule. This is also a good way to mark different types of messages to make finding them easier and faster. The result after running this rule is as seen on Figure 10.



✉	Facebook	N 28.04.2011 20:02	[redacted]	■ Facebook
✉	Facebook	N 28.04.2011 16:18	[redacted]	■ Facebook
✉	Facebook	N 28.04.2011 16:07	[redacted]	■ Facebook
✉	Miriam O'Brien	N 28.04.2011 13:26	[redacted]	
✉	Facebook	K 27.04.2011 0:08	[redacted]	■ Facebook
✉	Zynga Poker	K 27.04.2011 0:00	[redacted]	
✉	Facebook	T 26.04.2011 20:15	[redacted]	■ Facebook
✉	heiaheia.com	E 25.04.2011 22:37	[redacted]	
✉	Facebook	E 25.04.2011 9:43	[redacted]	■ Facebook

Figure 10. Flags added to specified messages

This labelling technique is really good to use if any kind of folder and subfolder system is not sufficient for some reasons but after all it keeps the user's Inbox organized and finding emails related to some kind of topic is easier than in a situation where all emails are in the usual black and white colour scheme.

After a test period of two months following the rules in action the results were well above expectations. During this period I received 302 email messages and all of them were classified exactly as needed. Email notifications which were related to Geni or university were in correct folders and also all notifications from Facebook were labelled with proper blue tags in Inbox.

For classifying messages the built in tool in MS Outlook does really good job to organize received email and is really helpful.

3.2 Mozilla Thunderbird

Similarly to Microsoft Outlook, Mozilla Thunderbird (version 3.1.7) also has a built-in categorization tool which can be found under the Tools menu. Distinct from Outlook this feature is not found on the menu after making a right click on any email message in the Inbox. As in Outlook let us try to create similar rules to categorize emails in Thunderbird and see how they work. For this I connected my Gmail account to Thunderbird and downloaded the messages.

As seen on Figure 11 I have many unread messages in my Inbox. For the correct result I cleared the result of Outlook classification and moved all messages back to Inbox. Now let's see, how the categorization and rule creation works in Thunderbird.

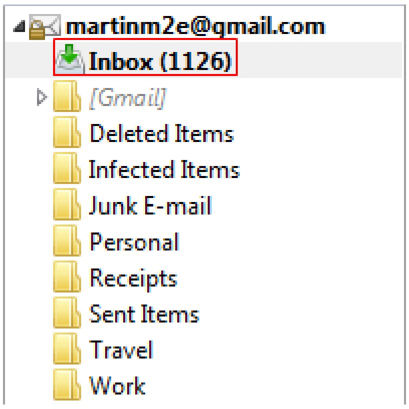


Figure 11. Total amount of unread messages in Inbox

First, let us create the folders "Geni" and "UT" Figure 12 where the messages matching the rules will be moved.

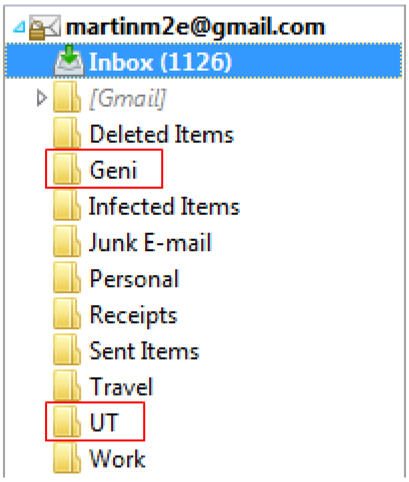
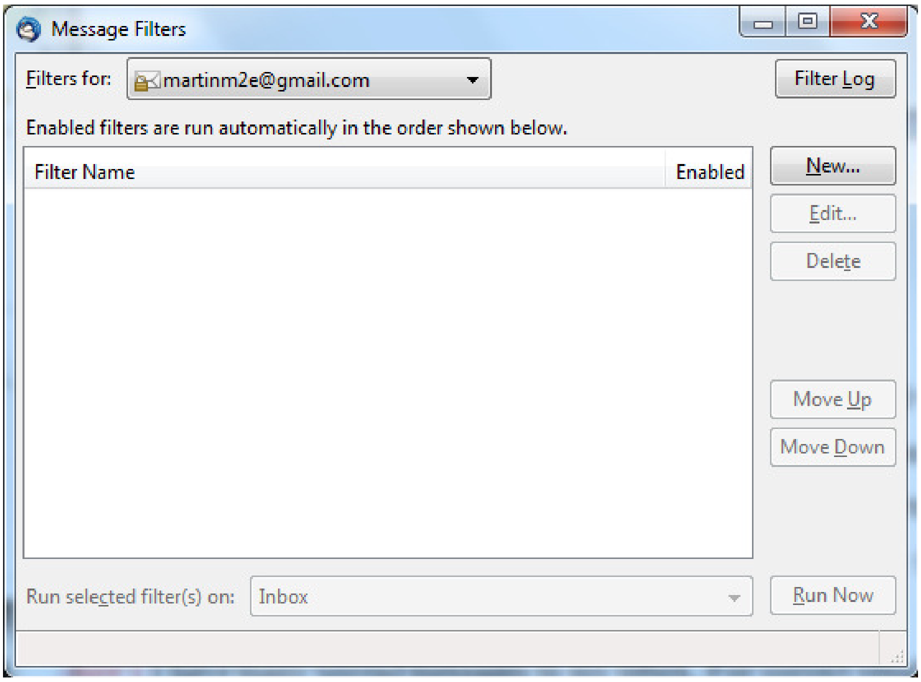


Figure 12. Two created folders for emails

To start creating rules that will categorize emails let us choose Tools > Message filters. A popup appears as in Figure 13.



Message Filters

Filters for: martinm2e@gmail.com

Enabled filters are run automatically in the order shown below.

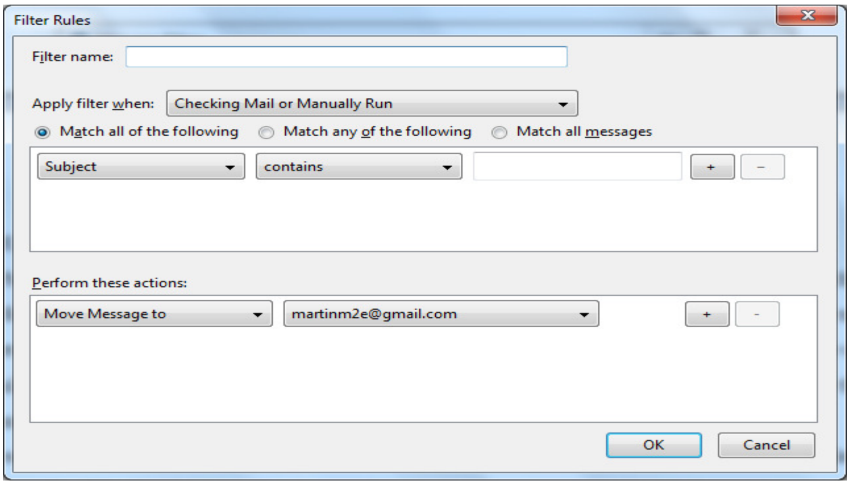
Filter Name	Enabled

Run selected filter(s) on: Inbox

Figure 13. Empty list of rules

As we do not have any rules defined let us click *New...* button. Another popup appears where rules can be defined as in Figure 14. Let us take a brief look at the options that can be used for creating a rule:

- Filter name
- Apply filter when
- Different conditions for rule from what field which keyword is searched for and what kind of action is performed if something is matching the rule



Filter Rules

Filter name:

Apply filter when: Checking Mail or Manually Run

● Match all of the following ○ Match any of the following ○ Match all messages

Subject contains

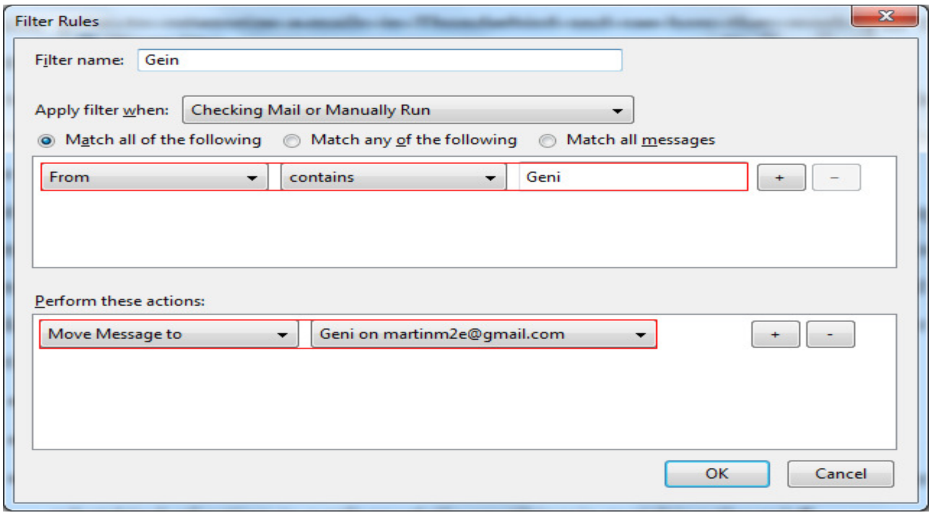
Perform these actions:

Move Message to martinm2e@gmail.com

Figure 14. Options for creating a rule

All main conditions can be selected from the dropdown lists that make creating the rule very easy. Keywords in these lists are very intuitive containing all main fields of an email like From, To, Subject, Body, etc. Although it does not have as many options to combine to create a rule as Outlook it is easier and more intuitive to use. Thunderbird also can Move or Copy messages to specific folder as Outlook only had the option to move. Functionality for copying could be used if more than one rule meets the conditions of an email message.

As in Outlook, let's first create a rule that categorizes all messages sent from Geni to folder called "Geni". To achieve this, let us choose the parameters as on Figure 15. As a result all emails from Geni are moved to folder specified.



Filter name: Gein

Apply filter when: Checking Mail or Manually Run

● Match all of the following ○ Match any of the following ○ Match all messages

From | contains | Geni

Perform these actions:

Move Message to | Geni on martinm2e@gmail.com

Figure 15. Values for created rule

Now let us create the rule for university-related email where we use the code of the subject to categorize the messages to folder specified. The rule will be defined like this: when subject line contains the specified code of subject it will be moved to folder "UT". After applying these two rules to my Inbox the result is on Figure 16 where the amount of unread messages in Inbox has decreased significantly.

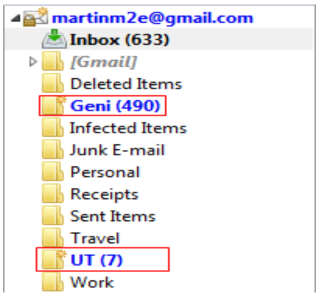
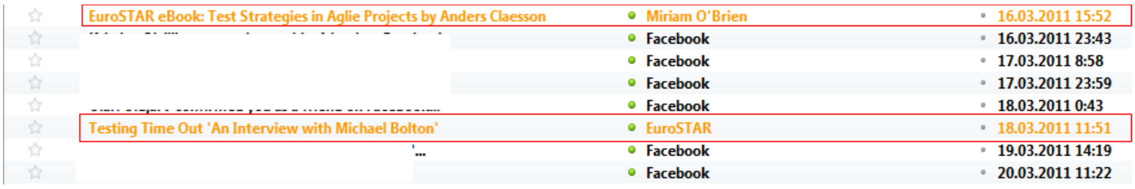

martinm2e@gmail.com
Inbox (633)
[Gmail]
Deleted Items
Geni (490)
Infected Items
Junk E-mail
Personal
Receipts
Sent Items
Travel
UT (7)
Work

Figure 16. Classified Geni and UT messages in specified folder

To see if one message that meets more than one rule will appear in all proper folders let us create the third rule as I did in Outlook and categorize messages sent by Geni with the subject “Birthday Reminders from Geni...” to folder “Geni BD reminders”. To achieve the same situation as in Outlook where these messages were in both Geni and Geni BD reminders folders user must keep in mind to use the Copy Message to option. Otherwise the messages meeting the rule will be moved to folder specified and no copy remains in folder Geni. If the last described situation is the needed one it is useful to use the Move option.

It is also possible to create rules that do not move or copy messages but tag them. Creating that kind of classification rule is pretty much the same as described before only the part where actions applied to messages are described the user can describe what kind of tags are set to messages meeting the rule. A result of this can be seen on Figure 17 where all messages connected to EuroStar are tagged with orange colour as work related.



☆	EuroSTAR eBook: Test Strategies in Aglie Projects by Anders Claesson	● Miriam O'Brien	16.03.2011 15:52
☆		● Facebook	16.03.2011 23:43
☆		● Facebook	17.03.2011 8:58
☆		● Facebook	17.03.2011 23:59
☆		● Facebook	18.03.2011 0:43
☆	Testing Time Out 'An Interview with Michael Bolton'	● EuroSTAR	18.03.2011 11:51
☆	'...	● Facebook	19.03.2011 14:19
☆		● Facebook	20.03.2011 11:22

Figure 17. Flags added to specified messages

This kind of labelling can be used in a subfolder to distinguish emails matching some kind of rule with some kind of more specific aspect.

Rules created in Mozilla Thunderbird were working for two months. During this period I received 302 emails. As in MS Outlook the result here was also really good as I did not notice any mislabelled or misclassified emails. All EuroStar emails were marked with orange labels and the notification emails from Geni were correctly moved to proper folders.

Altogether I can say that the built in tool for classifying emails in Mozilla Thunderbird is definitely worth trying for organizing everyday emails and classifying them with different labels or moving them to different folders.

3.3 Comparison

Using built-in classification tools in Microsoft Outlook and Mozilla Thunderbird is probably the easiest way to organize the incoming email messages. Both tools were easy to use although the right click way to create a rule in Microsoft Outlook might be a bit handier for the user because of some prefilled fields to make creating a rule easier. Both did their job well as I did not notice any misclassified messages so they really help to manage the emails. Instead of moving Mozilla Thunderbird also has the copy option if the user wants to store a message in inbox after classifying it to a specific folder.

Overall, both tools are pretty intuitive to use but Microsoft Outlook may have a slight advantage with its right click option to start creating rules but in any situation the help of both email programs will guide user through the process.

Conclusion

This bachelor thesis describes different classification methods like *k*-nearest neighbour, decision tree learning, naive Bayes classifier, and introduces basic information extraction tasks like named entity recognition, relation detection and others, key phrase and keyword extraction is also briefly presented.

It also gives a brief overview of email message format, most widely used email clients and information about amount of emails sent on years 2009 and 2010 that is showing a real need for classifying emails as the number is growing through years. It also introduces the message overload problem which is becoming increasingly important as we take a look at the statistics about amount of emails received every day.

Some experiments were made with two email clients – Microsoft Outlook and Mozilla Thunderbird - to test their capability to classify emails automatically after creating some rules. These experiments were successfully completed and did their job well as all emails received from the point of creating rule and even previously received emails were classified correctly. To manage daily received emails using these built in features can help user a lot.

As our classification process in Outlook and Thunderbird requires some rule creation then next step from here could be finding some way to totally automatically classify email messages. It would also be interesting to compare mentioned classification methods in real action as parts of some classification software to classify emails.

Elektronkirjade klassifitseerimine

Martin Mäe

Bakalaureusetöö (6 EAP)

Resümee

Tänapäeval on elektronpost üks enimkasutatud rakendusi, mis arvuti jaoks on läbi aegade leiutatud. Kuna saadetavate ekirjade hulk kasvab kiiresti oleme me aina enam seismas silmitsi probleemiga, kus infot tuleb liiga palju ja selle hulgast vajaliku leidmine muutub üha raskemaks. Antud töö eesmärk on anda ülevaade erinevatest klassifitseerimismeetoditest ja võimalustest antud probleemi lahendada läbi ekirjade klassifitseerimise.

Antud töö annab ülevaate erinevatest klassifitseerimismeetoditest, võtmesõnade ja võtmefraaside leidmisest ning sellest, kuidas tekstist leitud informatsiooni erinevatesse klassidesse jagada.

Samuti tutvustab lühidalt elektronkirja formaati, annab ülevaate, milliseid programme kasutatakse enim elektronkirjade lugemiseks ning toob välja statistika saadetud elektronkirjade hulga kohta aastas. Samuti tutvustab põgusalt suurest ekirjade hulgast põhjustatud infokülluse probleemi.

Töö lõpus viiakse läbi ka reaalne katse kasutades meililugemisprogramme – Microsoft Outlook ja Mozilla Thunderbird – ja neisse sisseehitatud kirjade klassifitseerimise funktsionaalsust. Katse tulemusena võib öelda, et mõlema meiliprogrammi vastav funktsionaalsus töötab hästi ja on kasutajale igapäevaselt suureks abiks, et hoida saabuvad kirjad kontrolli all ja klassifitseerida neid vastavalt kasutaja soovile, et seeläbi lihtsustada vajaliku info leidmist.

Bibliography

[1] D. Fisher, A.J. Brush, E. Gleave, M. A. Smith. Revisiting Whittaker & Sidner's "Email Overload" Ten Years Later. In *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work CSCW 06, 2006.*

[2] E. K. Jacob. Classification and Categorization: A Difference that Makes a Difference. Available from: `http://findarticles.com/p/articles/mi_m1387/is_3_52/ai_n6080402/`, pages 3-10 [Last visited May 25, 2011].

[3] D. Jurafsky, J. H. Martin. Speech and Language Processing: An Introduction to Speech Recognition, Computational Linguistics, and Speech Recognition, Second Edition. Chapter 22: *Information Extraction*, page 725-749, 2009.

[4] R. Nallapati, J. Allan, S. Mahadevan. Extraction of Key Words from News Stories. `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.3.5301&rep=rep1&type=pdf` [Last visited May 25, 2011].

[5] M.-S. Paukkeri and T. Honkela. Likey: Unsupervised Language-independent Keyphrase Extraction. In *Proceeding SemEval '10 Proceedings of the 5th International Workshop on Semantic Evaluation,* page 162-165, 2010.

[6] S. Whittaker and C. Sidner. Email overload: exploring personal information management of email. In *Proceeding CHI '96 Proceedings of the SIGCHI conference on Human factors in computing systems: common ground*, 1996.

[7] Keyword – Dictionary.com. Available from: `http://dictionary.reference.com/browse/keyword` [Last visited May 25, 2011].

[8] Email client market share – limitus. Available from: `http://litmus.com/resources/email-client-stats` [Last visited May 25, 2011].

[9] Inverse document frequency. Available from: `http://nlp.stanford.edu/IR-book/html/htmledition/inverse-document-frequency-1.html` [Last visited May 25, 2011].

[10] Internet 2009 in numbers – Royal Pingdom. Available from: `http://royal.pingdom.com/2010/01/22/internet-2009-in-numbers/` [Last visited May 25, 2011].

[11] Internet 2010 in numbers – Royal Pingdom. Available from: `http://royal.pingdom.com/2011/01/12/internet-2010-in-numbers/` [Last visited May 25, 2011].

[12] k-nearest neighbour – scholarpedia. Available from: `http://www.scholarpedia.org/article/K-nearest_neighbor` [Last visited May 25, 2011].

[13] Naive Bayes Classifier – StatSoft, Electronic Statistics Textbook. Available from: `http://www.statsoft.com/textbook/naive-bayes-classifier/` [Last visited May 25, 2011].

[14] Keyphrase – webopedia. Available from: http://www.webopedia.com/TERM/K/keyphrase.html [Last visited May 25, 2011].

[15] Decision tree learning – wikipedia, the free encyclopaedia. Available from: http://en.wikipedia.org/wiki/Decision_tree_learning [Last visited May 25, 2011].

[16] Email – wikipedia, the free encyclopaedia. Available from: http://en.wikipedia.org/wiki/Email [Last visited May 25, 2011].

[17] Naive Bayes Classifier - wikipedia, the free encyclopaedia. Available from: http://en.wikipedia.org/wiki/Naive_Bayes_classifier [Last visited May 25, 2011].

[18] Tf-idf – wikipedia, the free encyclopaedia. Available from: http://en.wikipedia.org/wiki/Tf%E2%80%93idf [Last visited May 25, 2011].