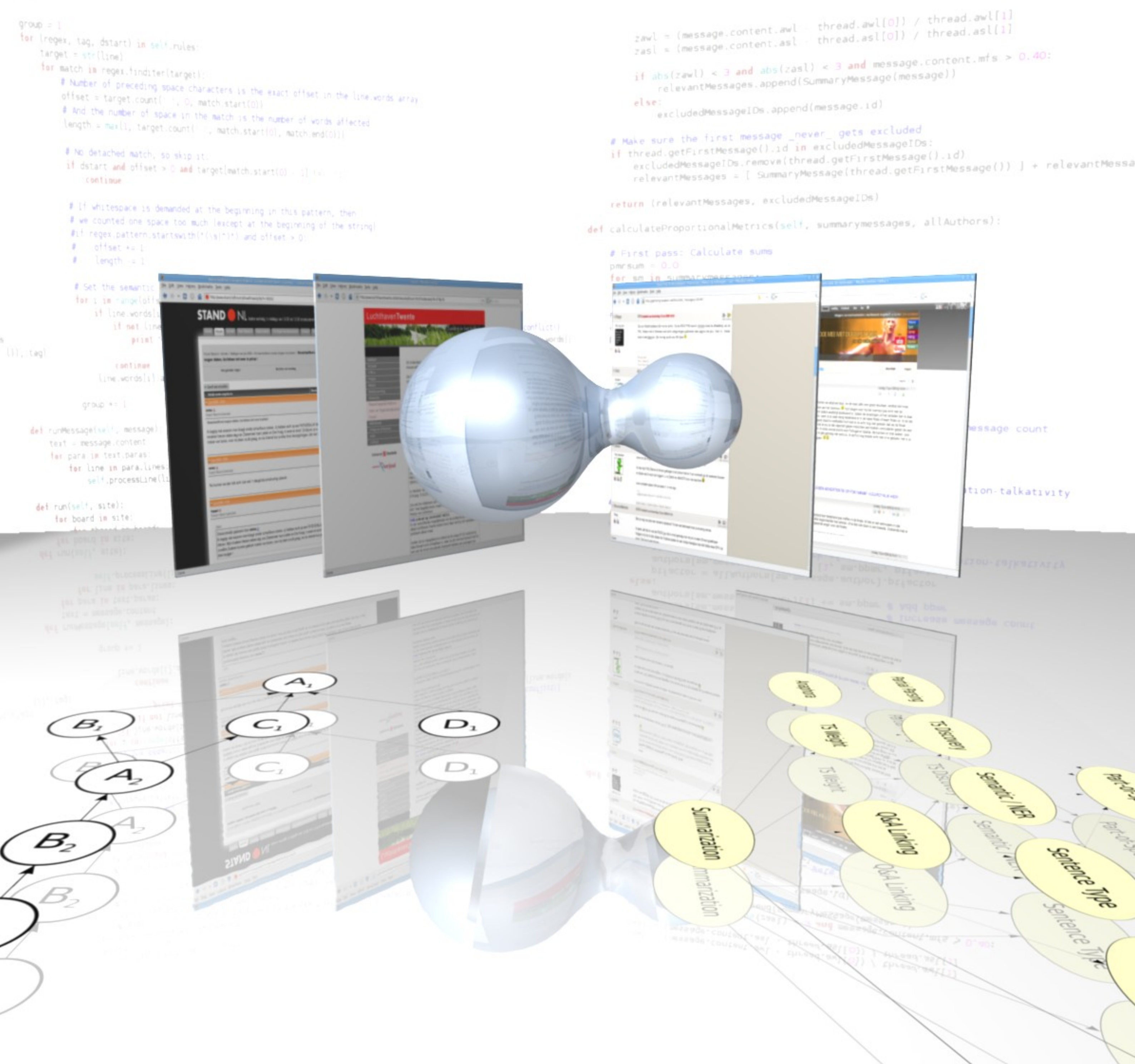
Automatic Discussion Summarization

A Study of Internet Fora



Almer S. Tigelaar

Human Media Interaction

University of Twente
Enschede - The Netherlands

LSBJ 4-81-516234-2

Master's Thesis

Automatic Discussion Summarization
A Study of Internet Fora

Ing. Almer S. Tigelaar

June 2008

Graduation Committee:
Dr. Ir. H. J. A. (Rieks) op den Akker
Dr. D. K. J. (Dirk) Heylen
Prof. Dr. Ir. A. (Anton) Nijholt

Human Media Interaction Group
Department of Computer Science
University of Twente
The Netherlands

‘Any sufficiently advanced technology is indistinguishable from magic.’

(Arthur C. Clarke, 1917-2008)

‘Everything is theoretically impossible, until it is done.’

(Robert A. Heinlein, 1907-1988)

‘I do not fear computers. I fear the lack of them.’

(Isaac Asimov, 1920-1992)

Summary

English

The purpose of this research was finding automated methods to summarize discussions held on Internet fora. A second goal was building a functional prototype implementing these methods. This explorative study tries to find what technologies and methods can be usefully combined into an automatic discussion summarizer. The focus of this research is on two types of threads: Problem-Solution and Statement-Discussion. Although Dutch is the main language used, much of the presented work is also applicable to other languages.

Compared to summarization of unstructured texts (and spoken dialogs) the structural characteristics of threads give important advantages. We studied how these characteristics of discussion threads can be exploited. Messages in threads contain explicit and implicit references to eachother. They also have a relatively structured internal make-up. Therefore, we call the threads hierarchical dialogues. The algorithm produces one summary of an hierarchical dialogue by cherry-picking sentences out of the original messages that make up the thread. For sentence selection we try to find the main focus of the discussion that is useable to obtain an overview of the discussion. The system is build around a set of heuristics based on observations of real discussions.

We developed a functioning prototype. The performance of this system was evaluated for Dutch only, but the system also supports English. Various aspects of parts of the system and the methods developed were evaluated. Much can be done to improve the current approach. Although the idea of building a summarization system in the way presented in this thesis is feasible.

Dutch

Het doel van dit onderzoek was het vinden van methoden voor het automatisch samenvatten van discussies die gehouden worden op Internet fora. Een tweede doel was het ontwikkelen van een functionerend prototype dat deze methoden gebruikt. In deze verkennende studie wordt getracht uit te zoeken welke technieken en methoden zinvol kunnen worden gecombineerd tot een automatische discussie samenvatter. De focus van dit onderzoek is op twee type discussies: Probleem-Oplossing en Standpunt-Discussie. De gebruikte hoofdtaal is het Nederlands, hoewel veel van het werk ook toepasbaar is voor andere talen.

Vergeleken met het samenvatten van ongestructureerde teksten (en gesproken dialogen) geven de structurele eigenschappen van Internet discussies belangrijke voordelen. We hebben bestudeerd hoe deze karakteristieken kunnen worden gebruikt. Berichten in discussies bevatten expliciete en impliciete verwijzingen naar elkaar. Ze hebben ook een relatief gestructureerde interne opbouw. We noemen de discussies daarom hierarchische dialogen. Het algoritme levert één samenvatting op van een hierarchisch dialoog door zinnen te plukken uit de oorspronkelijke berichten waaruit de discussie bestaat. Voor zinsselectie proberen we de rode draad van de discussie te vinden die bruikbaar is voor het krijgen van een overzicht van de discussie. Het systeem is opgebouwd uit een verzameling heuristieken die gebaseerd zijn op observaties van echte discussies.

We hebben een werkend prototype ontwikkeld. De prestaties van dit systeem zijn alleen voor het Nederlands geëvalueerd, maar het ondersteund ook Engels. Verscheidene deelaspecten van het systeem en de ontwikkelde methoden zijn geëvalueerd. Er kan veel worden gedaan om de huidige aanpak te verbeteren. Echter, het idee van het bouwen van een samenvattingssysteem op de manier waarop dat in deze these is gedaan is steekhoudend.

Preface

'Ideas don't stay in some minds very long because they don't like solitary confinement.'

(Anonymous)

This research is primarily the brainchild of my first tutor Rieks and me. We went through several possible graduation projects during the end of the summer of 2007. One of our frustrations was that on-line discussions often tend to become repetitive. People frequently do not seem to take the trouble to properly read what has already been discussed. This assignment was also the least 'crystallised' one at the time, which is also the reason I chose it. There is a lot to be said for improving existing methods and technologies, but I wanted to do something that was both creatively challenging, explorative and unconvential. What lies before you is the result which represents about seven and a half months of work.

I did some things deliberately different from what is the usual waterfall-style of research that usually starts with an intensive literature study. I performed only orientation on literature in the beginning, developing methods and software in parallel with reading in-depth literature. I found this to be a very useful approach especially for prototyping. However, there are downsides too. In the beginning my goals were not so clear-cut. It was actually quite a challenge to set concrete and interesting research goals. This can cause one to drift in many directions, some of which are less important. Fortunately, I had Rieks who kept me on the right track at such times which shows how important a tutor can be.

What you will *not* find in this research is extensive use of machine learning technologies. While I realise that using such technologies is the trend nowadays, I do not believe that it is the appropriate methodology for explorative research. Remember, that much of our field was initially based on heuristics. Think for example of *tf.idf* whose underlying principles are still used to this very day. Machine learning provides a very useful toolbox, but when used in the wrong way it can create false impressions. This is due to problems inherent to machine learning such as lack of data and overfitting. In much of the literature I studied, I noticed people have trouble explaining their results when applying machine learning with tons of (usually lexical) features.

Applying machine learning does not foster a feeling for the data. Looking at it yourself, trying to find patterns and experimenting with it does. Annotation studies employing machine learning are much more useful in that sense, since they can aid in seeing the patterns. That said, I do think that machine learning is very usable for many tasks within Natural Language Processing (NLP) that are relatively well understood (like tokenisation and Part-of-Speech tagging). However, for those tasks, heuristic and rule-based approaches were also used prior to applying machine learning techniques. Notice any pattern?

Compared with other research this thesis covers a relatively wide array of language technologies. My learning goal was to see how technologies, usually treated in isolation, fit together. This broad focus naturally sacrifices some depth. Most technologies were self-implemented as components of the prototype. Hence, a lot of software testing was performed. Where possible system components were also evaluated.

I think the primary contribution of this research to the field of NLP is that it shows the task of summarization can be greatly aided by metadata combined with a set of heuristics. In addition, it also makes a case for treating summarization of (hierarchical) dialogues differently from traditional monologues. The focus of the research is Dutch instead of English. Showing that explorative research can also be applied to a minority language. It are, after all, the underlying patterns that matter.

Making summaries automatically remains a challenging task. I hope that this research provides a basis and direction for future research in this area.

- Almer සම්පත් Tigelaar

June 2008

Acknowledgements

I would especially like to thank Rieks op den Akker for his guidance and feedback. Also Ruben Wassink for his thorough review of this thesis and Marco Gerards for his comments and insights. Dirk Heylen for providing me with subjectivity literature early in the process, Mariët Theune for testing my initial experiments with determining sentence polarity, Anton Nijholt for providing feedback on early drafts of the introduction chapter and Gert Bolmer for several textual corrections.

I also want to thank my family and especially my parents who have been very supportive of my studies through the years, my friends, the neighbours in my dorm for providing a good social environment, my fitness and party buddies and all the people who have (previously) been instrumental in determining the direction of my studies.

Contents

Chapter 1

Introduction

> 'Give a person a fish and you feed them for a day; teach that person to use the Internet and they won't bother you for weeks'
>
> *(Anonymous)*

With the advent of the Internet, it became possible to send messages across large distances in the blink of an eye. One of the Internet's first killer application was electronic mail (e-mail). It appeared as early as 1972, providing a new way for people to communicate. E-mail was largely geared towards one-on-one communication which lead to the birth of new one-to-many messaging technologies like the mailing list (that is essentially built on top of e-mail facilities) and Usenet newsgroups [56, 52].

Nowadays, the World-Wide Web is a popular vehicle for deploying all kinds of applications. Communication services that have protocols of their own are also made accessible through web interfaces like webchat and webmail. This research focuses primarily on web-based discussion fora. Usenet newsgroups can be seen as the pre-cursor to these fora.

There are many discussion fora on the web usually devoted to a specific topic or a group of related topics. The way in which fora are used varies wildly. From basic question-answer exchanges to full-blown society-issue discussions. This variety in content makes it an interesting, but also difficult medium for Natural Language Processing (NLP).

As a discussion becomes longer, it requires increasingly more effort of the user to follow. Consequences of not getting the gist of a discussion are posted messages containing arguments or solutions that have already been mentioned earlier. Related to that is the act of purely venting one's opinion in a post thereby reducing a forum to a soapbox instead of fostering a discussion. These phenomena, among others, make it more difficult to learn from the discussion content [25].

Hence the idea of creating supportive technologies for Internet fora. It would be very useful to point out (parts of) relevant messages in a discussion to a user. Not only would this save time, it would make it easier to learn from a discussion, lower the effort threshold to make contributions to a discussion, and improve the quality of such contributions (thereby also reducing the load on forum moderators). Such technologies can be viewed as an extension of the normal search process. Many fora already offer some search facility. However, these are usually simple keyword based retrieval systems and are not capable of capturing the gist of a discussion [23].

There are many solutions that could aid the user in understanding a discussion. An indirect route would be checks during message input, for e.g. repetition, to safeguard the quality of a discussion. Another option would be providing background information on what is being discussed. However, we focus on one solution to this problem: summarization.

This research focuses on a way to provide useful summaries of several types of discussion threads in Internet fora. The idea of summarizing threads is not new and is referred to as *hierarchical discourse summarization*. However, very few researchers have concentrated on one-to-many type of (written) discussions. Those that did almost exclusively focused on newsgroups [24, 50, 54]. Although there is some recent work focusing on blog comments as well [41].

A larger amount of research which has similarities with the task at a hand is specifically related to summarizing e-mail threads [13, 53, 74, 92]. Some even focuses on finding the relation between questions and answers which is important to understand the crux of a discussion [50, 63].

Nevertheless, e-mail has different characteristics than discussion groups. For example, the discussions have fewer participants. According to Dalli [17] e-mail threads are relatively short with about 87% having three messages or less. It is also an accepted practice to reply to unrelated messages to save oneself from having to specify the recipients again. This is not applicable to Internet fora. Additionaly e-mail has a tendency towards mixed formal and informal content whereas the latter is more common on fora. Another difference is the absence of a thread structure on many discussion boards. They exhibit a predominantly flat message structure leaving the discovery of hierarchy up to the user.

There have also been studies however that specifically analyze on-line discussions, but not in the context of summarization. Kim, et al. [48] focus on finding a way to semi-automate grading based on the quality of discussion participation. Their corpus consists of discussion threads from University of Southern California (USC) undergraduate computer science students. In a related paper they use speech acts at the message level to find threads with unanswered questions and confusions [47]. Instructors can use this information to help determine where to focus their attention. Their findings are interesting: about 95% of all threads start with a question post, that question is directly followed by an answer in 84% of all cases. They also found that acknowledgements are usually found at the end of a thread (73%). Nevertheless, their corpus is very domain specific and consists predominantly of threads that consists of only two messages (question-answer pairs). Such threads are far less interesting for summarization. Since the usefulness of (and the need for) a summary increases with the length of a thread.

Feng, et al. [25] use the same corpus as Kim, but try to detect the topic of the threaded conversation for question-answer functionality. They also focus on slightly longer threads (four messages on average). However, similar to Kim they rely on hand annotated speech acts assigned to messages. They take into account the authority of authors in a thread (they refer to this as trustworthiness). They use a combination of the HITS algorithm [51] and their manual annotations. However, they do not motivate why this cyclically oriented algorithm is necessary for their data which is essentially a directed acyclic graph.

Our research is related to document summarization. There exists evidence that methods that work well for the traditional single-document summarization task fare poorly for discussion dialogues. Treating a discussion as one monolithic document simply does not work [50]. Our interests match better with multi-document summarization. However, there are significant differences between the traditional multi-document summarization task, that focuses on summarizing multiple monologue documents covering the *same* subject, and the task at hand, which targets dialogues by means of message exchange.

Finally, our summarization algorithm also supports a preference for including subjective or objective content. This idea of using subjectivity as a factor for summarization has been voiced before by Wiebe and Hatzivassiloglou in the form of an aid for relevance judgements. We follow their idea with the difference that we apply it as an input to our system as opposed to an extra characteristic of the output [32].

The following paragraphs clearly define the terminology and the functional and research goals.

1.1 Overview and Terminology

A short overview of various means of Internet communication is shown in table 1.1. The conceptually closest non-electronic counterpart is shown for each of them. The rest of the columns indicate the typical sender-receiver ratio (multiplicity), the nature of the communication (synchronous or asynchronous) and the way of message delivery (instant, stored). With push we mean that messages are delivered to (a resource controlled by) the user, whereas pull requires manual action on the part of the user. Note that the presented patterns are rough usage indicators and not intended as strict dividing lines between the various technologies.

This research primarily focuses on *many-to-many & store-pull* type communication which we will refer to as *forum* for the remainder of this document. It applies to mailing lists as well

Table 1.1: *Overview of Internet messaging technologies (research focus colourshaded).*

Non-Electronic	*Multiplicity*	*Nature*	*Delivery*	*Examples*
Conversation	1-1	Sync	Instant	ICQ, MSN
Conversation	n-n	Sync	Instant	IRC, Webchat
Letter	1-1	Async	Store-Push	E-mail
Newsletters	1-n (n-n)	Async	Store-Push	Mailing list
?	n-n	Async	Store-Pull	Newsgroups, Web Fora, Weblogs, Wiki's

and in essence it is also practically applicable to e-mail and to some extent even to instant messaging. However, note that these latter two have several important different structural differences (multiplicity, nature and delivery as shown in the table). Especially, these contentual data characteristics makes these media deserve studies of their own, several of which already exist [26, 53, 74, 99].

A wide variety of terms is used to indicate the concepts in the domain of fora. We will adopt the following terminology:

- *Sites* are places where one or more *boards* are hosted.
- *Boards* (or *Fora* or *Groups*) are devoted to a general topic.
- *Threads* (or *Topics*) consist of (one or more) related messages within a *board* that concern a specific topic. The topic is frequently expressed via a topic title.
- *Messages* (or *Posts*) are coherent texts posted either in an existing *thread* or as the start of a new *thread* (*Initial Post*).

Sites are usually devoted to a specific domain[1] (e.g. computers). Fora on the site encompass some topic within that domain (e.g. motherboards) and topics focus on specific issues or questions in such a domain (e.g. “How to fix [SomeIssue] with my [BrandName] motherboard?”). Users can post messages in an existing thread (follow-up posts or replies) or post a message that starts a new thread (initial post). Threads can be *closed* (usually by a moderator) in which case no new posts can be made to the thread.

A number of user types are involved in these fora:

- *Administrators* handle the technical issues surrounding the site (or a specific forum).
- *Moderators* take care of approving new messages or removing irrelevant messages.
- *Members* can have elevated privileges such as access to private boards.
- *(Anonymous) Users* haves access to (parts of) the site and can (sometimes) also post.

Note that these concepts map very well onto the related and popular weblog domain. Sites frequently host multiple weblogs (Boards). Here the first message (Initial Post) is usually posted by the owner of the weblog (Member) concerning some subject (Topic). Follow-up messages are called reactions or comments (Posts).

[1] Sometimes sites host a multitude of fora covering different domains.

1.2 Goal and Motivation

With the terminology defined the main purpose of this research can be stated:

> Automatically Summarizing Threads

Recall from the previous section that a thread is an exchange of messages between forum users about a related topic. In this research we only consider threads that consist of at least two messages by different authors in line with the definition of Kim, et al. [48]. We also do not handle topic drift and assume that threads remain on-topic.

Many forum search facilities can aid in finding threads with interesting topics, but that is about as far as the user is automatically assisted in a useful way. Frequently the main question (or statement) is clearly represented in a title of a topic, but to find the actual answer(s) (or most relevant reactions) to the question (statement) the thread needs to be read manually. Instead of this tedious process we pose that it would be highly useful to be able to obtain a summary of a thread automatically.

A second motivation is to see how existing Natural Language Processing (NLP) techniques can be combined effectively to form an integrated system. While useful research has been done on separate topics and areas of NLP, there is fewer research into fusing these methods and technologies. These studies are relevant since they give an indication of the combined real-world potential of the many existing building blocks.

Fora can be used for other purposes than pure discussions. Frequently Asked Questions (FAQ) and threads concerning posting rules are very common. Instruction manuals and reviews also appear regularly. Mass topic threads where posters post all kinds of (sub)issues regarding a certain main topics (effectively pulling the *board* level to the *thread* level) are also encountered. All of these are *not* subject of this research.

We focus exclusively on threads of the following types:

- *Problem-Solution*: A main question is posed, replies are posted, and (optionally) follow-up questions are asked.
- *Statement-Discussion*: An (opinion) statement is voiced, replies are posted, stances are revised.

The term *thread* when used in the remainder of this document refers exclusively to these types of threads.

For Problem-Solution threads the ideal output would be a clear problem definition and one or more possible solutions (similar to the concept of conversation focus as presented by Feng, et al. [25]). For Statement-Discussion threads, the main statement should be output in addition to the major stances of authors in the discussion (and how these changed over time).

Statement-Discussion threads are generally very ‘wide’. As a follow-up to the initial post, many authors respond to give their insights. Problem-Solution threads are usually ‘narrow’ and involve more contributions of the initial author to refine the exact problem and work towards the solution. Nevertheless, in both types of threads a main focus can be found which what should be captured by the summarization process.

The prime focus of this research is the creation of *monolithic extractive data-focused semi-informative* summaries based on *hierarchical dialogues*. This term is explained in section 6.1.

An other dimension to this is what kind of information a user is looking for. For Problem-Solution threads this might be primarily objective information, whereas for Statement-Discussion threads subjective information is more telling. To aid with this, we add an extra dimension to the summarization process: the ability to indicate a preference for either a more objective or subjective summary.

Threads are not static. They develop over time as new posts are made. Hence there is also a time dimension. Our prime interest is in threads that have already developed over time consisting of at least several postings.

Note that with the exception of emoticons we do not consider threads that contain images or other multimedia content in this research. References to some external sources contained in the message are detected, but not given preferential treatment.

1.3 Research Questions

There are several research questions with respect to the goal that we would like to answer.

1. How to automatically build summaries of threads?
   a) What are structural characteristics of threads?
   And how can these characteristics be exploited for summarization purposes?
   b) What technologies and methods are necessary for this exploitation?
   c) How should these technologies and methods be combined?
2. What is the performance and usefulness of a thread summarization system?
   a) What is the performance of the individual components? (systematic evaluation)
   b) How do users rate the performance of the entire system? (user evaluation)
   c) What do different types of users think of the usefulness of such a system?
      i. Are automatically built summaries a useful addition to the search process?
      ii. Does the objective-subjective summarization preference add value?

1.4 Approach

A first step to defining a summarization system is regarding it as a black box and clearly defining its inputs and output. These are as follows:

Inputs:

- A (flat) message thread.
- The size of the desired summary (expressed as a compression ratio or a desired number of lines[2]).
- Possible preference for objectively or subjectively formulated content.

Output:

- A summary of the input thread with the desired size and objectivity.

The insights presented in this thesis are based on real data. Forty threads (half of which are Statement-Discussion threads and the other half Problem-Solution threads) on a technical forum[3] were manually examined. Several other resources were also used for enhancements and checks[4].

1.5 Language

The language this research primarily focuses on is Dutch. Nevertheless, we developed our prototype bilangually with United States English as the second language. The reason for this is that it forces one, from the very start, to separate language dependent resources from the main ideas and algorithms. The current design allows for adding support for other major Western European languages by simply extending several language resource files. Chapter 7 contains details regarding the modules in the prototype that are language dependent.

Using Dutch as the primary language provides some extra challenge. Many of the traditional Natural Language Processing resources and techniques are geared towards English. Resources for Dutch, like large corpora, are more scarce.

Evaluation and thorough testing was done only for Dutch. Hence, it is unknown how good the performance is for English. But we expect the performance to be as good or better, given the fact that resources are more plentiful for English.

As a final remark on language, keep in mind that many of the important techniques that are central to this thesis, like thread structure, are for the most part language independent.

[2] Where the term 'line' is used in this thesis, it is considered equivalent to 'sentence'.

[3] `http://gathering.tweakers.net`

[4] Primarily `http://www.stand.nl/forum` and `http://forum.fok.nl`, but earlier also the now defunct forum on `http://luchthaventwente.nl`

1.6 Thesis Structure

Throughout the thesis we will gradually work from the defined inputs and their characteristics to the output. We first take a look at the data under consideration (chapter 2) which gives rise to employing some basic (essential) Natural Language Processing techniques (chapter 3). Some related and higher-level technologies have main sections of their own (chapters 4, 5 and 6). A design of the entire system can be found in chapter 7 which is followed by an evaluation section (chapter 8). Conclusions are drawn in chapter 9 and the thesis is closed by a section on future work (chapter 10).

Note that the evaluation in chapter 8 is a broad evaluation of the entire prototype system. (External) evaluation results of individual parts of the system are referred to in the section describing the underlying technology. When such evaluations were done as part of this study the results are generally included as appendix (specifically appendices A and B).

Chapter 2

Data Structure

> 'We've heard that a million monkeys at a million keyboards could produce the complete works of Shakespeare; now, thanks to the Internet, we know that is not true.'
>
> *(Robert Wilensky)*

We need to understand the characterics of the data under consideration to be able to exploit these for the end goal of summarization. This chapter looks at several important properties and derives suitable methods from them that are applied later in the summarization process.

2.1 General Characteristics

Threads are essentially a concrete incarnation of written multi-party dialogue. They have a specific set of characteristics. Several important ones are [50]:

- *Domain-independence.* There is a wide-variety of fora covering many subjects.
- *Informality.* The writing style is generally less formal than for other media like e-mail.
- *Diverse message structure.* There are little structural clues present in messages.
- *Multiple authors.* The dialogues are a mixture between contributions of many authors with different styles.
- *Low signal-to-noise ratio.* Spam, off-topic posts and trolls negatively affect quality.
- *Dialogue structure.* Messages refer to each other yielding a communication structure.
- *Author tracking.* Some fora provide extra background information on their authors that can be exploited.

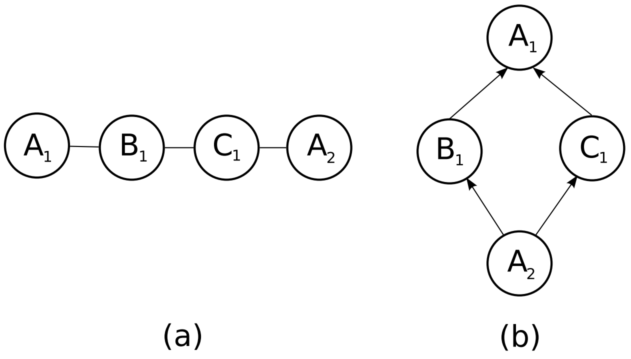


Figure 2.1: *Conceptual thread structure discovery. Messages represented by circles. Each character represents a unique author and each number a unique post by that author. Links in (a) represent temporal links (C_1 was posted after B_1) whereas links in (b) show references (A_2 refers to B_1 and C_1).*

The content of the messages under consideration is generally recognised as being a cross between the informal Instant Messaging (IM) / Chat and traditional writing such as letters [17].

The signal-to-noise ratio on moderated fora is generally higher than on unmoderated ones, and on fora without any form of registration (also called Shoutboxes). For this purpose, we have also tried to 'emulate' some of the task that is traditionally executed by forum moderators: filtering out certain messages. This can be found later in this chapter. First, we focus on the structure of a discussion.

2.2 Thread Structure

2.2.1 Discovering

Many fora display flat lists of messages. The only explicitly encoded information in such lists is usually the date and time of each posted message. In fact, the authors in the threads usually reply to (one or more) specific messages. We broadly distinguish two types of references used in Internet fora:

- Explicit mentioning of author names.
- Quoteblocks (usually with explicit source message references) [13].

Using these references we can find the relations between messages. The conceptual task is illustrated in figure 2.1. We want to go from (a) to discovery of the relations as depicted in (b). Thus transforming a linear temporal message chain (based on message metadata) into a *directed acyclic graph* by exploiting semantic information contained within the messages. Note that there is still implicit temporal ordering in (b) namely that C_1 was posted after B_1 which is depicted by placing C_1 to the right of B_1. Time is thus represented in the graph by top-bottom and left-right ordering that both map onto an earlier-later scale.

Table 2.1: *Quoting example.*

Suresh:

I am having a Dell Inspiron laptop and it has a Broadcom 440x Ethernet card, i am not able to configure Ethernet connection... I am running Redhat 9.... Please help me out with this issue..

Mohinder:

Suresh wrote:
> I am having a Dell Inspiron laptop and it has a Broadcom 440x Ethernet
> card, i am not able to configure Ethernet connection...

Exactly what have you tried to do? What error message did you get when you tried? Are you using the correct network cable? Are you using static or DHCP? What does /sbin/ifconfig -a and /sbin/route -n show?

On the (b) side of figure 2.1 we see that the second post of author A, which is A_2, refers to B_1 and C_1. We call this *multi-quoting*. To our knowledge no attention has been paid to this in prior research even though it occurs more often as threads become longer. The phenomenon also appears in e-mail and newsgroups, but their limited one-on-one thread structure does not allow this to be expressed explicitly (although it could also be recovered there by using the same approach used for Internet fora).

Table 2.1 shows a reply to one message that quotes another. It can be observed that Mohinder quotes a part of Suresh's message and that the name of Suresh is also explicitly mentioned in the reply. This is quite common on message boards.

Schuth performed a study specifically aimed at finding the reply structure in comments on news articles [80] (he calls this a *reacts-on* relation). He found a variety of interesting features that combined lead to fairly good performance (recall of 0.39–0.66 and precision of 0.83–0.95). To allow spelling errors in author name citations he employed the Ratcliff/Obershelp similarity measure (using as similarity parameter $r = 0.85$). We adopt his features in this research with some adaptations for Internet fora. Remark that the domain of news articles is entirely devoid of any explicitly coded references which is a difference with fora[1].

To detect reply structure based on mentioning of author names we first collect the names of all the authors in the thread. The next step is finding all candidate matching words in a post. We do this first by looking for exact matches (similar to Schuth's word boundaries method). This leaves misspellings of author names for which we employ the Ratcliff/Obershelp algorithm (with $r = 0.85$). This algorithm is rather expensive[2] and there is a chance on false positives when

[1] We do not explicitly consider spam or off-topic posts in this research, although this is implicitly handled by mechanisms presented further in this chapter.

[2] Cubic in worst case, normally quadratic and linear in best case, see http://www.python.org/doc/2.3.5/lib/module-difflib.html

loosely matching. Therefore we need to select candidate words in each post that we believe might be a possible match. For this we consider words that are Part-of-Speech (PoS) tagged as proper nouns and words that follow an @ sign (Schuth's PoS-tagging and @-Trigger) and those that precede the word 'wrote' ('schreef' in Dutch). If an author name is mentioned in a post, that post is assumed to refer to the last post made by that author. When an author name is found in a post it is immediately also tagged semantically as being an author name (this is a dynamic portion of Named Entity Recognition for Person names, describe in detail in section 4.2.2).

Quoteblocks are more characteristic of fora than of news article comments. This is probably the reason that Schuth does not address it. Quoteblocks can be recognised either by '>' marks or HTML blockquote tags. Quotations are frequently shortened (as visible in table 2.1). What we want to find is the overlap of the quote with preceding messages in the thread. Quoteblocks can appear in several forms: with an explicit link to the source message (most common) and/or with explicit mentioning of the author name and without any references. Author names are covered by the approach described in the previous paragraph. The presence of explicit links provide a more detailed reference and thus always overrides mentioning of author names.

We use a simple algorithmic approach. Each line in a quoteblock is compared (from bottom to top) to the lines in preceding messages (lines in messages are also traversed bottom to top, messages chronologically from most recent to first post)[3]. This is also done using the Ratcliff/Obershelp algorithm (with $r = 0.75$, allowing for slightly more dissimilarity then for author names). Once a line in the quote matches a line in a preceding message, a reference is created to that message[4]. This process continues until either all lines in the quote are matched or there are no more messages to match against. When looking at previous posts, quote blocks in those posts are skipped.

What if there is an explicit link in a quote to the source message? In such a case we first compare the quoteblock against the mentioned source message. After this, we run the algorithmic approach as described in the previous paragraph as normal. If all these steps yield no reference we create a reference to the explicitly mentioned source message. This approach is computationally more expensive, but it does prevent creating erroneous links when the explicitly mentioned source message number is malformed (something we observed in our data). This fallback ensures that the reference is created even if the quote has been significantly altered in the referring post.

There is still another case to consider. We regularly observed that there was no quoting at all in replies to original texts. So, what assumptions should be made for messages without such references? We studied several threads and made some interesting observations:

- If there is an absence of references then the initiating author of the thread always responds to the most recently posted message. When these are messages of himself they are usually

[3] The reason for this processing order is largely to be able to efficiently handle quoting of lines of different source posts in the same quoteblock.

[4] The 'quote count' of the specific line in the source message is also increased. This is used later for line selection during summarization.

elaborations or reports of advancements made towards solving a problem (common in *Problem-Solution* threads).

- Messages of other authors are almost always follow-ups to the last message of the initiating author in the thread or responses to the message they themselves were last quoted in (or referred to by name). There are some exceptions to this heuristic like people responding to an older message of the initiating author (usually to give extra suggestions). However, these exceptions can not be automatically detected without actually interpreting the text. The heuristic appears to cover most of the cases quite well.

To discover the structure of threads we use a combination of clues (explicit mentioning of authornames and quote block recognition) and rules based on the above two observations. At the end of the discovery process, all messages in a thread (except the first post) have at least one reference to an other message.

2.2.2 Weighing

Now that we know which messages refer to which other messages we need to exploit this structure. Consider the structure of a completed discussion shown in figure 2.2. We can see several interesting characteristics here:

1. The message of D_1 is apparantly not so interesting (as it has been posted early in the thread and no one has replied to it).
2. Author A responds by quoting parts of the messages from both B and C in A_2.
3. Author B apparantly answers some of author A's follow-up questions in B_2 after which author A confirms his understanding.

The above interpretation is created entirely without looking at the messages themselves. Many threads can in fact be quite accurately analyzed like this, as there are patterns in their referential structure. The A-B-A at the bottom can for example be classified as a Question-Response-Thanks pattern [50].

However, more interesting is determining the relative importance of the messages from these perceived patterns. Kim, et al. found that the number of responses to a message is indicative of its importance in a discussion [48]. This is also expressed in the catalyst score f_c that Klaas uses, which is defined as [50]:

$$f_c\left(m\right) = \left|D_m\right| + \gamma \cdot \sum_{c \in D_m} f_c\left(c\right) \tag{2.1}$$

Where D_m is the set of direct descendants of a message m and γ is a discount factor (determined by experiment to be most optimal in the range 0.5 - 0.7). According to Klaas, this puts proper emphasis on messages in the beginning of the thread while not disregarding later contributions.

During our own observations of threads we noticed different phenomena. These hold for both *Problem-Solution* and *Statement-Discussion* type threads:

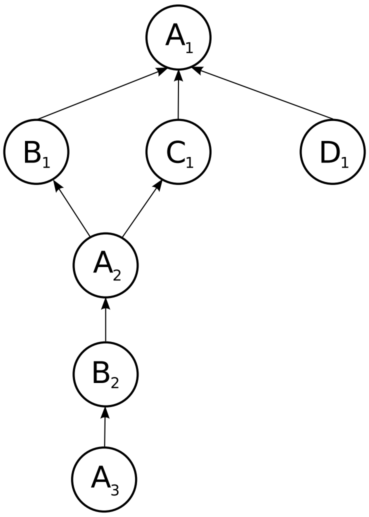


Figure 2.2: *Message thread tree structure example (A, B and C are author, numbers indicate the n^{th} post of an author).*

- ✦ The beginning of the thread (top of the tree) is especially important since it is devoted to either clarifying the problem or the statement (consistent with Klaas).
- ✦ In addition to this the end of the thread is similarly significant and populated more densely by *working* solutions and summaries of preceeding posts (contrast with Klaas).
- ✦ Messages by the thread initiating author (especially for *Problem-Solution* threads) are usually quite informative even if they are leaf nodes (final posts) (extension of Klaas).

This yields a different formula that incorporates the height of the tree. We name this the *Positional Message Relevance* (PMR):

$$f_{pmr}\left(m\right) = \left|P_m\right| + \left|C_m\right| + A_m + \left(H\left(m\right) - {}^1\!/\!_2 \cdot HT\right) \cdot 2 + \sum_{c \in C_m} f_{pmr}\left(c\right) \tag{2.2}$$

Where P_m is the set of parent messages that message m points to, C_m the number of child messages that point to m, A_m is 0 normally and -1 if the message is a leaf and the author is not the thread initiator, *H(m)* the height in the tree (relative to the first post) at which m is and *HT*, the height of the entire thread. The height of a message *H(m)* is always that of the longest path between the first post and message m. The purpose of the height factor is to slightly discount messages earlier in the thread and to give some extra weight to those at the end of the thread. This can cause PMR values to drop below 0 in which case they are fixed to zero. The extra multiplier 2 was determined by experiment to be useful especially for helping representation of messages near the end of the thread. We believe that it would be good to automatically tune this parameter to the thread, possibly by using the average width of the thread or a branching factor.

Table 2.2: *Message thread sample calculations.*

Catalyst ($\gamma = 0.5$)	*PMR*
A=4.75	A=25.5
B=1.75, C=1.75, D=0.0	B=12, C=12, D=0.0
A=1.5	A=10.5
B=1	B=7.0
A=0	A=3.5

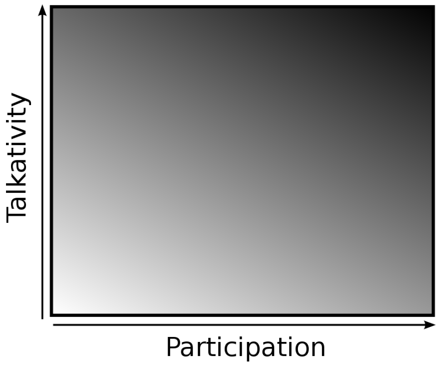


Figure 2.3: *Talkativity / participation weight (darker color means more weight).*

Besides the relative height factor (4th term in the formula) the sum of the other parts of the formula equate to 1 for normal leaf nodes. There are two exceptions: 1. the leaf message was posted by the thread initiator (in which case 1 is added); 2. a leaf node has multiple parent nodes (in which case each extra parent adds 1 to the weight of the node). This captures the fact that strongly referring leaf nodes are interesting for summarizing since they are usually already partial summaries by themselves.

Table 2.2 shows the values for both Klaas's catalyst score and the PMR. Values are shown at each height level corresponding to figure 2.2. Relatively the values are somewhat similar, PMR ranks the final message A_3 from the initiating author A higher than the dead-end posted by D. The effect of distance from the middle of the thread is not extremely visible here, but will become more pronounced as the height of the thread increases.

Besides the factors represented in the PMR there are another phenomena to consider. Especially for longer discussions (Statement-Discussion type) some authors post much more frequently than others. They have a higher degree of *participation*. Linked to that is that some authors contribute much more text (actual content). They have a higher *talkativity*. Authors with a high participation degree and talkativity usually have a much larger steering influence on the discussion. Hence it is quite important for their contributions to have additional weight. This idea, applied to spoken discussions, is also treated in the work of Rienks [75].

To cope with this we calculate the participation degree of each unique author in the thread. This is expressed as the proportion of messages of a specific author relative to the total number of messages a thread consists of:

$$f_p\left(a\right) = \frac{\sum_{m\epsilon M} author\left(a, m\right)}{\left|M\right|} \tag{2.3}$$

Where m is a specific message, M the set of all messages and the author function equates to 1 if author a is the author of message m.

Similarly, we also calculate the talkativity for each unique author in the thread which is the number of words an author contributed to the total number of words a thread consists of (excluding quotes):

$$f_t\left(a\right) = \frac{\sum_{m\epsilon M} length\left(m\right) \cdot author\left(a, m\right)}{\sum_{m\epsilon M} length\left(m\right)} \tag{2.4}$$

Where *length(m)* is the length in words of message m.

During assignment of summarization weights over messages we would like to give extra weight to authors that exhibit both a high talkativity and a high participation degree. The 'maximised' case (both f_p and f_t value 1) would only be true for a thread with one message and one author (which we do not consider in this research). The weight distribution is expressed in figure 2.3. The darker the color, the more weight should be assigned. This weight preference can be expressed in a simple weighted combined function:

$$f_{pt}\left(a\right) = {}^1\!/\!{}_2 \cdot f_p\left(a\right) + {}^1\!/\!{}_2 \cdot f_t\left(a\right) \tag{2.5}$$

We call this the Participation-Talkativity (PT) factor. Just like the PMR it is used later for message weighing.

The relevance of contributions of specific authors in a discussion is also modelled by Feng, et al. in their research on conversation focus [25]. However, they use speech acts combined with a graph based algorithm called HITS [51]. Specifically, they rate an author's value based on the positivity of the reactions on their posts. Such an approach works well on a manually annotated corpus, but to reliably find the polarity direction and type of post automatically is a difficult task with much domain specificity. Our approach is more simplistic, but scales better because it is completely automatic. The PMR is intended to take care of determining the less relevant information at the message level rather than at the author level.

Both Klaas [50] and Farell [24] demand that some part of each message should be included in a summary. We take a different approach and allow no content of a message to be included if it turns out to be irrelevant.

2.3 Readability

The following two sections provide details on ways to identify messages that stand out in a way that makes them candidate for filtering. The result is that their content is not included in a summary. One could say that with this we try to perform very basic automatic moderation. There are many other criteria that moderators use that we have not modeled here. Hence, this is not an attempt to provide full automatic moderation.

2.3.1 Indices

Readability is a very important aspect of texts. There exists a wide variety of metrics for this today, numbering well over two hundred. Best known are probably those made by Flesch[5]. These formulas are frequently based on the number of words per sentence and the number of syllables per word. The latter is difficult to automatically determine using machines (syllable dictionaries are necessary for this). Hence, there exist several formulas that use the number of characters in words as opposed to the number of syllables [19].

Readability scores are usually used for long stretches of running texts, like manuals and books. We believe their direct applicability to other types of texts, such as newsgroup messages can be disputed. These messages are typically much shorter. Nevertheless, researchers have attempted this and found that most such messages exhibit a reading ease somewhere between *fairly easy* to *normal* [77].

For this research we are not so interested in exact readability scores, but more in relative readability scores of messages in the same thread. We assume that messages deviating from the average readability in the thread are of less importance. This assumption stems from the fact that texts are usually written for a particular audience and that authors of forum messages also adjust their communication to their audience (in this case: other posters in the thread).

Because of the difficulty (and language dependence) when using syllables in readability scoring we resort to character counts instead. Conceptually, our approach is close to that of Coleman and Liau [15], with the exception that no single combined score is created. Table 2.3 shows the formulas. Punctuation marks are not counted as words, nor are certain types of semantically recognisable characters sequences that are not technically words and can skew the statistics (notably emoticons and addresses like URL's as recognised by the semantic tagger, see section 4.2).

It remains somewhat dubious to compare these statistic between posts because of length differences. Nevertheless, we will assume that the Average Word Length (AWL) and Average Sentence Length (ASL) within a thread are normally distributed, and that deviations for these measures larger than a z value of three in either positive or negative direction are indicative of less important messages. So, posts of authors that are unusually terse or that write very lengthy passages (decreased readability) are less likely to be read with respect to what is 'normal' in the thread (as this varies between different thread types).

[5] The Flesch reading ease of this thesis is 43.94, close to the Wall Street Journal and readable for college students.

Table 2.3: *Readability statistics (**m** is a message).*

Average Word Length	$f_{awl}\left(m\right) = \frac{\sum_{i=1}^{n} length(w_i)}{n}$	n is number of words in post, w_i is a word
Average Sentence Length	$f_{asl}\left(m\right) = \frac{\sum_{i=1}^{n} length(s_i)}{n}$	n is number of sentences in post, s_i is a sentence

We conducted a small experiment to find if there is any correlation between the PMR of a message in a thread and the AWL and ASL. Based on 147 samples we found that there is a small negative correlation (~-0.10) between PMR and ASL which suggests that messages with shorter sentences are preferred. The correlation between AWL and PMR is too small to be of any significance. However, there is a negative correlation (~-0.15) between AWL and ASL, suggesting that shorter sentences also contain shorter words.

2.3.2 Formatting Characteristics

The quality of a post is hard to define. In fact properly rating post quality would require exhaustive semantic knowledge. Intuitively, certain formatting characteristics of a message can be indicative of poor quality. Examples are missing distinguishing capitalisation, missing punctuation marks, repeated exclamation marks in a sentence and sentences that consist of all capitals. There are also several more difficult semantic characteristics requiring extra knowledge, such as the number of spelling errors and the amount of foul language used.

Weimer, et al. investigated the effectivity of several types of features for a good / bad classification task of forum posts. They based their research on a corpus of human rated posts. The surface features used here are similar to the ones they used. Some of their other features are represented in our research in other ways[6] like readability (section 2.3.1) and thread structure (sections 2.2.1 and 2.2.2) [93].

We will focus on four easily calculatable surface features to assign a score to a message[7]. Formulas and examples are shown in table 2.4.

A more elaborate definition of *No Capitalisation*: If the first word in a sentence consists of either all lowercase or all uppercase characters (excluding one character words) we consider this a lack of distinguishing (start)capitalisation. Under this definition ‘PEPSICO’, ‘pepsico’ and ’pepsiCo’ are wrong and ’Pepsico’ and ’PepsiCo’ are considered right. Starting a sentence with a digit is considered valid and not counted as missing capitalisation.

[6] Specifically ‘quote fraction’ is represented via positional message relevance and readability could be considered a lexical feature.

[7] The suggestion of using surface features was actually initially inspired by comments made by Steven de Jong, a moderator of the NRC weblog. He independently observed the same phenomena as Weimer: there is a correlation between certain surface features and the quality of posts. In contrast, his observations are based on moderation experience whereas Weimer’s are based on a corpus.

Table 2.4: *Formatting characteristic formulas ($\boldsymbol{n}$ is the number of sentences or words, $\boldsymbol{s_i}$ is a sentence, $\boldsymbol{w_i}$ is a word, $\boldsymbol{m}$ is a message).*

No Capitalisation	$f_{ncs}\left(s\right) = \frac{\sum_{i=1}^{n} capitalised(s_i)}{n}$	[*m*]issing capitalisation.
All Capitalised	$f_{acw}\left(w\right) = \frac{\sum_{i=2}^{n} capitalised(w_i)}{n}$	I [*LIKE SHOUTING*]!
No Punctuation	$f_{nps}\left(s\right) = \frac{\sum_{i=1}^{n} nopunct(s_i)}{n}$	missing punctuation[]
Repeated Exclamation	$f_{res}\left(s\right) = \frac{\sum_{i=1}^{n} repeated(!,s_i)}{n}$	now this will help[*!!!*]

$$f_{mfs}(m) = 1 - ({}^{1}/_{4} \cdot f_{ncs} + {}^{1}/_{4} \cdot f_{acw} + {}^{1}/_{2} \cdot f_{nps} + {}^{1}/_{2} \cdot f_{res})$$

For *All Capitalised* only words after index two in a sentence are considered and only words that are longer than one character and contain only alphabetic characters are counted. For *Repeated Exclamation* the exclamation mark need not necessarily appear at the end, but may be followed by other punctuation marks. In such cases the sentence will still be counted as having the repeated exclamation property.

A grand global score known as the *Message Formatting Score* (MFS) is calculated based on four individual characteristics. An MFS of 1.0 indicates a well formatted messages whereas an MFS of 0.0 indicates a very poorly formatted one. Note that one might expect the four factors to be weighted equally (${}^{1}/_{4}$ each), this is only true for *No Capitalisation* and *All Capitalised* however. These two can co-exist in one sentence, whereas *No Punctuation* and *Repeated Exclamation* can not. Therefore these latter two factors share their score space, both weighing in at ${}^{1}/_{2}$ instead of ${}^{1}/_{4}$.

An example of applying the formulas is shown in table 2.5. First for a badly formatted message and second for a well formatted one. This approach is simply intended to give a rough estimate of how well formatted a message is. The *All Capitalised* characteristic may inadvertedly penalise abbreviations or names (such as *ACW*). Ideally we would filter these cases out with a list of common all-capital words. However, for the sake of simplicity and the relatively low score impact, we ignore this.

The MFS expresses the correlation between how well a message is formatted and the quality of a message. We chose four factors, based on prior research and expert knowledge cited earlier, but this could easily be extended with other formatting features in the future.

Table 2.5: *Message excerpt MFS examples (lines displayed between brackets,* italic *words counted for f_{acw}).*

DANGER... DANGER... *DANGER* ! ! ! ... ! ! ! ... *DEPORT THEM NOW* ! ! ! ... *BEFORE* IT'S *TOO LATE* ! ! !
$f_{mfs} = 1 - (1/4 \cdot 4/4 + 1/4 \cdot 7/7 + 1/2 \cdot 0/4 + 1/2 \cdot 4/4) = 0.0$
You have your opinion, and I have mine . Why not leave it at that, as you always want people to do with your posts ?
$f_{mfs} = 1 - (1/4 \cdot 0 + 1/4 \cdot 0 + 1/2 \cdot 0 + 1/2 \cdot 0) = 1.0$

Chapter 3

Foundation Technologies

> 'A successful man is one who can lay a firm foundation with the bricks others have thrown at him.'
>
> *(David Brinkley)*

There are many elemental tasks in Natural Language Processing that are necessary for all kinds of higher-level applications like the one developed as part of this research. Many of these are (and have been) well researched providing a relatively solid basis. These technologies that are not a core part of this research, but that form the foundation for it, are discussed briefly in this chapter.

3.1 Tokenising

One of the first things that need to be done with a text is splitting it into meaningful units. To a computer a text is just bytes without any meaning beyond the character level. Recognisable units are paragraphs, lines and words. The field that deals with these kinds of issues is called *text segmentation*, although the task is frequently referred to as *tokenisation*.

Paragraph boundaries are fairly easy to recognise (by either the presence of an indenting tab or a blank line). However, line boundaries pose a challenge. The most basic approach is splitting sentences on the dot (.), exclamation (!) and question mark (?). But consider the following examples (sentence boundaries denoted by square brackets):

- [Mr.] [Brooks came into my office today!] [!] [!]
- [It is exactly 3.] [85 meters long.]

Applying those simple splitting rules does not work very well here. For the first sentence, this would yield a one-word line with only 'Mr.' followed by three more lines because of the repetition

Table 3.1: *Word properties.*

Property	*Examples*
Quoted	‘word’, “word”
Emphasised	_word_, *word*,___word___, **word**, ...
Bracketed	(word), [word], <word>, {word}
Uppercase	WORD

of exclamation marks. While we would all consider this to be one coherent sentence. Something similar is happening in the second sentence for the dot used in the numeric expression (3.85).

Hence, the task of tokenising is not as easy as it seems. For this research the Punkt sentence tokeniser [49] was employed for properly finding the lines a paragraph consists of. A complete discussion of Punkt is beyond the scope of this document. We mention shortly that its tokenisation employs a variety of heuristics to detect sentence boundaries like ortographic hints and more importantly collocations. It is also language independent (for Western European languages) which is an additional advantage. Performance for finding sentence boundaries, tested on a newspaper corpus, is >99% for Dutch (and most other European languages).

Word-level tokenisation is a bit more straightforward. The basic approach used is to split a sentence on spaces. This however leaves problems with comma’s and quotation marks. Consider the following example:

- “I know a man, who ‘sits’ behind a (large) machine (all day)”.

It is obvious that just splitting this on whitespace will yield some undesirable word units such as ’sits’ with the quotation marks attached to it, man with a comma attached, large between parenthesis, etc. Several rules were employed to cope with this:

- If a word ends with a comma, the comma is split off and treated as a separate ‘word’.
- Quotes and parenthesis around a word are removed and set as properties of the word (this is to prevent having to strip off these characters at each subsequent processing level, while retaining the ability to restore them in the final output).
- Quotes and parenthesis around a sequence of words are split off and treated as separate ‘word’.
- All words are lowercased and the original casing is set as a property of the word. The entire original word is also stored primarily for output purposes later on.

Hence we essentially work (throughout most of the system) with lowercased words with other surrounding typographic symbols removed. These can be accessed and restored at any desired time. A complete list of these word properties is shown in table 3.1.

3.2 Part of Speech (Syntactic) Tagging

In Part-of-Speech (PoS) tagging each word is labeled with its grammatical function, like *noun* or *verb*. This can be done manually by creating rules that derive the PoS from the lexical structure of a word and possibly the surrounding words. However, this approach is very labour intensive. Nowadays, the most common approach to PoS tagging is using a machine learning model that learns the tagging patterns from a large manually annotated corpus [43].

Many machine learning models have been unleashed on the PoS tagging task. However two models are fairly tried and tested: Transformation Based Learning (TBL), also known as Brill Tagging, and the Hidden Markov Model (HMM). For this research HMMs were chosen, mostly based on the fact that TBL taggers tend to take a somewhat longer time to train, especially on large corpora, while having no performance advantage over HMM's [85].

Handling of unknown words is an important aspect of PoS tagging as they can greatly impair performance. This handling can be done by using heuristics such as changing the case of a word, taking compounds (less useful for English) and morphological analysis [61].

Two PoS tagged corpora have been used. For Dutch the Spoken Dutch Corpus (CGN) and for US English the well-known Brown corpus. These corpora use different tagsets with some very fine-grained distinctions that are not really useful for the task at hand [22, 27]. To solve this a unified tagging scheme was developed which consists of only 26 tags. The 179 possible Brown tags and 72 CGN tags (already a reduction of the larger 320 tagset) were projected onto this reduced tagset. The unified tagset can be found in Appendix A.

Developing a PoS tagger is not the goal of this research. We use PoS tagging as a foundation on which to build other functionality. We experimented with the bigram HMM tagger included in the Natural Language Toolkit (NLTK) [7]. Lack of unknown word handling and long loading times eventually led us to the decision to use the external Hammer Tagger Toolkit [84]. This is the successor to the earlier developed TaggerTool at the University of Twente which produces taggers with a similar error rate but has better speedwise performance.

The biggest problem for taggers remains handling unknown words. Hammer taggers have a number of built-in strategies for handling such words. A detailed discussion of these is beyond the scope of this document. We briefly mention here that a Hammer tagger first tries to decompound a word. When failing it performs morphological analysis. Finally the tagger falls back to simply assigning the most likely tag based on frequency information.

Despite this rich unknown word handling there are still some problematic cases in our data. Messages are often filled with numbers, addresses, and the likes. To cope with this we extended the Hammer Tagger Toolkit with support for unknown word handling based on regular expressions. This module is located just one stage above the simple frequency fallback. It handles many 'words' that appear in on-line communication. These are not easy to include in training data due to their variation. Yet they exhibit lexical continuity that can easily be captured with regular expressions. Concrete examples are URI's[1], local file paths, e-mail addresses and

[1] This is the notation underlying the better known URL. This includes *http://* but extends to other protocols as well (e.g. *smb://* and *file://*).

number sequences. Fixed PoS tags are assigned to each of these to aid the tagging process.

PoS tagging is an important base technology used in many of the higher levels of analysis. We conducted an evaluation with our unified tagset. Results can be found in Appendix A. We mention here that the tagging accuracy is about 97.5%. Whilst not perfect, this is in-line with present-day state-of-the-art tagging performance [61].

3.3 Partial Parsing

When PoS tags for all words are known, the next step is to recognise groups of words (also called constituents or phrases). Consider the following sentence:

The	car	drives	over	the	road	towards	the	little	pond
D	N-C-1	V-C	R	D	N-C-1	R	D	A	N-C-1

The underlined parts of the sentence are actually word groups and in this case specifically noun phrases (NPs). There are many types of phrases, mostly named after the most meaning bearing Part-of-Speech in the phrase. In the example above '*drives* over the road' is a verb phrase (VP) with 'drives' as the central word.

There are different approaches towards parsing. Traditionally sentences are fully parsed which means that they are broken down in a (fairly deep) parse tree. In the sentence above 'the car' and 'the road' are noun phrases, although they are actually part of the encompassing 'drives' verb phrase.

Another approach is dependency parsing. In this form of parsing a dependency tree is build which allows distinguishing of grammatic sentence level relations. Enabling finding phenomena such as: 'The car' is the subject of 'drives' while 'the little pond' is the object. There exists a dependency parser for Dutch called Alpino [11, 43].

The question that we must ask ourselves is what we actually need as opposed to what is technologically possible. For tasks like anaphora resolution (see next section) it is sufficient to be able to recognise noun phrases. Computational cost is also a concern. Alpino for example appears to be quite slow even on modern hardware.

Instead of doing full parses, we take a different lightweight approach in this research known as partial parsing (sometimes called chunking). In this approach no tree is created, but sequences of tags are recognised as phrase types. Chunking can be seen as yielding a flattened parse tree. Even though there can be multiple levels of phrase types, such as a noun phrase consisting of two other noun phrases, we consider only those at the lowest level in the tree. Partial parsing is a lot more pragmatic and faster than full parsing, mainly because it remains relatively unaffected by the other parts of the sentence. Thus a full parse of a sentence is not necessary if finding only the noun phrases is what matters.

We use the built-in capabilities of the Natural Language Toolkit[2] for partial parsing. This works by specifying regular expressions[3] at the PoS tag level. An overview of applied rules and an evaluation can found in Appendix B. The performance is around 87% recall and 95% precision.

3.4 Anaphora Resolution

An anaphor is a specific type of reference pointing back to an antecedent in a text. The concept is frequently confused with coreference which is a related, but different, concept. The definition of an anaphor is that it *depends* on its antecedent for correct interpretation [18].

When making extractive summaries, sentences are 'cherry picked' out of a larger piece of text (see chapter 6). This leaves us with the problem of anaphora in those selected sentences referring to previous sentences that are not included in the summary. This phenomenon is known as *dangling anaphora*.

We are interested in *indicative* anaphoric resolution. This means instead of replacing an anaphor we augment it with a likely antecedent. This 'suspected' antecedent is always placed behind the anaphoric expression in parentheses:

> I bought a videocard today. The *card [videocard]* is noiseless.

This still leaves the possibility for a user to determine the anaphoric resolution is wrong (the right antecedent could be manually looked up in the original full source message).

We focus on three kinds of anaphoric relations which we believe to be useful in summaries:

1. Pronouns: $The\ car_\alpha$ broke down. We had to move it_α by hand.
2. Generalisers:
   a) I bought a $videocard_\alpha$ today. $The\ card_\alpha$ is noiseless.
   b) The $huge\ mansion_\alpha$ stood on the edge of a hill. $The\ mansion_\alpha$ was once inhabited by ....
3. Acronyms: The $Free\ Software\ Foundation_\alpha$ made a plea. Proponents of the FSF_α say ....

3.4.1 Traditional Approaches

Anaphora resolution is usually performed by looking at previous sentences[4] for candidate noun phrases. These candidates are then scored according to several features. The weights of these

[2] Note that the Hammer tagger (mentioned in section 3.2) also has chunking capabilities based on machine learning. However the functionality is not very well tested and evaluated. Hence we decided not to use it and opt for this simpler approach instead.

[3] The expressions that we used are based on a grammar for Dutch by Rieks op den Akker

[4] Note that we are not concerning ourselves with anaphora resolution within the same sentence. In fact if we do find that an anaphor can be resolved within the same sentence it is ignored.

features are usually referred to as *salience* in line with the terminology used by Lappin and Leass [55].

Hoste and Daelemans [36] have attempted to use a machine learning approach to learning coreferential relations (which in their view includes anaphora) specifically for Dutch. They used a variety of features for finding such relations. Nevertheless the results are somewhat disappointing indicating the challenge of the task (recall ~67%, precision ~42%). We adapt some of their methods for our anaphora resolution as explained below. Their 2007 paper, co-authored with van den Bosch [37], shows the importance of partial matching among several other features. Follow-up research by Hendrickx [33] uses WordNet and unsupervised clustering, but only marginally improves their prior results. In a practical sense, such a small improvement does not seem to justify the implementation and run-time costs that are incurred.

Resolving pronominal anaphora was recognised by Zechner to be important in dialogue summaries, even though he does not actually perform it in his own research [98].

Researchers have used heuristics and contextual information in various ways to aid anaphora resolution. We restrict ourselves to several basic ones:

- Pronoun resolution is restricted to occuring in the preceding three sentences [66].
- Partial matching between the head of a noun phrase and (parts of) candidate antecedents [36, 37].
- Building acronyms from noun phrases (called the 'alias' feature by Hoste [36]).
- Weighing definite descriptions stronger than indefinite ones (*the boy* versus *a boy*) [2].
- Noun phrases preceded by a preposition are weighted lower (*on the bike* versus *the bike*) [2].
- Number agreement between the anaphor and its antecedent [43].
- Noun phrases between parentheses are weighted lower (they are less likely to belong to the main stretch of text).

A notable omission is gender agreement. Most referred research employs a separate list of words with gender information for this purpose. Although it would not be difficult to add support for this at a later time, the current incarnation of the system does not use gender agreement yet. CELEX could be used for this. Gender however appears to be less helpful for Dutch than it is for English [36, 37].

The specific pronouns we look for are shown in the top part of table 3.2. Handling is not exhaustive as pronouns can appear in many other forms, for example reciprocal pronouns and numbers. Nevertheless, this provides a basis.

Apart from pronouns we also handle generalisers. These are found in two ways. First by matching head nouns[7] of noun phrases to other head nouns and seeing if they form a suffix (as in *card* being a suffix of *videocard*) also known as partial matching (not performed when words

[7] The head noun is taken to be the rightmost noun in a noun phrase. This is not always correct, but a reasonable approximation.

Table 3.2: *Resolved pronouns.*

Form (person)	*English*	*Dutch*
Personal (3rd)	he, she, him, her, they	hij, zij, hem, haar, (hen, hun, zij, ze)
Possessive (3rd)	his, her, their	zijn, haar, hun
Reflexive (3rd)	himself, herself, themselves	zich, zichzelf[5]
Demonstrative	it, this, these, that, those	het, deze, dit, dat, die[6]
Personal (1st)	I, me	ik, mij
Personal (2nd)	you	jij, u, gij
Possessive (1st)	mijn	my
Possessive (2nd)	your	jouw, je, uw

[5] These can inadvertedly apply to 2nd person as well.

[6] The latter two can appear in relative context too.

contain explicit wordbreaks). The second is by finding exact head noun matches preceded by more other words, such as adjectives or nouns, with respect to the anaphoric noun phrase (the phrases thus share the same head). This is not done if the anaphoric phrase contains numbers or numeric expressions (which indicate more detail). The idea behind this handling of generalising anaphors is that both longer head nouns and nouns augmented with adjectives (antecedents) provide more specific information than the anaphor itself. Including this information can aid in understanding.

Acronym handling quite simply builds a set of possible acronyms from the first letters of all nouns in each noun-phrase and also variations that use other words between the first and last nouns in the noun phrase. For example, the noun phrase: ‘Department of Heuristics And Research on Material Applications’ would yield the following acronyms: DHRMA, DOHAROMA and DHARMA. With for alle these additional variants containing dots between the letters and optionally a dot at the end (e.g. D.H.R.M.A and D.H.R.M.A.). Head nouns of other noun phrases are matched against these acronyms. If a match is found an anaphoric relation is created between the two noun phrases.

3.4.2 Exploiting Specific Thread Characteristics

Which lines do we consider for all the above resolutions? For pronoun resolution we only look at the three preceding sentences [66], even across post boundaries. The first line of a post is allowed to refer pronominally to the last three lines of all referred-to posts (found by thread structure discovery, see section 2.2.1) and the chronologically previous post(s). For the second and third lines the number of allowed referred-to lines of previous posts is reduced accordingly. For those lines the preceding lines within the same post have a higher selection priority than those in any of the previous posts.

For the other types of resolution twelve preceding sentences[8] in a thread are considered in the

[8] This was determined by experiment to be a relatively good trade-off allowing enough matching in context and

Table 3.3: *Noun phrase sentence weight distribution (note that the* ordering *is what matters,* not *the exact values).*

Noun Phrases	*Weights*
1	1
2	$3/4$, $1/4$
3	$4/8$, $1/8$, $3/8$
4	$8/16$, $1/16$, $2/16$, $5/16$

referred-to and chronological order. This is similar to pronoun resolution but with a different restriction on the number of backward lines. Of course lines closer to the expression are always weighted stronger than those further away.

Within a line, the weights of a noun phrase depends on the position of the phrase in the line. We have attempted to model what is in focus by using a simple heuristic. The first noun phrase is always preferred, thereafter the last noun phrase in the sentence is preferred. The remaining noun phrases are preferred in right to left order. See table 3.3 for an impression. Of course, the preferences expressed by the heuristics mentioned in the previous section are also kept into account.

There is yet an other aspect to this. Especially since we are dealing with dialogue we would also like to resolve pronominal references in the first (I, me) and second (you) forms (shown as the second part of table 3.2)[9]. As it is very difficult to determine who contributed what to the discussion in a summarized thread otherwise. Due to the structure of threads and the available meta information this task can be somewhat simplified. The following heuristics are used:

- For self reference (first person), the anaphor is resolved to the name of the poster.
- For second person reference, the anaphor is resolved to the author of the message the current message is replying to (if any) provided that there is only one such reference[10].

We were not able to find a good Dutch on-line (and free) corpus for anaphora resolution. The KNACK-2002 corpus used by Hoste is close, but it does not appear to be available on-line [36]. As such, it is hard to say anything about the performance of the anaphora resolution algorithm described. Anaphora resolution is considered an extra that aids in the interpretability of a summary. Hence, it is not a central part of this research.

disallowing matching too far outside the current dialogue context.

[9] These specific forms are usually not referred to as anaphora

[10] This could be made more intelligent by taking into account locality of reference with respect to especially quote blocks or name citations.

3.5 Rhethorical Structure Theory

Rhetorical Structure Theory (RST) identifies relationships between sentences (and sentence parts). The core of a sentence is called the nucleus. The nuclues has a relation with one or more satellite sentences. These last two sentences are satellites of the first sentence in this paragraph with an elaboration relation.

RST was initially conceived during the eighties by Mann and Thompson and is still popular today. It has been used in the context of summarization by Marcu. He brought into practice earlier ideas regarding RST concerning the usage of nuclei within a text as summary [60, 62].

We want to learn if messages in threads exhibit characteristics that could be exploited by RST. This is the prime reason for investigating this technique. We are especially interested in weeding out certain types of satellite sentences that do not add much information to a summary.

RST is based on monologues. It has no special handling or relations for dialogues. There have been some attempts to extend RST with dialogue handling, but there has been very little scientific research into the effectiveness of these approaches [86, 5].

Human annotation of RST relations is rather labour intensive. Work has been done to be able to automatically recognise these relations. Such an approach would be valuable for us, since creating a large annotated corpus is unfeasible. Notable are the beforementioned work of Marcu for English and Timmerman for Dutch [62, 88].

For English, automated systems that detect intra-sential RST relations have been developed offering good performance, but these are less interesting for us because of their language specificity [81]. These systems however have been used notably by Kim, et al. [48] for discussion fora. They found that *attribution* (who owns the text), *elaboration* (easing understanding) and *enablement* (increase potential ability of the reader) are the dominant intra-sential relations.

RST distinguishes two broad classes of relations between sentences and sentence parts: *paratactic* being indicative of sentences with equal importance and *hypotactic* in which the nucleus is considered essential to the writer's purpose whereas the satellite(s) is (are) not. We are interested only in the latter relationship since this would enable omitting less important information.

Timmerman focuses on only a few RST relations which he claims cover nearly 90% of all relations that he found in his corpus. The corpus he uses is specific to the medical domain. The relations he uses are [88]:

- *Elaboration*
  His name is Kevin Johnson. He works on the Kahana.
- *(Non)Volitional Cause*
  Walt fell from the raft. He was pulled by one of the others.
- *(Non)Volitional Result*
  The captain commanded 'engage'. The ship went into warp speed.
- *Concession*
  Darwin as a geologist. Although he tends to be viewed now as a biologist.

Timmerman detects these using discourse markers relying mainly on pronouns and adverbs. An approach that is appealing to us. Although we are not interested in finding the types of relations. We simply want to discount satellite sentences as to decrease their possibility of being selected for a summary. Timmerman reports several interesting findings like the fact that in nearly 80% of all relations the satellite sentence appears after the nucleus and that about 60% of the relations appear between directly adjacent sentences. However, he makes a number of assumptions such as coherency of the text, information ordering and last but not least a restriction to the medical domain. We should also not forget his data is structured as a kind of encyclopedia. The main problem is the assumptions he makes about the nature of his data might or might not hold for the type of data under our consideration.

After some deliberation, we eventually chose not to include an RST component in our system. Using Timmerman's existing implementation was considered, but its reliance on the rather slow Alpino dependency parser makes it unsuitable. Far more significant in this decision was RST's lack of dialogue handling, the trickyness involved in automatic domain-independent recognition of the relations and the limited applicability to very short texts[11].

[11] Even though longer messages do appear in fora. They are less common than short message exchanges. This severely limits the applicability of RST.

Chapter 4

QA Related Technologies

> 'The Star Trek computer doesn't seem that interesting. They ask it random questions, it thinks for a while. I think we can do better than that.'
>
> *(Larry Page)*

MANY technologies used for question-answering are aimed at finding factoids or longer spans of text based on a user query formulated as a question. In contrast we are interested in finding questions and answers in existing spans of texts. This chapter shows the question-answering techniques we use, and also how we use them in different way geared towards our application.

4.1 Sentence Type Detection

Detecting the type of a sentence is useful for a variety of task. First of all for filtering out (for a summary) less relevant sentences [23] and secondly for finding links between sentences across messages. We detect only a few sentence types.

4.1.1 Openers, Closers and References

Openers are sometimes places at the beginning of a message (we noticed usage of this especially in first posts) and are simply greetings. Similarly closers are usually thankings (or even just mention the author name). Some examples (English translations shown in *italic*):

- Beste ....,
  Dear ....,

- ✦ Alvast bedankt!
  Thanks in advance!

These types of sentences are detected by a combination of cue-words and sentence length information and marked as line type ‘Opener’ or ‘Closer’.

Lines consisting of only one word that has been semantically tagged as an Address of somesort (see section 4.2) are marked as ‘Reference’.

4.1.2 Lists

Lists are summaries of points made by an author. They usually start off with a sentence ending in a colon (:) which is followed by a number of sentences that use specific characters (such as - or *) or letters/numbers (1, 2, a, b, etc.).

An example:

1. Swan.
2. Orchid.
3. Temple.

The items themselves (three sentences above) are marked as ‘ListItem’ whereas the initiating sentence (‘An example:’) is marked as ‘ListStarter’.

4.1.3 Questions

Detection of questions is a more difficult task than one might expect. It might seem obvious sentences ending with a ‘?’ are questions. However, they need not always be for example in the case of rhetorical or suggestive questions. Question marks can also be ommitted entirely, while a sentence is still a question. We recognise these tricky issues, but restrict ourselves solely to sentences ending in a question mark in this research and questions that are formulated in an interrogative form [63].

Question sentences are marked when found and if possible an answer type is also indicated. We detect possible answer types based on the first word present in a sentence optionally combined with at least one clue word which can appear at a random location in the sentence. A brief inventory of supported questions is shown in table 4.1. Also shown are the entity types that can be used to answer each question. The question typology is based on that used by Webclopedia, but has been kept much smaller [39].

The word ‘what’ has been ommitted from the table, since it can appear in many contexts:

- ✦ What is the name of the inventor or C++? [Who]
- ✦ What was the location James went to? [Where]

Table 4.1: *Question types.*

English	*Dutch*	*Response (Entity Type)*
Who?	Wie?	Person, Organisation
Whose?	Wiens? Van wie?	”
Where?	Waar?	Location
When?	Wanneer?	Quantity-Time, DateTime
How long?	Hoelang?	Quantity
How many / much?	Hoeveel?	”
How often?	Hoevaak?	”
How far?	Hoe ver?	”
How big / large?	Hoe groot?	”

Hence, ‘what’ is allowed only when appearing with specific cue-words, like location and name.

Other ommission are ‘why’ that indicates a cause / effect relationship, ‘how’ (on its own) that requires an elaborate explanation. And ‘which’ that requires selection from a set of options.

As mentioned before, our question typology is rather small. It could be extended in the future, but we wanted to show the basic usefulness of the approach here.

4.2 Semantic Tagging

Semantic tagging concerns itself with finding recognisable constituents in a text for a higher-level purpose. The type of semantic tags necessary to be found is linked strongely to the usage goal. So, how do we determine what type of tags we need?

There exist large hierarchies of named entity types[1]. However, a high amount of types can be quite complex to detect and can also lead to a lot of ‘noise’[2]. We will restrict ourself to only a small set of types, namely those that are typically used as answers to questions (see section 4.1).

Now that we know what entities to detect (last column of table 4.1), we still need to come up with a way to detect them.

4.2.1 General Semantic Units

There are many highly structured expressions that can be recognised automatically quite easily. Think of URL’s, e-mail addresses, dates, times and quantities. We use regular expressions to find these comparing only against the surface form of words. Our approach was inspired by that of Lam [53], although our approach can recognise many more specific types.

[1] See for example `http://www.yooname.com`

[2] A similar problem to a smaller versus bigger PoS tag set, as explained in section 3.2

The largest group of regular expressions handles all kinds of numeric expressions, also ones that span multiple words. Some examples:

- Quantity-Mass: 10kg, ten kilogram
- Quantity-Money: USD 30
- Quantity-Electric: fortyfive Wb

It also deals with various other expressions that commonly appear in forum posts, like:

- Address-Email: j.j.rousseau@nospam.net
- Address-File: C:\NCK\NCK.exe
- Address-URI: http://hmi.ewi.utwente.nl, imap://imap.find815.com

Emoticons, such as :), :(, etc. are also found and tagged accordingly. We distinguish between emoticons that express a state (happy, sad, undecided) and those that express an action (laughing, crying).

The regular expressions are specified in a separate XML file. It uses a number of basic expressions (called blocks) for things like SI units and higher-level rules expressed in these blocks. These are expanded into (very long) regular expressions that are sequentially matched at the sentence level in messages. Currently, Dutch and English are supported. We have attempted to provide reasonable coverage, but of course it is quite difficult to provide regular expressions that cover all instances of a given concept. However, the usage of a configuration XML file makes it very easy to add new expressions.

These tags are used at various other places in the system. A full list of semantic tags (including the mentioned semantic units) can be found in Appendix C.

4.2.2 Named Entity Recognition

Named Entity Recognition (NER), sometimes called Name Identification and Classification, is a task within the field of Information Extraction. It is concerned with finding *entities* (e.g. Australia, Kate) and their *types* in texts (e.g. country, person). This is usually defined as a two-step process as some entities may have a different type depending on the context, as in:

> *Den Haag* beweert dat de stabiliteit van dit land ernstig in gevaar komt.
> *The Hague* claims that the stability of this country is in serious danger.

It is obvious that *The Hague* here does not refer to a location, but an Organisation (namely the Dutch government).

There are several approaches to the named entity recognition task. The most traditional is the use of look-up lists (gazetteers). Hand-coded rule-based (pattern) approaches have also been used, but require linguistic knowledge. More recent approaches focus on finding important indicative features by applying machine learning techniques. This introduces the additional

Table 4.2: *Gazetteer list sizes (approximates).*

List	*~Size*
Firstnames, Occupations	12000
Lastnames	13750
Organisations	2600
Locations	1500

problem of entity boundary detection. For this orthographic features (especially word capitalisation) are important. Morphological features, such as the prefixes and suffixes, seem to be the most important for actually recognising entities [8, 30].

There is evidence that the look-up list based approach is actually also the best performing approach [46, 64, 83]. Therefore, we have chosen to take the power of this approach augmented with some very simple heuristics discussed below. The approach is somewhat similar to that used by De Meulder, et al. [64] without usage of context rules.

We primarily rely on lists to find Person names, Organisation names and Locations. The size of our lists are displayed in table 4.2. These lists were manually constructed by using various websites amongst which is Wikipedia. The usefulness of using Wikipedia for entity recognition, albeit in an automated fashion, has been shown in prior research [46]. Although the lists are oriented towards Dutch, they also contain many common English ones (also in their English spelling). This is due to the fact that even on Dutch fora the English variants are also used.

For both Locations and Organisations, there has to be a literal match to the names contained in our lists. The first word *must* also be a noun. In case of multi-word matches the case of the words following the first word in the match are allowed to differ in case from what is defined in the list, provided they have a length of more than six characters[3]. Matches to occupations are lowercase by default.

For names we allow only firstnames, only lastnames and combinations of first and lastnames to appear. For firstnames a case mismatch is allowed provided the word is more than six characters long[4]. For lastnames (and combinations) the same length rule for case-mismatching applies as for Organisations and locations. Excluding case variations this alone allows for well over $25750 + 12000^{13750}$ unique matches. During thread structure discovery (see section 2.2.1) names of authors in the thread are semantically tagged as well (that process uses more relaxed name matching rules).

If there are multiple possible tags for a given word (sequence) we do not disambiguate. Instead, we simply register both tags as being possible for the sequence. In our approach, we do not look at context and as such we can not properly disambiguate. Allowing for multiple tags does not restrict functionality, while leaving open the possibility for disambiguating in the future.

[3] six is the approximate average length of words in Dutch and seems a good choice in preventing ambiguity[87].

[4] The average lengths of all firstnames in our list is six.

The performance of the named entity recogniser (for the types Person-Other, Organisation and Location) was tested against the CoNLL-2002 shared task. The data used is that of a Belgian newspaper [79].

We manually first tuned our data using the training set. Additions and corrections were made to the lists of person, organisation and location names based on the training data (although we were conservative and made only those additions significantly boosting performance).

Whilst the pure gazetteer look-up approach had a very high precision (around the 95%) the recall was quite low. Especially for the shorter Organisations and locations lists. We made several attempts to improve the performance for these entity types.

Our first attempt used the word features suggested by Bogers [8]. In addition to matching candidate words against our list, we also matched against collected character pre- and postfixes (collected from the lists, four characters). Whilst slightly increasing recall (approx. 5% for both types) it significantly cut down a precision to nearly a half (for locations) to a quarter (for companies). Similar findings of relatively low performance with Bogers's approach are also confirmed by Sporleder [83].

Dropping the automated approach, we simply examined the CoNLL training data and examined lists of false negatives for patterns. For locations we made a simple list of dynamic word ending heuristics (e.g. -city, -park, -street, -lane). This boosted performance slightly less than the pre- and postfix approach (3%), but impaired precision only slightly (~7%).

We also looked for a solution for recognising Organisation names and found that many of the unrecognised names consisted of all capitals. Hence, we added a simple heuristic for name recognition specifically for this: nouns that consist of all capitals are by definition semantically tagged as (possible) company names. This decreased the precision about as much as it improved recall (~15%). Even though this may tag phrases incorrectly, we do not believe this is a problem since the tagging is only relevant for us in the presence of a question. A bigger issue is the presence of all capital sentences (or messages). However, these are likely to be filtered out due to their formatting (see section 2.3.2).

After obtaining satisfactory performance on the training set we finally ran our entity recogniser on the test set. The results of both are shown in table 4.3. Compared to the work of De Meulder, et al. (note that this is a different dataset) our recogniser has a 10% lower F-score for names, but a higher (6%) score for organisation names. The F-score for location recognition is about the same. They typically have higher recall, whereas we have higher precision. In the CoNLL2002 task this approach would (overall) have taken place 10 amongst the 13 participants [64, 79].

For testing we removed the occupations from our person recogniser (since that was not part of the CoNLL task). Some names in the CoNLL corpus were (strangely) marked as MISC. This is a category we do not support. This slightly impairs the precision and recall statistics for person recognition.

We recognise the prime weakness of the list approach is lack of coverage. The heuristics were added to cope with this. We found that results were highly dependent on properly recognising

Table 4.3: *Named Entity Recognition performance (CoNLL2002). Training set results on the left (used for manual tuning and indicating upper expected performance bound), test set on the right.*

	Train			Test		
	Precision	*Recall*	*F*($\alpha = 1/2$)	*Precision*	*Recall*	*F*($\alpha = 1/2$)
Person	96.33%	38.33%	67.33%	92.22%	37.15%	64.68%
Organisation	78.93%	34.93%	56.93%	83.82%	35.43%	59.62%
Location	88.87%	60.63%	74.75%	83.82%	65.23%	74.52%
Overall	90.17%	43.19%	66.68%	86.88%	42.07%	64.48%

just very few words correctly. As such, the best solution for now is to make forum specific lists (or at the least: update the lists with forum specific terms). Effectively covering new domains. This is a time consuming task. An investigation into using automated methods for extending these lists, given a forum as input, would be useful.

4.3 Question-Answer Linking

The identification of question and corresponding answer blocks is a more objective task than that of summarization [63]. It has also be shown (on specific corpora) that a significant portion of fora posts consist of questions (36%) and answers (43%) [47].

Since we are capable of detecting question sentences (section 4.1) and a number of semantic tags (section 4.2), we turn to a last step which is linking questions and answers. In contrast with Kim, et al. we find these relations at the sentence level instead of the message level, since the former is more useful for summarization [47]. The usefulness of linking has been shown by Zechner who concludes that while such linking does not significantly affect the informativeness of dialogue summaries it does significantly increase the coherence [98].

To find a question-answer pair we look at each message and find the messages its refers to. In those messages we identify all question sentences which have at least one answer type set (the answer types are the possible semantic tags that match to the question). With this information we scan the current message for lines with matching semantic tags. Once such a line is found the question-answer score of that (answer)line and the questionline are increased.

This has a few implications, for example when a question in one message has multiple answers in other messages, the question-answer score of the question can become quite high (indicating the importance of the sentence). Similarly, if one line in a message provides multiple answers for different questions in (one or more) referred to messages, the line would have a high score. This is exactly the behaviour we want.

One might argue for summarization purposes it would be best to always include both the question and answer part of a pair. However, doing this is tricky since their can be multiple pairings. It would also conflict with weighing at the message level (see section 2.2.2). The best solution to this would probably be creating a separate dependency structure that would explicitly require pair selection and merge this into the message weighing process. This still leaves a question-answer multiplicity problem. We did not investigate this further.

Chapter 5

Polarity and Subjectivity

'Objectivity requires taking subjectivity into account.'

(Lorraine Code)

The increasing appearance of Weblogs has sparked interest in finding opinion clauses in texts [16, 70]. The two important concepts in computational linguistic research related to this are subjectivity and polarity. The following sections give an overview of what these concepts actually mean and how they are related.

5.1 Concepts

5.1.1 Polarity

With polarity we mean the semantic orientation of words, sentences or entire texts. This orientation can be either positive or negative. As such polarity has a directional component (+/-) and a strength (e.g. good, better, best). There appear to be more words with a negative polarity, but the positive ones are much more frequently used [16]. We also observed this trend in the Spoken Dutch Corpus (CGN) [22].

The smallest unit in language that exhibits polarity information is the morpheme. Good examples are affixes like *ill-* and *well-*. However, much research has focused on the word level partly due to the focus on English language in which the sub-word level is less interesting because of the rarity of compounding.

Adjectives are used to assign qualities (e.g. 'the *nice* party'). Prior research has found that they are frequent bearers of polarity information. There also seems to be a very high degree of agreement concerning the direction of the polarity for a large set of commonly used adjectives

[12, 31]. Material adjectives (Dutch: stoffelijke bijvoegelijke naamwoorden) have comparatively less subjective payload by themselves: the stone house, the wooden table (with some rare exceptions: ‘een *gouden* kans’ (a *golden* opportunity)). But they can act as prefixes with an intensifying meaning (e.g. ‘*het steenouden huis*’ (the really old house)) or just by repetition (‘she was getting redder and redder’).

Besides adjectives there are other classes of words carrying polarity information as well, most notably verbs and nouns. Adverbs interact with these to either change the direction of the polarity (e.g. ‘*not* funny’) or the strength (e.g. *‘very* funny’). It has been found that the evidence provided by the polarity of adjectives, adverbs and verbs combined yield the best performance for sentence level polarity classification [35].

These findings are also reflected in Dutch language. A glance at the word frequencies in the Spoken Dutch Corpus (CGN) reveals many frequently occuring adjectives and adverbs have some polarity effect. The adverb ’heel’ (very) occurs most frequently and the adjective ’goed’ (good) is in second place. However, there are also many infrequently occuring adjectives that exhibit polarity. This confirms the finding of others that a low frequency by itself is indicative of polarity [96].

For verbs the picture is different. There are over three times as many verbs as there are adjectives in the CGN. No inherently polarising verb can be found with a frequency larger than approximately one thousand (< 0.07% of the total number of verbs). This reflects many frequently occuring verbs have a polarity value depending on the subject of the verb and for transitive verbs also on the direct object of the verb (e.g. ‘*the man* beats *the child*’ (-) versus ‘*the boxer* beats *his opponent*’ (+/-)). The indirect object can play a role in the strength of the direction of the semantic orientation (e.g. ‘he beats him with his *fist*’ versus ‘he beats him with a *baseball bat*’).

The frequencies suggest that annotating the polarity of adjectives and adverbs is where the most gain concerning polarity classification can be made and is also the most useful approach for bootstrapping machine learning procedures. Nevertheless many polarised words appear infrequently which brings us to the single biggest problem with polarity recognition which is coverage.

While using finite word lexicons with polarity information is a commonly used approach, it suffers from the limitation that many of the low frequency polar words are not included [1]. Having a complete coverage polarity lexicon is not only impractical but also impossible due to the highly creative usage of natural language. Researchers have found various ways to cope with this, from clustering [31], using WordNet [45] (which is somewhat disputed because of its generality [78]), estimating the polarity of words using search engine technology [90, 95], to automatically building huge polarity corpora by clever usage of the structure of HTML pages [44].

An overview of word classes and their effect on polarity is shown in table 5.1. Some of these depend only on the word itself (adjective, verb and noun), while all the others somehow interact with other words to modify an existing or define an initial polarity. Notably, a number of frequently used adverbs and adjectives engage in a gradability relation with the word they act

Table 5.1: *Word classes and polarity effects.*

Class / Constituent	*Direction*	*Strength*
Adjective	Defines (*good*, *beautiful*)	Defines (*good*, *better*, *best*)
Adjective, Adverb	Inverts (*not* good) / Negates (*too* beautiful)	Intensifies (*very* good) / Diminishes (*probably* better)
Verb	Defines (*understand*)	-
Verb / Subject	Defines (*the man* beats / *the boxer* beats)	-
Verb / Object	Defines (*he* hit *the boy* / *he* hit *the button*)	-
Verb / Indirect-Object	-	Intensifies (with a *crowbar*) / Diminishes (with his *finger*)
Noun	Defines (he was a *bastard*)	Defines (he was a *lousybastard*[1])

[1] This compound is not valid in English, but is in Dutch were adjectives can be 'glued' to the noun.

upon [32]. Patterns of word classes that interact to form a polar meaning are also found in other research [76].

Adverbs have a number of interesting effects not seen for other word types. They can invert the direction of polarity. Inversion usually does not make the meaning of a polar word equal to its antonym, as is suggested by some research (e.g. *not good* is not the same as *bad*) [67]. Hence it has an inherent diminishing effect as well. Adverbs can also negate whatever follows as is the case for the word 'too'. Adverbs can also modify the strength of other words, either making it stronger (intensifying) or weaker (diminishing). Diminishing is sometimes also called *speculation* [94].

Table 5.1 is not exhaustive or definite. There are exceptions like noun phrases appearing in a gradable way (e.g. '*beetje*' [*a little*]). There are other classes of words that have some effect on polarity as well, like conjunctions [31]. Additionally there are also pattern interactions between these classes, for example many adjectives postfixed with *-ness* (or *-heid* for Dutch) become nouns with the same polarity as the adjectives (e.g. *goodness*, *badness*).

There is a large body of purely linguistic research for Dutch and other languages dealing with polarity and in particular with polarity in the negative direction (also referred to as Negative Polarity Items (NPIs)). Work in that field also indicate verbs are quite important for polarity and relatively underlighted [42]. This is probably due to their more complex relation with other words as shown in table 5.1.

Determining the direction of polarity is not free of problems. In particular phenomena such as sarcasm and metaphors make this a challenge. Also mapping the polar strength to absolute values is difficult as well since this is highly dependent on a persons perception and the surrounding context. Nevertheless, the strength ordering of polarities of different annotators seems to be highly consistent [96]. Linguistic researchers place negative polarity expressions into four

Table 5.2: *Subjectivity and polarity.*

	+ Polarity	*- Polarity*
Objective	The bike is fixed	The bike is broken
Subjective	The bike is beautiful	The bike is a joke

categories: superweak (nonverdical), weak (descending), strong (anti-additive) and superstrong (antimorph). However, this model is not free of problems [34]. In the case of negation it becomes less clear which category applies, which is similar to the inversion effect described earlier.

We conclude this section with a list of polarity tasks arranged from (relatively) easy to hard:

- Determining whether an expression has a polarity.
- Finding the direction of the polarity.
- Finding the strength of the polarity.

5.1.2 Subjectivity

Subjectivity is difficult to define. A commonly used definition is that subjectivity has a *source* and a *target* and that it conveys a *private state* [94]:

- $The\,party_{target}$ is big. (source is the writer)
- I_{source} think $the\,party_{target}$ is big.
- $Desmond_{source}$ thinks $the\,party_{target}$ is big.
- "$The\,party_{target}$ is big", $Desmond_{source}$ said.

Something essential to realise is that many first-person expressed subjective sentence can be rephrased in a seemingly objective way and vice verso. The first two sentences above are examples of this phenomenon. Nevertheless, replace 'big' with 'fun' and suddenly it is obvious that the first sentence is really always subjective regardless of its phrasing. Hence, some words are *inherently* subjective. These words are the stand-alone polarising words we met in the previous section. Most notably adjectives, adverbs, verbs and nouns attributing a polarity by their mere presence.

This brings us to the relation between subjectivity and polarity which is expressed in table 5.2. It is non-trivial to try to make the objectivity-subjectivity distinction really explicit, since it comes down to a philosophical issue. The most intuitive definition is that objectivity is shared by a vast majority. A simple test for an even stronger definition incorporating polarity is: Is there anyone who would hold the opposite point-of-view? (would anyone state the bike is not broken?) If so, it is subjective, otherwise it is objective. The fact that there exist polar objective expressions is recognised by other research as well and they are sometimes called *evaluative factual* [1, 69].

If you were to prepend every objective sentence in table 5.2 with 'I think' these sentences would become subjective. The word 'think' is part of a larger class of cue words signalling subjectivity (specifically the opinion subset). We can say sentences are necessarily subjective due to the presence of inherently subjective words or cue words. But we can never state a sentence is necessarily objective without checking the truth value of the statement. This of course is impossible to do automatically since it requires inspection of reality. Therefore a more practical point-of-view used in subjectivity research is that objectivity depends on the intention of the author. If something is presented in an objective way it is assumed to be objective otherwise it is subjective [12].

So it is important to realise that subjectivity and objectivity as used in computational linguistic research is *not* the same as subjectivity and objectivity used in everyday language. This has a direct effect on identifying opinion statements which in this field is synonymously used with finding subjectivity [35].

For finding subjective sentences one needs to look at (ordered from easy to complicated):

- Cue phrases (*I think*, *I believe*) [see next section].
- Words with some inherent subjective polarity (*good*, *bad*).
- Words which have some definite subjective polarity when they co-occur with other words (*the man* hit *the boy*).

5.1.3 Opinions

The task of finding opinions in a text is known as opinion mining. Some research has also focused on finding the sources of expressed opinions [14, 16]. Or even considering the relation between opinions and economic effects [28, 65].

Opinions are always subjective. Hence, the strong interlink between the two concepts. But there are sentences that are subjective and not (phrased as) an opinion and recall from the previous section that it is also possible to phrase an opinion as a fact. We ignore these problems conforming to the definition of subjectivity used in the field.

For the type of content under consideration cue phrases as suggested by Nigam [69] and also used by Choi [14] seem quite promising. In this method, multi-word phrases are collected that are indicative of the expression of an opinion. These are later used to classify sentences as opinion bearing. Some examples (cue phrases are in italic):

- *Ik denk* dat je je daarin vergist.
  I think you are mistaken.
- Ik vind het heel erg apart.
  I find it very awkward.
- Dan moet je *volgens mij* al richting ziekenhuis.
  According to me you have to go to the hospital then.[2]

[2] This translation is intentionally a bit awkward to capture the approximate phrasing of the Dutch original.

When taking this approach the risk is in mislabeling sentences. For example, the sentences ‘I think about her’ and ‘I find bacon’ do not express opinions yet they contain the cue phrases. Since there is no real solution for this (other than including many long and specific cue phrases), we acknowledge the existence of this problem in this research, but ignore its effects.

5.2 Approach

It would be very useful to be able to find opinion clauses (subjective) in texts for this research. Our domain however differs somewhat from that which has been used in most research in this field to date which has mostly focused on (well-edited) newspaper articles. Recent years have seen some shift of focus to blogs which is closer to the type of data this research focuses on [16, 65, 71, 70]. Nevertheless a difference remains, namely that we are dealing with hierarchical dialogue summarization.

There are three tasks that interest us:

1. Finding objectivity / subjective for sentences in general.
2. Finding opinion clauses automatically.
3. Finding the source of the expressed opinion.

The last task is somewhat aided by the availability of metadata. We already know who has posted what message and we also know which other participants there are in a discussion.

For finding opinions we use simple cue-phrase detection as described in the previous section. A list of approximately forty Dutch cue phrases was used for this purpose. The phrases are based on a manual analysis of about twenty threads.

Finding subjective sentences is probably the most challenging of these three. After creating several prototypes for the task we finally decided to use a very simple approach. First, we assume sentences that contain polar adjectives are by definition subjective. This is a simplification, but the prior research mentioned shows this is a reasonable assumption given the adjective list is of good quality. Second, we are not interested in the direction of the polarity, just whether there is polarity at all (and using that as a subjectivity indicator).

We use the lists of adjectives as constructed by Hatzivassiloglou [31]. These lists of positive and negative adjectives were fused into one and translated to Dutch by machine translation followed by manual corrections. We extended the list by using several terms as bootstrap words to the Dutch EuroWordNet [91]. The synonym and antonym relations of EuroWordNet were used. The results were manually corrected wielding out strongly context dependent adjectives. This yielded another 582 adjectives on top of Hatzivassiloglou’s 1203 providing a coverage of 1785 adjective lemmas for Dutch. For English the original list are used without any augmentations (i.e. no effort was made to extend this list further, as English is not the focus of this research).

A problem was that all the Dutch adjectives in the adjective list were in baseform. To solve this we created a small tool that takes the original list to create all inflected forms. It creates comperative and superlative forms of each adjective. For these forms and the baseform it also creates wordstem + e forms (specific to Dutch). This generates a final list of 10691 adjective forms. If a sentence contains one of these words (and the word is PoS tagged as being an adjective) the sentence is marked as being subjective.

Apart from words we also use the presence of several other phenomena. Wiebe states that sentences ending with an exclamation mark are always subjective [94]. Hence, we also mark such sentences as being subjective.

We also use a self invented addition which is specific to on-line communication. The presence of emoticons can also convey a private state. Similar to the assumptions underlying the adjective-based subjectivity marking we ignore the cases where a private state is conveyed having no bearing on the sentence. All sentences containing emoticons are marked as being subjective. Emoticons are recognised by the semantic tagger using regular expressions (described in section 4.2).

Chapter 6

Hierarchical Dialogue Summarization

‘One of the Internet’s strengths is its ability to help consumers find the right needle in a digital haystack of data.’

(Jared Sandberg)

Interestingly, there are many views of what constitutes a good summary. We adopt the definition given by Hovy [38]:

> ‘A summary is a text that is produced from one or more texts, that contains a significant portion of the information in the original text(s), and that is no longer than half of the original text(s).’

Portions of the texts of previous messages are often quoted in posts, so we consider the original text to be all unique (i.e. non-verbatim-repeated) text in a thread. The reduction of text length that is the result of summarization is frequently referred to as compression. This is frequently expressed as a percentage relative to the size of the original text(s): The compression ratio. The exact ratio should be user selectable in a summarization system, but an upper bound of 50% fits the definition above. Useful (single-document) summaries appear to be no longer than 35% of the original text and no shorter than 15% [38].

In this chapter the unique challenges posed by our data as well as the difficulties in summary evaluation are discussed. Finally, an algorithm for summarizing our data is presented.

6.1 Aspects

There are several aspects related to the summarization task:

- Type: Do we use the original document wording or rephrase it? *Extractive* versus *Abstractive.*
- Focus: Should the summary be based on the document content alone or on a provided query? *Generic* versus *Focused.*
- Cardinality and Structure: How many documents need to be summarized? *Single-document* versus *Multiple-document* versus *Hierarchical.*
- Participants: By how many people have the documents been written? *Monologue* versus *(Multi-Party) Dialogue.*
- Purpose: Should the summary be exhaustive or only give a superficial impression? *Informative* versus *Indicative.*

A detailed description of these:

- *Extractive versus Abstractive*
  In the first type of summary a portion of the original text is *extracted* and arranged in a particular order. This is sometimes called selection-based summarization. In the other approach the text is first interpreted to yield semantic representation, which is used to generate an *abstractive* summary. This is also referred to as knowledge-based summarization [43]. Abstractive summaries can not be generated without domain knowledge. Frequently template based approaches from the field of information extraction are used. However, because of the manual labour involved this is not a practical approach for applications requiring wide-scale deployment.
- *Generic versus Focused*
  When a user is looking for very specific information the summary can be geared towards the information seeking goal. This requires the user to express his (or her) search goal as a query and is also called focused (or query-based) summarization [9]. If an explicit information driving goal is not present, a generic summary can be created instead.
- *Single versus Multi-Document versus Hierarchical*
  Initially the summarization field focused on summarizing a single document. However, recent years have shown a shift of focus towards summarizing multiple documents, which is an inherently more complex problem [38]. The present day interpretation of this term means assembling multiple documents with overlapping information (such as news items concerning the same event) into one combined summary. This type of multi-document summarization has received extra attention due to the tasks at the Document Understanding Conference (now part of the Text Analysis Conference). Extra challenges for such systems are: redundancy, temporal aspects, higher required compression ratios and difficult co-reference resolution [29]. Even though thread summarization deals with multiple documents, the relations between these are distinctly different from the assumption of traditional multi-document summarization. Hence we adopt the term *hierarchical* for this, coined by Farell, which implies implies some referential connection between documents (and not pure overlap) [23].

- *Monologue versus (Multi-Party) Dialogue*
  Most of the research has focused on what are essentially monologues. News articles, and more recentely, blog postings, are commonly used. However, this research focuses specifically on messaging interactions between multiple participants. This is distinctly different, because there is an ongoing dialogue and there are relations between (parts of) the messages.

- *Informative versus Indicative*
  Indicative summaries are generally shorter and serve to give an impression of what is in the text. They can be used for selecting whether a text is interesting or not. Informative summaries are usually longer and attempt to cover all topics raised in the original text [23].

Studies regarding either the Hierarchical or Dialogue facets are uncommon in present day summarization research. They raise some interesting questions: Are single-document dialogues possible? Yes, think of a chatlog. And hierarchical monologue? Yes, this would be consistent with for example a cluster of hyperlinked web-pages. A multi-document dialogue would then be several dialogues about the same topic (possibly by different participants).

We would like our research to be applicable easily to a wide variety of fora. Therefore, using an ontology for each domain would not be practical. Hence, we focus on *extractive* summarization. We use questions posed in the posts themselves as a indicators for extraction. In that sense we do not generate a generic summary. However, since the user is not explicitly requesting a certain direction in the summary, we can not really call this focused either. Therefore, we adopt the term *data-focused* by which we mean the focus of the summary is determined by the data itself. It should be obvious our data is *hierarchically* arranged *dialogue.* Since we already know the user is interested in a particular thread, there is not much use for a purely indicative summary. But generating a full-coverage informative summary is difficult. Therefore this research focuses on finding the main focus of the thread. We call this *semi-informative.*

There is still one thing left to define. Farell uses the term hierarchical discourse summarization to generate a summary of each message in a thread individually as opposed to generating one large summary. In this research we focus on the latter instead. We introduce the term *monolithic* for this, the opposite (Farell's case) being *manifold [23].*

To 'summarize': This research focuses on generating *monolithic extractive data-focused semi-informative* summaries based on *hierarchical dialogues.*

6.2 Evaluation Methods

Evaluation of summaries is a difficult task. This is largely due to the fact that different people do often not agree on what the best summary for a given text is [38]. Even the same person can disagree with himself over time on the quality of a summary. It has been shown that inter-annotator agreement for summaries is low. Also, for simple sentence extraction there is disagreement as to the importance of sentences. Nevertheless, humans do appear to have some level of agreement on this task. The longer the summary, the more possible variation and the less agreement there is [10, 82].

Evaluation of a summary must always be done with the goal of the summary in mind. Different summaries are appropriate for different tasks. Remember that our summaries are intended to be semi-informative and data-focused. Alterman gives some general considerations that are also indicated to be a good starting point by Bosma. These are that a summary should increase the *utility* for the reader, maintain *coherence* (linguistic quality) and maintain *coverage* (content quality, sometimes called informativeness) [4, 10]. Translating this to evaluation metrics: *Intrinsic* evaluation measures the quality aspects and *extrinsic* measures the utility [38].

Recent years have seen the rise of the suite of Recall-Oriented Understudy for Gisting Evaluation (ROUGE) metrics, largely due to their use at the annual Document Understanding Conference (DUC). Without going into the details, these are synthetic n-gram overlap metrics. They have shown to correspond reasonably well with human judgement. Although their prime weakness is their inability to detect paraphrasing, synonymy and equality of importance for phrases. A possibly more appropriate measure is provided by Basic Elements (BE), which also takes the subsentence level and the beforementioned weaknesses of ROUGE into account. Alternatives include factoid and the related pyramid evaluation methods. However, these require heavy manual annotation [10, 40, 57, 82].

Given the weaknesses of synthetic metrics for the conceptually simpler single-document summarization task, it should come as no surprise that no specific metric exists for multi-document summarization let alone for hierarchical dialogue summarization. This raises the question what was used before metrics such as ROUGE existed? Such studies usually revolve around comparisons against a gold standard in the form of multiple human-written reference summaries. Whilst these provide a performance indication they are also necessarily incomplete. A summary may exists which meets both coherence and coverage criteria while differing from the reference summaries. Additionally, there are problems with specifying summarization instructions that can have a huge impact on what sentences are selected. Other systems are evaluated with user studies (either intrinsic, extrensic or both), which is an approach we also adopt in this research [24, 50, 59, 58, 98, 100].

Concerning the development of a synthetic metric for evaluation of hierarchical dialogue summaries, Zhou and Hovy [100] argue that especially the BE metric may be useful here. However, by doing so effectively ignore the dialogue aspect, which is so characterising for this type of data. The evaluation difficulty caused by conversation context is recognised by Kim, et al. [47].

It currently remains doubtful whether a useful synthetic metric can be derived even from a large annotated corpus. Such a metric must take many variables into account and it must also allow for multiple summaries to be equally good. But the most important aspect remains the purpose of the summary for the user. In this light the *Problem-Solution* type of threads offer the most promise. A summary of such a thread should aim to clearly state both the problem, the proposed solutions and indicate the working solution. Compared to other types of threads this could be evaluated relatively easily based on an annotated corpus. Whilst not yet a synthetic metric, it is one step in the right direction.

6.3 Approach

Much research in the field beyond single-document summarization has focused on aggregating multiple news articles which is a multi-document monologue summarization task [21, 72, 73]. With at least one exception [80], almost no attention has even been given to the comments on such news articles. Even though it has been suggested we need to look beyond newswire data into new types of data such as on-line discussion fora and blogs [100].

The field of multi-document summarization was given a boost by the PageRank derived LexRank approach to the problem. This approach essentially arranges sentences in a graph where vertices represent sentences and edges are defined in terms of a similarity relation between pairs of sentences. Graph centrality based heuristics are used to determine important sentences [20]. A different approach was taken by researchers from Microsoft who observed that summaries include frequently occuring words which lead to the creation of SumBasic with several different later incarnations including Pythe [68, 89]. Using such approaches directly in the hierarchical dialogue summarization task is not a good idea. Using word frequencies is unreliable, because multiple people use (slightly) different terms to describe the same concept 80% of the time [23]. Hence, frequency based centrality metrics, for example those used by Radev and LexRank by Erkan, are less useful for our data [72, 20]. Taking the hierarchical structure of a thread into account is essential. Treating the entire discussion as one monolithic unit has been shown to yield poor summarization results [50]. Surprisingly, closely related prior research exclusively uses simple term frequencies even while recognising the ineffectiveness of this approach [23, 50].

Lexical chains are more promising here since they can deal with synonymy and other word relations. However, their prime weakness is the reliance on WordNet, which is difficult to keep up to date and at least always one step behind present day language use [6]. Maximal Marginal Relevance (MMR) method is more applicable due to its language neutrality. It was invented primarily for multi-document summarization to select sentences close to a stated query, but that are sufficiently dissimilar to each other to all be included in one summary. Hence, it relies on a query (which could be taken to be the thread topic) and on measuring similarity between candidate sentences (Ratcliff-Obershelp comes to mind, as cosine similarity is again based on word similarity) [29, 98].

In hierarchical discussion summarization human annotators are biased towards paragraph lead sentences *and* the last few sentences in a discussion post[1] [23]. Therefore, instead of relying on word frequencies, we mimic this approach and rely on the position of sentences augmented with referential information instead. The exact algorithm is detailed in the next section.

[1] This is a contrast with summary of news articles which usually follows a pyramid structure (with the first few paragraphs being the most important).

Table 6.1: *Summary distribution formulas.*

$f_{pmr}\left(\mathbf{m}, msg\right) = \frac{msg_{pmr}}{\sum_{m \epsilon \mathbf{m}} m_{pmr}}$	Where **m** is the set of all selected messages and *msg* is the message to recalculate the PMR for.
$f_{pt}\left(\mathbf{m}, \mathbf{a}, msg\right) = \frac{1 - {}^{msg_{pmr}}/{}_{author(msg)_{pmr}}}{\sum_{m \epsilon \mathbf{m}} author(msg) = author(m)} \cdot \frac{f_{pt}(author(msg))}{\sum_{a \epsilon \mathbf{a}} f_{pt}(a)}$	Where **m** is the set of all selected messages, **a** the unique authors that authored those message and *msg* is the message to calculate the PT weight for.
$f_{msgweight}(msg) = {}^{2}/{}_{3} \cdot msg_{pmr} + {}^{1}/{}_{3} \cdot msg_{pt}$	Where *msg* is the message to calculate the weight for and *pmr* and *pt* are message properties calculated by the above functions.

6.4 Algorithm

6.4.1 Message Filtering

Messages with an average sentence length or average word length whose z-score differs significantly from the average ($-3 \leq z \geq 3$) are filtered out and not considered for the summary. Similarly messages with a poor message formatting score (≤ 0.40) are filtered out as well[2]. Note that the first message in a thread is *never* filtered out.

6.4.2 Distributing Message Weight

Since messages have been filtered out we first recalculate certain metrics to ascertain they sum to one. These are the Positional Message Relevance (PMR) and the Participation-Talkativity (PT) factor (see section 2.2.2). The latter is expressed in a weight that is distributed inversely with respect to the PMR of the messages posted by the author. Meaning that PT weight for a specific author is shifted towards messages that have a low (or zero) PMR from that author. Of course, if the author has posted only one message, the entire PT weight is assigned to that message alone.

The PMR and PT are combined into the message weight $f_{msgweight}$. Fairly optimal weights have been determined by experimenting with various values.

[2] The 0.40 boundary for this was determined by experiment

6.4.3 Distributing Line Weight

One of the inputs to the summarizer is the requested number of lines the summary should consist of. This is either expressed as a percentage relative to the number of unique lines in the entire thread or as an absolute number of lines. The former is always calculated to an absolute number of lines. We call this variable simply *lines*.

These lines are now distributed over the messages by using the previously calculated $f_{msgweight}$ of each message. In the current approach the number of lines is always rounded upwards (ceil). This can lead to rounding errors, which we will get back to later. The basic line distribution formula is:

$$f_{lineweight}\left(msg\right) = ceil(f_{msgweight}\left(msg\right) \cdot lines\left(msg\right)) \tag{6.1}$$

Where msg is a message, and $lines\left(msg\right)$ is the number of lines the message consists of.

It is possible that messages get assigned more line weight for the summary than they are long. A pass is made over all messages to collect this excess assigned lineweight. This is redistributed starting at the message with the highest message weight downwards until there is nothing more to distribute (of course keeping in mind that redistribution does not result in excess line weight assignment).

There is a possibility the total number of assigned lines exceeds the number of requested lines for the summary due to the rounding errors introduced earlier. This is dealt with by removing lines from the message with the lowest message weight upwards. In case of equal message weight, messages that are posted later are preferred over those posted earlier.

The redistribution of lineweights yields two useful biases: a bias towards messages with a higher weight (increasing coherency); and a bias towards messages at the end of the thread providing a natural counter balance for the PMR's tendency to assign more weight to messages at the beginning of a thread (increasing coverage).

An example of the entire line weight distribution is shown in table 6.2. This is a thread consisting of four messages that is requested to be summarized to five lines. Assigned message weights are shown in the second column. The third column shows the line weights derived from this (rounded upwards, unrounded numbers are shown in parentheses). The next column shows the number of lines each of the messages consist of. Message B_1 is an erroneous empty message (but has been assigned weight based on its position). First, due to the rounding applied in the beginning there are now seven lines to be put into the summary while five were requested. Hence, the need for the weight removal step (fifth column). First one line is removed from the message with the lowest message weight A_2, this is still not enough. A line is thus also removed from B_1 which has the next lowest message weight and is more in the beginning of the thread (losing out against the equally message weighted, but 'better' positioned C_3). Even though we have five lines now, it is clear that the empty message B_1 still has to many lines assigned, shown in the sixth column. This is redistributed as shown in the seventh column of the table. A_1, the message with the highest weight is assigned the excess line weight of B_1. The last column

Table 6.2: *Line weight distribution example.*

Msg	$f_{msgweight}$	$f_{lineweight} \leq 5$	$lines\,(msg)$	*Rm*	*Excess*	*Redist*	*Final* $f_{lineweight}$
A_1	0.40	2 (2.00)	5			+1	3
B_1	0.25	2 (1.25)	0	-1	1	-1	0
A_2	0.15	1 (0.75)	4	-1			0
C_3	0.25	2 (1.25)	3				2

shows the final line weights that sum to five lines as requested. The intermediate messages B_1 (empty) and A_2 have effectively been canceled out by line weight redistribution and correction.

6.4.4 Selecting Lines

Now we know how many lines each message may contribute to the final summary, we can actually find *which* lines should be included. The heuristics used for line selection are as follows:

- *Question-Answer*
  Lines that are marked as either being an answered question or an answer to a question have the highest inclusion priority. Question lines with more corresponding answers are preferred over those with less and similar to that answer lines that answer more questions are preferred over those that answer fewer.
- *Longest (Standalone) Question*
  Next, the longest (in sense of character length) question line in the message is preferred. This was decided after observing that in the case of multiple questions the longest is usually the most descriptive and the shorter questions tend to be satellites of the longer (being much less important). Questions that are standalone (the only line in their paragraph) are preferred over others. Short questions (less than five words) are not considered, since they are less likely to be descriptive enough for a summary.
- *Quoted Lines*
  Lines that are quoted in one or more separate messages have third priority. The more a particular line is quoted, the more it is preferred over other quoted lines. This thus makes use of the referential structure present in threads at the line level.
- *Top-Bottom Interleave*
  Now that all specially marked lines have been determined. The next heuristic is picking top and bottom lines from the post (this is based on observations by Farell [23]). This heuristic is extraordinarily simple. From the remaining sentences (those not covered by any of the other heuristics) we simply pick the first one, then the last one, than the second one, etc. We call this *top-bottom interleave.*
- *Low Priority Lines*
  Finally, lines that have been marked as low priority are considered in longer to shorter order. For example greeters that are not often included in summaries [23]. Other low

priority lines are lines indicating the start or are part of a list, containing only an external reference and containing no alphabetic characters and closers.

For all the above line selection criteria (except for top-bottom interleave which is solely based on positional criteria) a second criterium is used when lines are of equal preference and a preference for either subjective or objective summary content is set. To illustrate this we use quoted lines: If two lines, one objective and one subjective, are quoted an equal number of times, then the second criterium is applied. Meaning if a preference is set for objective sentences the objective sentence is prioritized higher than the subjective one. If no preference is set, both objective and subjective sentences are treated equally. In such cases the normal behaviour is that sentences towards the end of the message have a higher priority.

Lines are always included in their original message order, not in the order they were selected by the algorithm as this would conflict with reading coherence. Compressing the dialogue can make it difficult for users to maintain context. Thus resolution of anaphoric reference is included in the summary as described in section 3.4. The bias towards selecting question and answer sentences was inspired by Klaas who stated the presence of a question is highly relevant to understanding the crux of a discussion. His conjecture is further supported by the evidence found by Kim, et al. : many interchanges of messages consists of questions and answers [23, 48, 50].

Consider the following message (lines are numbered for referencing):

1. Hello,

2. I noticed my computer crashes sometimes under heavy load.
3. Today I noticed that I can reproduce this.
4. When I compress a number of wave files with lame (2 threads) my computer always crashes after about four songs.
5. Just poof!

6. My BIOS settings have been ‘spontaneously’ reset more than once.
7. So I suspect the motherboard is the culprit.
8. It is not a very old computer, so the battery is probably not the cause?

9. Which component is most likely to cause this and how can I confirm this?

10. Thanks in advance for the help!

Assuming there is an associated answer to line 8 and that line 2, 3 and 4 are quoted in an other message. The priority order would be as follows: 8 (question-answer), 9 (longest question), 2, 3, 4 (quoted lines), 5, 7, 6 (top-bottom interleave), 1, 10 (low priority).

Thus, if this message were assigned a lineweight of three the following lines would be selected:

2. I noticed that my computer crashes sometimes under heavy load.
8. It is not a very old computer, so the battery is probably not the cause?
9. Which component is most likely to cause this and how can I confirm this?

These form a relatively coherent and informative summary given the content of the original message.

Table 6.3: *Summarizer output example.*

Topic: Bios update	
A:	I [A] ordered some new components for my [A's] pc that is yet to be delivered. It [the update] is therefore not possible in my [A's] case. Is it possible to perform a Bios update without a floppy?
N:	It is often possible to start the flash tool from the bios, you only have to put the bin/rom file in a read accessible place, for example a FAT partition or a USB stick. The advantage is that there's no need to make it bootable first.
A:	Ok, thanks everyone!
T:	That reminds me [T] of when I thought a game wasn't working because I [T] thought that my [T's] DVD-drive was incompatible.
M:	I [M] would do this via EZ-Flash (press F2 during start-up) and not via that small Windows-program.

6.5 Output

The output of the summarization process is presented as a dialogue. The topic title is included as well as the selected lines of the selected messages prepended with the names of the authors involved. Message boundaries are visible. So, structurally it looks like:

Topic: <Topic-Title>

<A>: ....
<B>: ....
<A>: ....
<C>: ....
etc.

A real world example is shown in table 6.3. Names have been abbreviated to their first letters, spelling and grammar errors have been left intact (the original version was in Dutch). The purpose of this example is purely to show the output format (not the relation between the input thread and the output text).

Chapter 7

Prototype Design

> 'There are two ways of constructing a software design. One way is to make it so simple that there are obviously no deficiencies. And the other way is to make it so complicated that there are no obvious deficiencies.'
>
> *(Charles Antony Richard Hoare)*

One of the purposes of this research is to present a prototype system in which the technologies described in the previous chapters are embedded. This chapter illustrates the functioning of the system as a whole from architectural / design perspective. Therefore, it is relatively technically oriented compared to the previous chapters. The main intention is showing how systems like these can be designed.

7.1 Background

For developing a system that does intensive natural language processing it is essential to choose a suitable language. Because of suitability and prior experience the Python language (the most recent stable version: 2.5) was chosen for the implementation of this system. One other important ingredient is the Natural Language Toolkit for Python, which is being developed primarily by Bird and Loper [7]. In a few places functions provided by Numpy were used for fast matrix manipulation (comparable to Matlab). Apart from the Hammer tagger toolkit mentioned earlier there are no other external software dependencies.

Table 7.1: *Scrapers (the term 'News' refers to news* ***websites*** *and not to newsgroups).*

Name	*Version*	*Reference Site*	*Type*
Admino	All	`http://luchthaventwente.nl`(defunct)	Forum
PhpBB	3.x	`http://www.phpbb.com`	Forum
React	1.9.5	`http://gathering.tweakers.net`	Forum
Replique	0.3.3	`http://forum.fok.nl`	Forum
vBulletin	3.6.x	`http://www.stand.nl/forum`	Forum
IMAP	4.x	Private Gmail Mailbox	Mail
GeenStijl	n/a	`http://www.geenstijl.nl`	News
NRC Weblog	n/a	`http://weblogs2.nrc.nl`	News

7.1.1 Scraping

Forum data is obtained using a technique called *scraping*. An HTML (or XML) page is parsed and we selectively extract some portions of the page. The specific parts to extract are determined manually and can differ across fora software. The modules of the system handling scraping are (naturally) called scrapers. Given an URL of a forum thread they scrape the page and possibly subsequent pages as well if a thread spans multiple pages. This is converted into an internal data structure which is discussed in section 7.2. Scrapers also filter out signatures. Usually, these are put in a different block inside each message. These blocks are ignored by scrapers (something which is less straightforward with e.g. e-mail as data source [17])

We have developed a variety of scrapers demonstrating the flexibility of the system. These are listed in table 7.1. As exploration in the early stages of this research, and partly in support of other research, scrapers were developed for other resources (news sites and e-mail). The Admino scraper was initially the main focus for this project, but the forum was shut down. Admino fora can still be found, especially on the University of Twente websites.

However, the primary resource used for scraping is the Tweakers forum, which uses the React forum software. Secondary sites are fok.nl (using Replique) and stand.nl (using vBulletin). The latter forum software appears to be the most popular on the Internet followed closely by the widely deployed PhpBB[1]. vBulletin is proprietary software, as are all other supported fora except for PhpBB.

7.1.2 Molding

The messages obtained by a scraper still need to be converted into our internal representation. This process is called *molding*. Message numbers, author names and posting dates (message metadata) are automatically molded inside the scrapers. However, molding of the message content is diverted to a special module called the HTMLMolder. This converts the HTML to

[1] See: `http://www.big-boards.com/statistics`

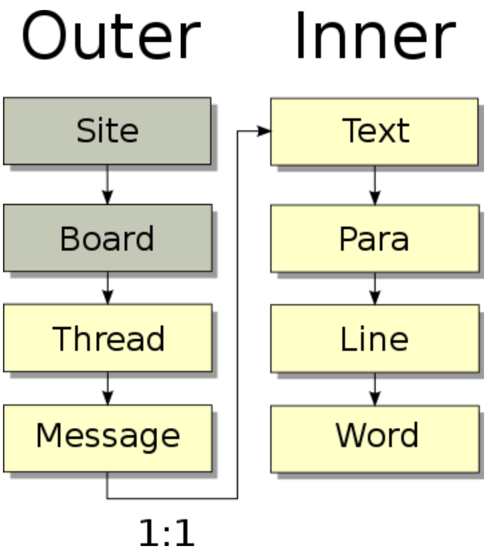


Figure 7.1: *Internal data structure. Relations are 1:*n *unless indicated otherwise. Site and Board have been coloured differently, since they are not relevant in the context of this research.*

Markdown[2], which is a plaintext representation. Subsequently, the TextMolder is invoked and performs the actual work using Punkt (included in NLTK) for sentence boundary detection and other postprocessing (see section 3.1 for details).

7.2 Data Structure

Figure 7.1 shows the internal data structure (the 'mold'). The Site and Board levels are included mostly for future extensions. The research at hand focuses on the thread level and downwards. The Outer structure uses statically defined built-in objects whereas the Inner structure inherits from the self-developed DynamicObject (more on the difference between the two in the next paragraph). Skipping Site and Board, you can read the figure as: a thread consists of one or more messages, a message has a text, text consists of one or more para(graph)s, etc. All the way down to the word level. Words also have a chars property which is of the built-in string type rather than a separately defined object.

The DynamicObject forming the basis for the Inner architecture is very flexible. When new properties are added to an object there is no need to explicitly define them elsewhere. DynamicObject offers functions for recursively converting objects (and their properties) to an XML representation (and also a more human-readable, but non-standard HML[3] output format). The conversion is invoked simply by calling: xml(text), xml(para), etc. The XML output is more

[2] For more information: `http://daringfireball.net/projects/markdown`

[3] HML actually means Human-readable Mark-up Language (a tongue-in-the-cheek reference to the goal of XML). It is used for internal development purposes only.

readable than Python's own serialisation, removing the explicit data dependency on the Python language.

The Outer objects also support XML and HML, but there the support code is explicitly hand-coded (and does not update automatically). The outer structure is static. It has few properties which are fixed. The Outer objects descend from the new-style Python class objects and all have their slots property defined to prevent properties from erroneously being added.

7.3 Modules

With the data structure in place the next step is actually operating on the data. Nearly all technologies described in this thesis were used and turned into modules (with the exception of tokenising, which is not technically a module7.1.2). The base class for modules is (straightforwardly) called Module. It provides standard iteration functions to traverse the various levels of abstraction present in the data structure (messages, paragraph, lines, etc.). All the descendants of this class are considered modules. Table 7.2 shows precisely what was implemented in which module, from bottom to top level, and where in the thesis the technology is discussed.

Most technologies were self implemented (using the interfaces natively provided by Python). There are two exceptions:

- Part-of-Speech (PoS) Tagging uses the external Hammer tagger toolkit with some minor adaptations.
- Partial Parsing was implemented using tools provided by the NLTK.

Most modules, with the exception of the Summarization module, enhance the datastructure with extra information (primarily by extending DynamicObject). This information is used by subsequent modules and thus creates dependencies between modules (e.g. the Partial Parser requires the information added by the PoS tagger). These dependencies are shown in figure 7.2.

Of course each module depends on the availability of data and indirectly on a scraper and the HTMLMolder/TextMolder as well. However, these rather technical dependencies between non-modules are not depicted in the figure.

Why did we implement many modules ourselves? Because we noticed that open-source (or free software) for the mentioned tasks is relatively scarce. The NLTK seems to offer the most promising collection of natural language technologies at this moment for Python. Although we also found the offering of FreeLing[5] (C++) impressive. Using many external packages also incurs design problems. Incompatible (and possibly changing) interfaces make it a demanding task to keep everything weaved together. This is draining on resources, even whilst still developing a prototype. Hence, we believe it is best to stick to a good integrated package (like NLTK) and only use other external components when it is necessary.

[5] http://garraf.epsevg.upc.es/freeling/

Table 7.2: *Implemented technologies (includes references to the thesis section discussing the technology in the third column).*

Description	*Module*	*Section*	*Level*
Part-of-Speech Tagging	postagger.py	3.2	Word
Partial Parsing	parser.py	3.3	Wordgroup
Semantic Tagging / General	semantictagger.py	4.2.1	Wordgroup
Semantic Tagging / Named Entity[4]	entityrecognizer.py	4.2.2	Wordgroup
Anaphora Resolution	anaphoraresolver.py	3.4	Wordgroup
Question-Answer Linking	answercoupler.py	4.3	Sentence
Sentence Type Detection	sentencedetect.py	4.1	Sentence
Subjectivity Detection	subjectivitydetect.py	5.2	Sentence
Message Readability	statisticscalculator.py	2.3.1	Message
Message Formatting	statisticscalculator.py	2.3.2	Message
Thread Structure Weight	threadstructure.py	2.2.2	Thread
Thread Structure Discovery	threadstructure.py	2.2.1	Thread
Summarization	summarizer.py	6.4	Thread

[4] Small part in Thread Structure Discovery as well.

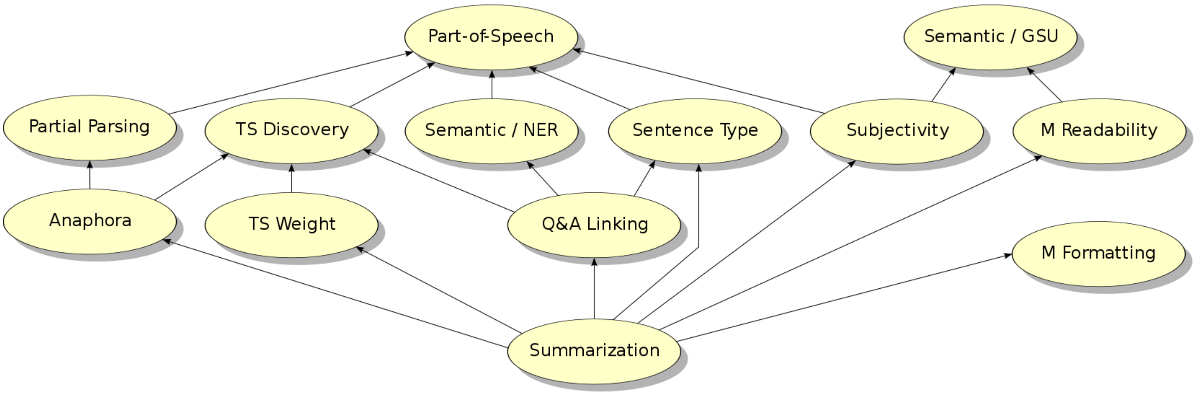


Figure 7.2: *Module dependency structure (abbreviated names are used, see table 7.2 for full names).*

We have come to believe Python is an excellent language for fast prototyping of applications using Natural Language Processing techniques. Building some of the more complex modules was a challenge, but Python's strong string manipulation capabilities greatly aided in the task.

7.4 Language Dependencies

The prototype uses some language dependent resources. We can divide modules into three groups: those that use no (direct) resources, those that only weakly use resources, and those that are strongly dependent on language resources. Language resource dependencies are as follows:

- No resource dependency:
    - Anaphora Resolution: Whilst dependent on Part-of-Speech (PoS) tags this module has no direct language resource dependencies. However, the heuristic determining what is in focus in a sentence could be tuned for new languages.
    - Question-Answer Linking: Relies on output of the Semantic Tagging parts, but uses no language dependent resources.
    - Message Readability: Calculated statistics are language neutral. The module adapts itself to the used language.
    - Message Formatting: The current implementation works with language independent clues.
    - Thread Structure Weight: Based on the language independent referential structure that is uncovered by the Thread Structure Discovery.
    - Summarization: As the last step, this depends on many other modules, but the algorithm is language neutral. Certain line length constants could be tuned for specific languages.
- Weak resource dependency:
    - Semantic Tagging/General: This uses a list with regular expressions covering a range of concepts. The larger part of this list is language independent, since it uses international abbreviations based on the metric system. However, adding support for a new language would require names for some numeric, date and duration expressions to be added (e.g. 'one', 'January', 'milliseconds').
    - Sentence Type Detection: Only the opening and closing phrases are language dependent.
    - Thread Structure Discovery: Directly dependent only for the 'wrote:' clue word, sometimes used to signal message quotations.
- Strong resource dependency:
    - Part-of-Speech Tagging: Requires a PoS-tagged corpora (CGN for Dutch, Brown for English) which is reduced to the Unified tagset (see appendix A).

- Partial Parsing: Needs rules for identifying noun phrases based on PoS tags.
- Semantic Tagging/Named Entity: Requires lists with names, locations and organisations. Each language would require new lists.
- Subjectivity Detection: Relies on a list with subjectivity signaling adjectives as well as a list with opinion cue phrases.

Language dependent resources have been made for both Dutch and United States English. We believe the system could easily be extended to support other languages. However, for non Western-European languages more adaptations might be necessary. Even though we use UTF-8 as internal storage format, which allows many typographical symbols to be used, the underlying model does make assumptions concerning, for example, the word order.

7.5 Testing

For the majority of the developed application, unit tests are used to verify the consistency of the internal workings. These consist of manually created datastructures passed on to modules or other components. If a unit test fails, either the software has been changed erroneously or the unit test is out of date. These tests have proven very helpful, especially to detect normally hard to debug problems at an early stage. As these test are very module specific, a complete discussion of each unit test is beyond the scope of this document.

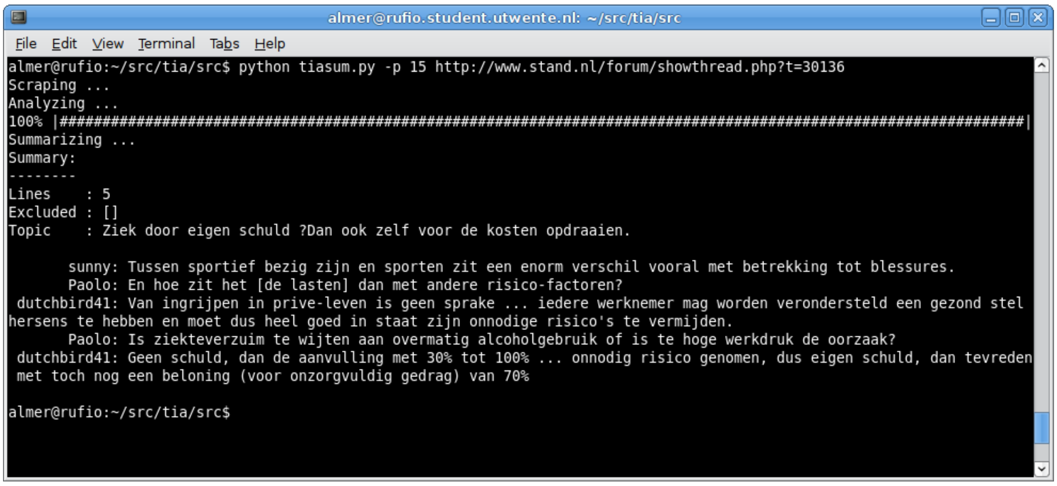
Besides automated tests, the software was also extensively hand-tested and tuned on many different threads. About forty threads were used for regular testing.

7.6 Interfaces

The developed prototype is codenamed Text Interaction Analyser (TiA). The primary input to the system is an URL pointing to a thread. The system only supports automatic detection for URL's for which the system was actually tested or where the user can explicitly indicate the type of forum software used. Broad automatic detection of the forum software used would be interesting and is technically feasible, but was not implemented due to time constraints.

Besides the URL the other inputs are:

- Language: The language the thread is in (default: Dutch).
- Retain Percentage: How much of the unique thread content should be retained in the summary as a percentage of the number of lines the thread consists of (default: 30%).
- Retain Lines: Amount of lines the summary should consist of. This overrides the percentage if set (default: unset).
- Bias: Either neutral, objective or subjective (default: neutral).



```
almer@rufio:~/src/tia/src$ python tiasum.py -p 15 http://www.stand.nl/forum/showthread.php?t=30136
Scraping ...
Analyzing ...
100% |#################################################################################################################|
Summarizing ...
Summary:
--------
Lines    : 5
Excluded : []
Topic    : Ziek door eigen schuld ?Dan ook zelf voor de kosten opdraaien.

       sunny: Tussen sportief bezig zijn en sporten zit een enorm verschil vooral met betrekking tot blessures.
       Paolo: En hoe zit het [de lasten] dan met andere risico-factoren?
 dutchbird41: Van ingrijpen in prive-leven is geen sprake ... iedere werknemer mag worden verondersteld een gezond stel
hersens te hebben en moet dus heel goed in staat zijn onnodige risico's te vermijden.
       Paolo: Is ziekteverzuim te wijten aan overmatig alcoholgebruik of is te hoge werkdruk de oorzaak?
 dutchbird41: Geen schuld, dan de aanvulling met 30% tot 100% ... onnodig risico genomen, dus eigen schuld, dan tevreden
 met toch nog een beloning (voor onzorgvuldig gedrag) van 70%

almer@rufio:~/src/tia/src$
```

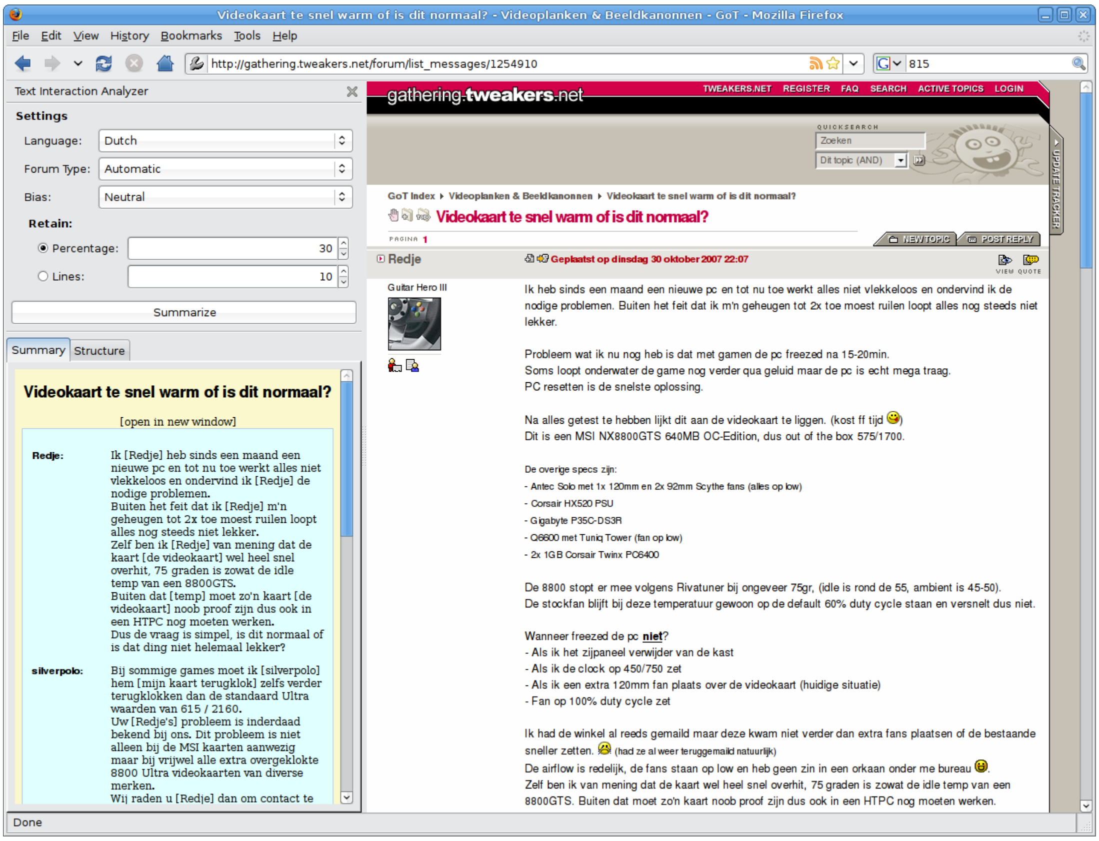
Figure 7.3: *Command-line interface (summary text is in Dutch).*

- Forum Type: This determines what forum scraper is used. For registered URL's this is set automatically, for others URL's this is a mandatory argument (default: dependent on the URL).

Two primary interfaces to the system were developed. The first, and most intensively used, interface is a simple command-line script shown in figure 7.3. The second is actually a HTTP webserver that treats a passed URL as a system query.

This second interface allows for dynamic application: as stand-alone website, part of an existing website, part of a rich Internet application or part of the webbrowser. For demonstration purposes we integrated our prototype into the Firefox[6] webbrowser as a sidebar using the webserver interface. This can be seen in figure 7.4.

[6] http://www.getfirefox.com



Videokaart te snel warm of is dit normaal? - Videoplanken & Beeldkanonnen - GoT - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://gathering.tweakers.net/forum/list_messages/1254910
815
Text Interaction Analyzer
Summarize
Summary
Structure
gathering.tweakers.net
TWEAKERS.NET REGISTER FAQ SEARCH ACTIVE TOPICS LOGIN
QUICKSEARCH
Zoeken
Dit topic (AND)
UPDATE TRACKER
NEW TOPIC
POST REPLY
VIEW QUOTE
Done

Settings

Language: Dutch

Forum Type: Automatic

Bias: Neutral

Retain:

● Percentage: 30

○ Lines: 10

Videokaart te snel warm of is dit normaal?

[open in new window]

Redje: Ik [Redje] heb sinds een maand een nieuwe pc en tot nu toe werkt alles niet vlekkeloos en ondervind ik [Redje] de nodige problemen.
Buiten het feit dat ik [Redje] m'n geheugen tot 2x toe moest ruilen loopt alles nog steeds niet lekker.
Zelf ben ik [Redje] van mening dat de kaart [de videokaart] wel heel snel overhit, 75 graden is zowat de idle temp van een 8800GTS.
Buiten dat [temp] moet zo'n kaart [de videokaart] noob proof zijn dus ook in een HTPC nog moeten werken.
Dus de vraag is simpel, is dit normaal of is dat ding niet helemaal lekker?

silverpolo: Bij sommige games moet ik [silverpolo] hem [mijn kaart terugklok] zelfs verder terugklokken dan de standaard Ultra waarden van 615 / 2160.
Uw [Redje's] probleem is inderdaad bekend bij ons. Dit probleem is niet alleen bij de MSI kaarten aanwezig maar bij vrijwel alle extra overgeklokte 8800 Ultra videokaarten van diverse merken.
Wij raden u [Redje] dan om contact te

GoT Index ▸ Videoplanken & Beeldkanonnen ▸ Videokaart te snel warm of is dit normaal?

Videokaart te snel warm of is dit normaal?

PAGINA 1

Redje Geplaatst op dinsdag 30 oktober 2007 22:07

Guitar Hero III

Ik heb sinds een maand een nieuwe pc en tot nu toe werkt alles niet vlekkeloos en ondervind ik de nodige problemen. Buiten het feit dat ik m'n geheugen tot 2x toe moest ruilen loopt alles nog steeds niet lekker.

Probleem wat ik nu nog heb is dat met gamen de pc freezed na 15-20min.
Soms loopt onderwater de game nog verder qua geluid maar de pc is echt mega traag.
PC resetten is de snelste oplossing.

Na alles getest te hebben lijkt dit aan de videokaart te liggen. (kost ff tijd 😋)
Dit is een MSI NX8800GTS 640MB OC-Edition, dus out of the box 575/1700.

De overige specs zijn:
- Antec Solo met 1x 120mm en 2x 92mm Scythe fans (alles op low)
- Corsair HX520 PSU
- Gigabyte P35C-DS3R
- Q6600 met Tuniq Tower (fan op low)
- 2x 1GB Corsair Twinx PC6400

De 8800 stopt er mee volgens Rivatuner bij ongeveer 75gr, (idle is rond de 55, ambient is 45-50).
De stockfan blijft bij deze temperatuur gewoon op de default 60% duty cycle staan en versnelt dus niet.

Wanneer freezed de pc **niet**?
- Als ik het zijpaneel verwijder van de kast
- Als ik de clock op 450/750 zet
- Als ik een extra 120mm fan plaats over de videokaart (huidige situatie)
- Fan op 100% duty cycle zet

Ik had de winkel al reeds gemaild maar deze kwam niet verder dan extra fans plaatsen of de bestaande sneller zetten. ☹ (had ze al weer teruggemaild natuurlijk)
De airflow is redelijk, de fans staan op low en heb geen zin in een orkaan onder me bureau 😃.
Zelf ben ik van mening dat de kaart wel heel snel overhit, 75 graden is zowat de idle temp van een 8800GTS. Buiten dat moet zo'n kaart noob proof zijn dus ook in een HTPC nog moeten werken.

Figure 7.4: *Firefox sidebar interface (summary and webpage texts are in Dutch).*

Chapter 8

Prototype Evaluation

'Statistics; The only science that enables different experts using the same figures to draw different conclusions.'

(Evan Esar)

It would be very useful and interesting to see how people judge the output of the system. This chapter is devoted specifically to describing the evaluation performed and discussing the results.

Note that evaluation of individual technologies can be found in the section of the technologies themselves. Concrete results of those evaluations are usually included as an appendix.

8.1 Design

As has been detailed in section 6.2 it is quite hard to evaluate a summarization system like this with synthetic metrics. Hence we have conducted a user evaluation. The main purpose of this evaluation was to let users judge the output of the system. This section roughly treats the design of the evaluation, for a complete overview of all questions in the evaluation see Appendix D.

Confronting the user with a thread and the summarizer output did not seem like a good approach. We decided it would be better if users were first actively engaged with the thread content. This would provide him/her with a clear image of what the discussion is about. Therefore, the users first had to make some assignments with respect to the thread content. Keep in mind that comparisons between human crafted and machine generated summaries are not central to this research. Priming the participants so that they were able to better judge the summaries was the main intent.

The evaluation consisted of the following steps:

1. *Introductory screen*
   Informed participants of the duration of the test (30 minutes) and navigation controls. Those who had no experience with on-line fora were asked to *refrain* from participation. Following were several questions aimed at finding the exact competence level of the respondent.

2. *First discussion (part 1)*
   This screen contained only assignments. Participants were confronted with a real (*Problem-Solution* type and neutral bias) discussion. They were asked to arrange the five most important messages in order of importance and to create their own extractive summary of the discussion by pasting sentences into a textbox.

3. *First discussion (part 2)*
   Participants were shown the same discussion again. This time with an additional automatically generated summary. Users were asked to rate various aspects of this generated summary.

4. *Second discussion (part 1)* (figure 8.1)
   Same set-up as for step 2 with a different discussion. This time it was a discussion of the *Statement-Discussion* type, also with neutral bias.

5. *Second discussion (part 2)* (figure 8.2)
   Same as for step 3.

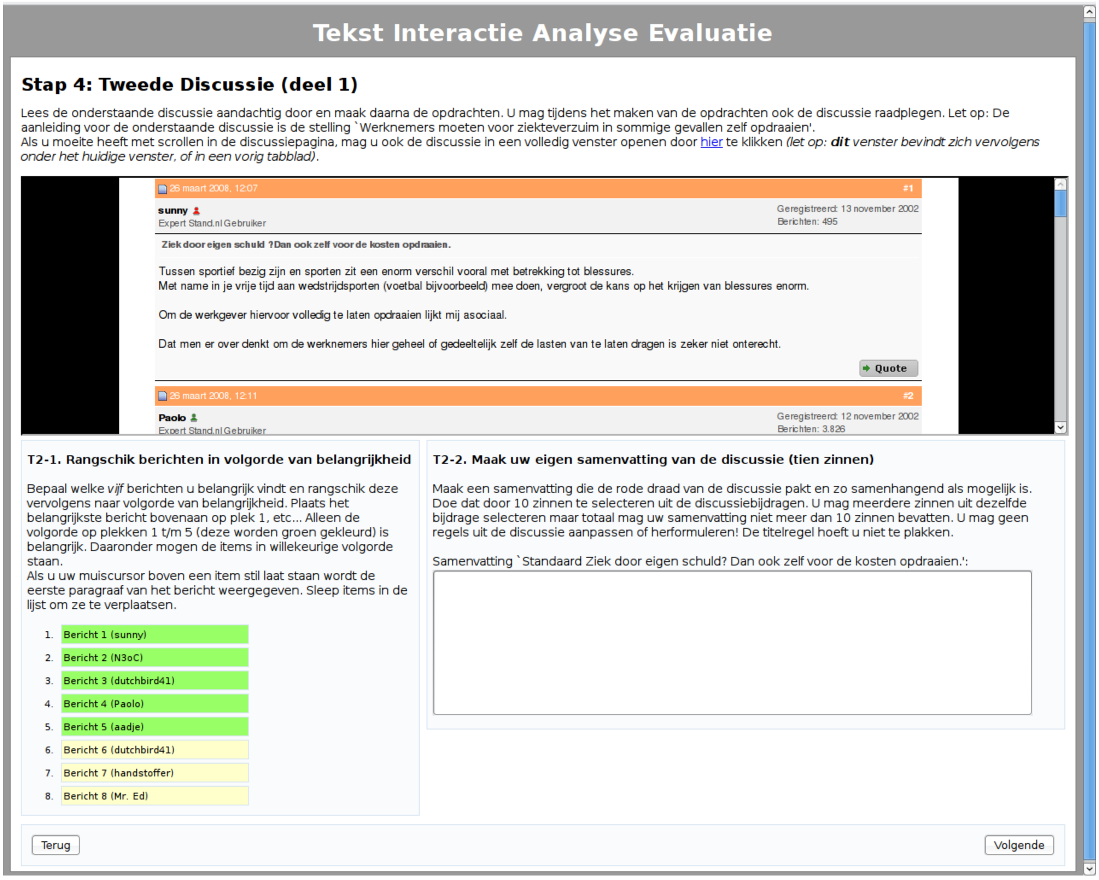
6. *Your opinion*
   Users were asked various questions regarding the usefulness of an automatic summarizer and of several included features.

7. *General information*
   Simple demographic attributes of the participants were asked. A box for entering additional comments was provided.

8. *Processing*
   This step actually stored all the information the participant entered in a database. They were confronted with a success message (provided everything was successfully stored).

Both of the selected discussions were short. The first consisted of 12 messages and 100 unique sentences, the second of 8 messages and 30 unique sentences. Using larger threads was considered infeasible for this small scale explorative evaluation.

Respondents were first asked to rank five messages in the thread in order of importance and to create an extractive summary manually. This is shown in figure 8.1. The top of the screen shows the thread in a scrollable frame, although participants also had the option to open the discussion in a separate window. The ordering assignment is shown in the bottom left. The basic representation was a numbered list of messages including author names. Respondents were able to click and drag items in this list to arrange them. Only the order of the first five (green colored) messages was of importance. Hovering the mouse over an item would display a tooltip with the first paragraph of the message as a memory aid. The bottom right shows



Tekst Interactie Analyse Evaluatie

Stap 4: Tweede Discussie (deel 1)

Lees de onderstaande discussie aandachtig door en maak daarna de opdrachten. U mag tijdens het maken van de opdrachten ook de discussie raadplegen. Let op: De aanleiding voor de onderstaande discussie is de stelling `Werknemers moeten voor ziekteverzuim in sommige gevallen zelf opdraaien'.
Als u moeite heeft met scrollen in de discussiepagina, mag u ook de discussie in een volledig venster openen door hier te klikken *(let op: **dit** venster bevindt zich vervolgens onder het huidige venster, of in een vorig tabblad)*.

26 maart 2008, 12:07 #1
sunny
Expert Stand.nl Gebruiker
Geregistreerd: 13 november 2002
Berichten: 495

Ziek door eigen schuld ?Dan ook zelf voor de kosten opdraaien.

Tussen sportief bezig zijn en sporten zit een enorm verschil vooral met betrekking tot blessures.
Met name in je vrije tijd aan wedstrijdsporten (voetbal bijvoorbeeld) mee doen, vergroot de kans op het krijgen van blessures enorm.

Om de werkgever hiervoor volledig te laten opdraaien lijkt mij asociaal.

Dat men er over denkt om de werknemers hier geheel of gedeeltelijk zelf de lasten van te laten dragen is zeker niet onterecht.

26 maart 2008, 12:11 #2
Paolo
Expert Stand.nl Gebruiker
Geregistreerd: 12 november 2002
Berichten: 3.826

T2-1. Rangschik berichten in volgorde van belangrijkheid

Bepaal welke *vijf* berichten u belangrijk vindt en rangschik deze vervolgens naar volgorde van belangrijkheid. Plaats het belangrijkste bericht bovenaan op plek 1, etc... Alleen de volgorde op plekken 1 t/m 5 (deze worden groen gekleurd) is belangrijk. Daaronder mogen de items in willekeurige volgorde staan.
Als u uw muiscursor boven een item stil laat staan wordt de eerste paragraaf van het bericht weergegeven. Sleep items in de lijst om ze te verplaatsen.

1. Bericht 1 (sunny)
2. Bericht 2 (N3oC)
3. Bericht 3 (dutchbird41)
4. Bericht 4 (Paolo)
5. Bericht 5 (aadje)
6. Bericht 6 (dutchbird41)
7. Bericht 7 (handstoffer)
8. Bericht 8 (Mr. Ed)

T2-2. Maak uw eigen samenvatting van de discussie (tien zinnen)

Maak een samenvatting die de rode draad van de discussie pakt en zo samenhangend als mogelijk is. Doe dat door 10 zinnen te selecteren uit de discussiebijdragen. U mag meerdere zinnen uit dezelfde bijdrage selecteren maar totaal mag uw samenvatting niet meer dan 10 zinnen bevatten. U mag geen regels uit de discussie aanpassen of herformuleren! De titelregel hoeft u niet te plakken.

Samenvatting `Standaard Ziek door eigen schuld? Dan ook zelf voor de kosten opdraaien.':

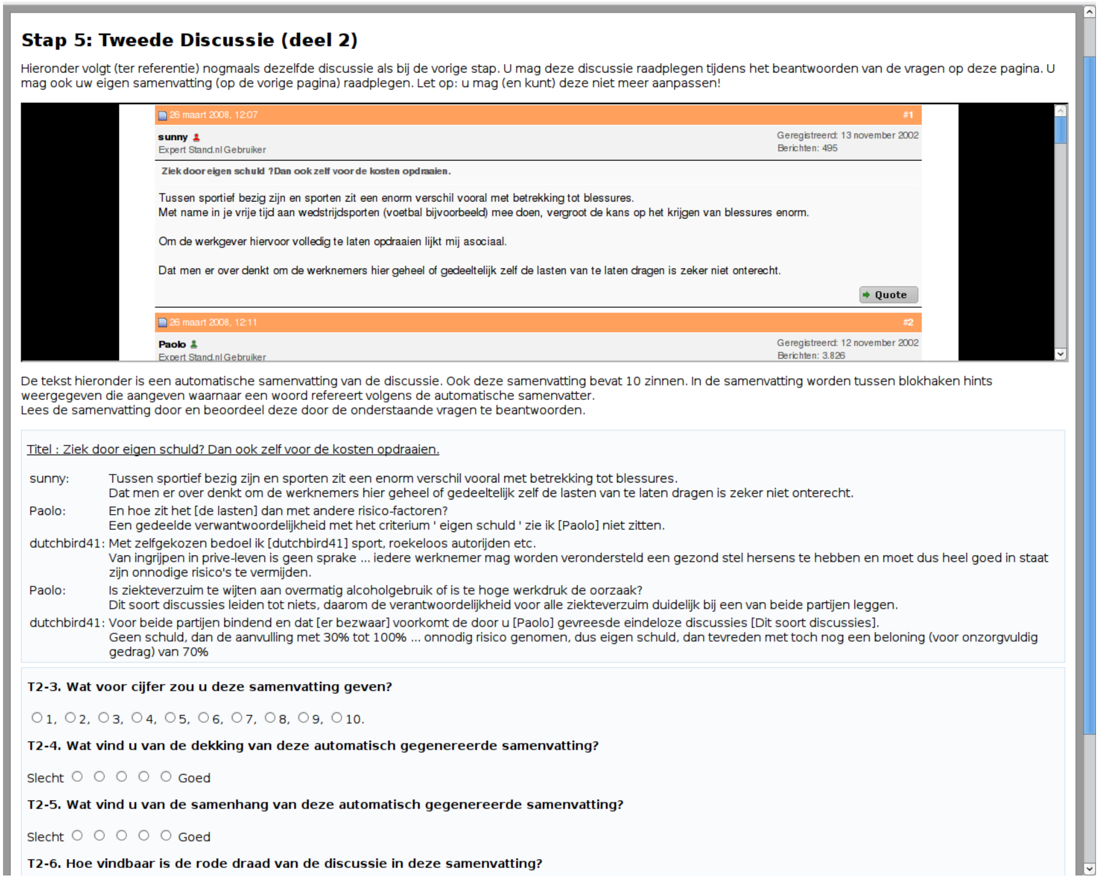
Figure 8.1: *Evaluation ordering and manual summary creation screen (texts are in Dutch)*

the textbox in which participants were able to construct their extractive summary by pasting sentences of the original discussion.

Selecting important sentences is no panacea. Since the discussions were already quite long, we did not want to strain the respondents and decided to ask for creation of fairly short summaries. The first discussion was allowed to consist of 12 sentences (12% of the original content and equivalent to the number of messages posted therein). The second discussion of 10 sentences (33$^1/_3$%). This nearly conforms to the lower bound and upper bounds for ‘useful’ summaries as explained in chapter 6. However, it is hard to directly compare these percentages in terms of information content due to the presence of repetition.

In the next step participants were presented with the automatically generated summary and asked to judge it. This can be seen in figure 8.2. Respondents were requested to assign a grade to the generated summary and to rate several other aspects such as the coverage and coherence of the text.

After the two discussions the participants could give their opinion regarding the usefulness of an automatic summarizer in general and various features (like the distinction between subjective



Stap 5: Tweede Discussie (deel 2)

Hieronder volgt (ter referentie) nogmaals dezelfde discussie als bij de vorige stap. U mag deze discussie raadplegen tijdens het beantwoorden van de vragen op deze pagina. U mag ook uw eigen samenvatting (op de vorige pagina) raadplegen. Let op: u mag (en kunt) deze niet meer aanpassen!

26 maart 2008, 12:07 #1

sunny
Expert Stand.nl Gebruiker

Geregistreerd: 13 november 2002
Berichten: 495

Ziek door eigen schuld ?Dan ook zelf voor de kosten opdraaien.

Tussen sportief bezig zijn en sporten zit een enorm verschil vooral met betrekking tot blessures.
Met name in je vrije tijd aan wedstrijdsporten (voetbal bijvoorbeeld) mee doen, vergroot de kans op het krijgen van blessures enorm.

Om de werkgever hiervoor volledig te laten opdraaien lijkt mij asociaal.

Dat men er over denkt om de werknemers hier geheel of gedeeltelijk zelf de lasten van te laten dragen is zeker niet onterecht.

26 maart 2008, 12:11 #2

Paolo
Expert Stand.nl Gebruiker

Geregistreerd: 12 november 2002
Berichten: 3.826

De tekst hieronder is een automatische samenvatting van de discussie. Ook deze samenvatting bevat 10 zinnen. In de samenvatting worden tussen blokhaken hints weergegeven die aangeven waarnaar een woord refereert volgens de automatische samenvatter.
Lees de samenvatting door en beoordeel deze door de onderstaande vragen te beantwoorden.

Titel : Ziek door eigen schuld? Dan ook zelf voor de kosten opdraaien.

sunny: Tussen sportief bezig zijn en sporten zit een enorm verschil vooral met betrekking tot blessures.
Dat men er over denkt om de werknemers hier geheel of gedeeltelijk zelf de lasten van te laten dragen is zeker niet onterecht.

Paolo: En hoe zit het [de lasten] dan met andere risico-factoren?
Een gedeelde verwantwoordelijkheid met het criterium ' eigen schuld ' zie ik [Paolo] niet zitten.

dutchbird41: Met zelfgekozen bedoel ik [dutchbird41] sport, roekeloos autorijden etc.
Van ingrijpen in prive-leven is geen sprake ... iedere werknemer mag worden verondersteld een gezond stel hersens te hebben en moet dus heel goed in staat zijn onnodige risico's te vermijden.

Paolo: Is ziekteverzuim te wijten aan overmatig alcoholgebruik of is te hoge werkdruk de oorzaak?
Dit soort discussies leiden tot niets, daarom de verantwoordelijkheid voor alle ziekteverzuim duidelijk bij een van beide partijen leggen.

dutchbird41: Voor beide partijen bindend en dat [er bezwaar] voorkomt de door u [Paolo] gevreesde eindeloze discussies [Dit soort discussies].
Geen schuld, dan de aanvulling met 30% tot 100% ... onnodig risico genomen, dus eigen schuld, dan tevreden met toch nog een beloning (voor onzorgvuldig gedrag) van 70%

T2-3. Wat voor cijfer zou u deze samenvatting geven?

○ 1, ○ 2, ○ 3, ○ 4, ○ 5, ○ 6, ○ 7, ○ 8, ○ 9, ○ 10.

T2-4. Wat vind u van de dekking van deze automatisch gegenereerde samenvatting?

Slecht ○ ○ ○ ○ ○ Goed

T2-5. Wat vind u van de samenhang van deze automatisch gegenereerde samenvatting?

Slecht ○ ○ ○ ○ ○ Goed

T2-6. Hoe vindbaar is de rode draad van de discussie in deze samenvatting?

Figure 8.2: *Evaluation summary judgement screen of the second discussion (texts are in Dutch). Note: not all questions are displayed, for a complete overview see section 8.2.4 and appendix D.*

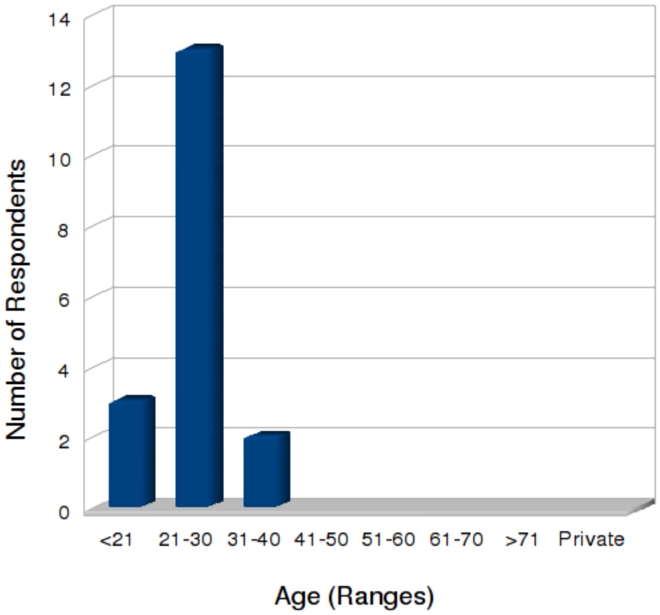


Figure 8.3: *(A2) Respondents' age.*

and objective). This was followed by a screen with some general demographic questions that closed the evaluation.

8.2 Results and Discussion

Detailed tallied results of the evaluation can be found in appendix E. This section presents and discusses the results by using graphs. Question identifiers are included in the graph captions.

In several places statistical tests are performed. The reader should be aware of the fact that these are based on 18 samples, which makes it difficult to draw conclusions, even for the statistically significant results.

8.2.1 Respondents

18 respondents participated in our evaluation. The first (1) and last step (7) of the evaluation largely focus on the users' background and fora usage experience. 72% of the respondents was male, 28% was female. Figure 8.3 shows that the older age range is underrepresented. Nearly all respondents are in the 21-30 age range.

Since this research aims to develop assistive technologies for fora users, it is important to know something about the usage experience of the evaluation respondents. An aspect of this is usage frequency, shown in figure 8.4. We can see that half of the respondents regularly (daily or

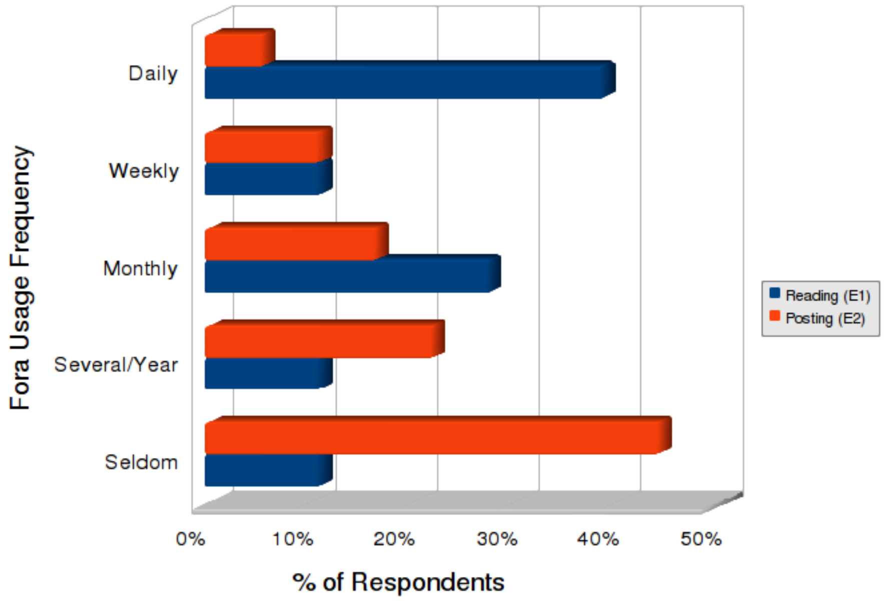


Figure 8.4: *(E1, E2) Discussion fora reading and posting experience.*

weekly) reads fora, but only 17% frequently posts. Figure 8.5 illustrates the number of fora that are regularly used. For almost three quarter of the respondents this is between one and four fora. Thus we can speak of relatively experienced participants. There is a positive correlation of 0.36 between E2 and E3 and 0.18 between E1 and E3. This suggests that people who frequently post, also use more different fora compared to those that only read discussions.

Figure 8.6 shows us that most respondents consider themselves to be quite agile when it comes to searching on the Internet, but less so on Internet fora. This supports the idea that there is a difference between these tasks.

Since the discussions used for the evaluation were in Dutch, experience with this language was definitely important for understanding them. Hence, we wanted to know the competency level of respondents. 89% of the participants had Dutch as native language. Figure 8.7 shows that 95% indicated having a high (4, 5) competency with regard to the language.

8.2.2 Message Ordering in the Discussions

Participants were asked to order the messages by importance. An ordering task like this is fairly difficult to perform especially for the bottom ranks. It is also precisely the bottom ranks that are less significant for the summarization process. Hence, we have asked respondents to pick the (for the sake of consistency) five most important messages from both discussions and rank them according to their importance.

The results of this manual ranking for the first discussion are shown in table 8.1. We see

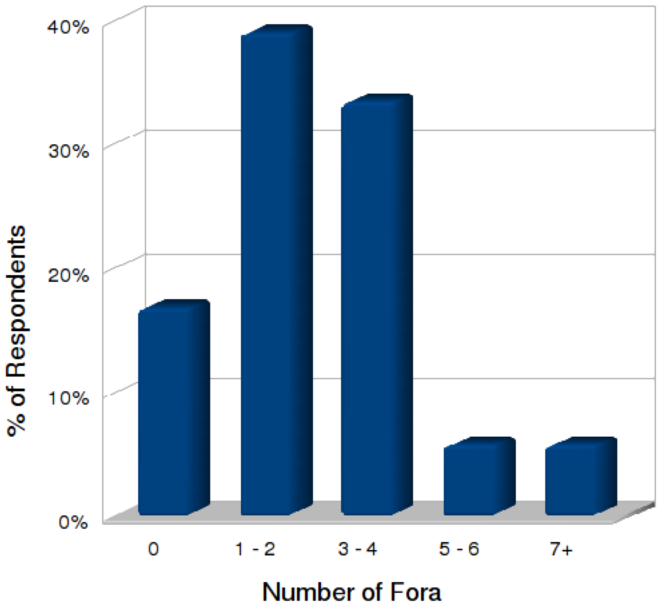


Figure 8.5: *(E3) Discussion fora regular usage experience.*

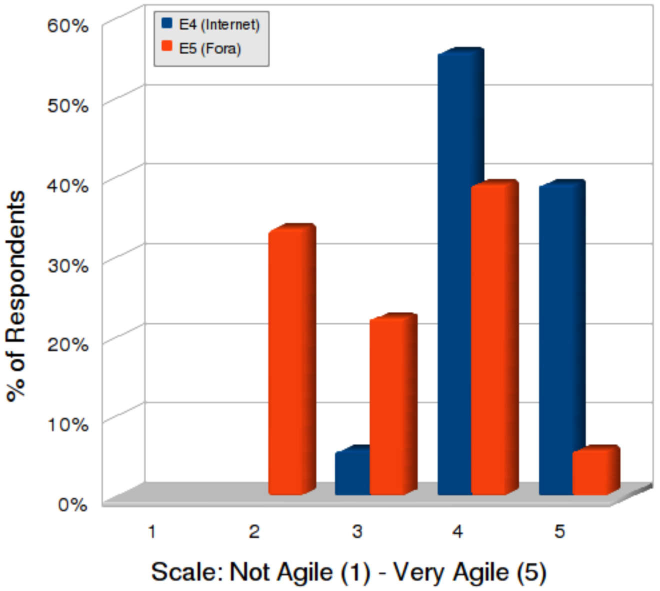


Figure 8.6: *(E4, E5) Fora and Internet searching experience.*

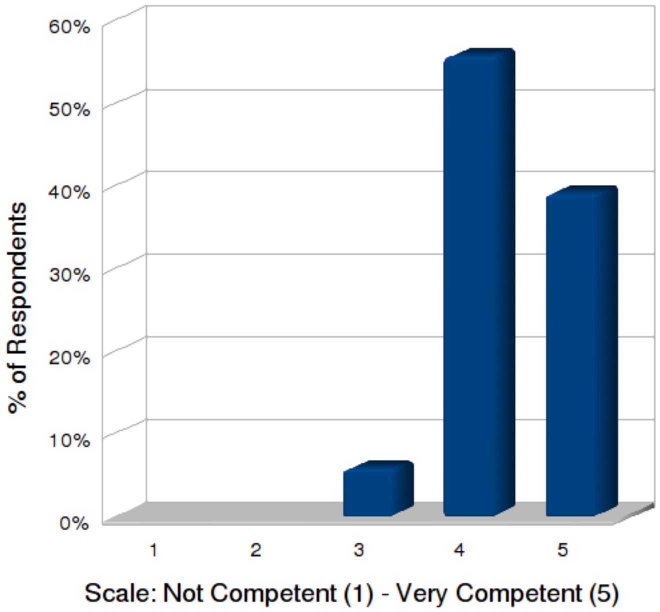


Figure 8.7: *(E7) Dutch language competency.*

that people are relatively sure about the first message. The spread increases towards the end, although the fourth rank is an odd exception with a strong preference there for the fourth message in the discussion. This justifies the choice for requesting only the first five message to be ranked and supports our intuition regarding the task for the lower ranks. Table 8.2 shows the results for the machine. It correctly identifies 1, 11 and 12. However, it misses 2 and 4 completely and instead opts for messages 5 and 8. This is caused by these clearly being tied to preceding messages by usage of explicit quotes and because they are more often referenced themselves. The machine still does score 3 out of 5 correct here, even though not in the same rank order as human annotation.

For the second discussion the same task was performed. Human annotation results are shown in table 8.3. We see that the rank is consistent with the message number for the first four messages. This pattern is broken at the end however, where the sixth message takes precedence over the fifth. The machine prioritisation can be seen in table 8.4. While the rank of message number 1 (and as a result of 2 and 3 as well) is incorrect, the same messages are identified as by the human annotation. The reason for the difference in ranking is twofold. First, the second and third messages are cited by other messages (the first is not). Second, the authors of the second and third messages are more prolific in the thread. In fact, there is no activity of the initial poster beyond the first post. These two aspects are captured by the Positional Message Relevance (PMR) and Participation-Talkativity (PT) factor, discussed in chapter 2.

Table 8.1: *(T1-1) First discussion human message prioritisation. The cells show the number of times each message (denoted by columns) was mentioned at a particular rank (rows 1 through 5). Zeros have been omitted to ease reading. The most chosen message for each rank is colourshaded in the table. According to human annotation the five best message by rank are: 1, 12, 11, 4 and 2.*

Rank	Message Number											
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
1	9		1							1	4	3
2	1	1		3					1		5	7
3	3	2	3	1		1	2			1	5	3
4				8			1			1	2	3
5		7	1	2			3	1		2	1	1

Table 8.2: *First discussion machine message prioritisation. The formulas used for calculation are equivalent to those in section 6.4. The $f_{msgweight}$ is derived from the positional message relevance f_{pmr} and the participation-talkativity factor f_{pt} (there is 0.01 rounding error here due to the decimal places, which we ignore). According to the machine the five best messages by rank are: 1, 5, 8, 11 and 12.*

	Message Number											
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
f_{pmr}	0.40	0.00	0.00	0.00	0.21	0.00	0.00	0.18	0.05	0.05	0.07	0.04
f_{pt}	0.07	0.06	0.05	0.08	0.06	0.06	0.06	0.07	0.06	0.05	0.16	0.22
$f_{msgweight}$	0.29	0.02	0.02	0.03	0.16	0.02	0.02	0.14	0.05	0.05	0.10	0.10

Table 8.3: *(T2-1) Second discussion human message prioritisation. Set-up the same as in table 8.1. According to human annotation the five best message are by rank: 1, 2, 3, 4 and 6.*

Rank	Message Number							
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*
1	11	4	3					
2		8	6	1		3		
3	2	2	6	5		2		1
4	2	2	1	9	1	3		
5	2	1	1	1	3	6		4

Table 8.4: *Second discussion machine message prioritisation. Set-up the same as in table 8.2. According to the machine the best five messages by rank are: 2, 3, 1, 4, 6 (all of these relatively close to each other).*

	Message Number							
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*
f_{pmr}	0.19	0.22	0.20	0.18	0.00	0.14	0.00	0.07
f_{pt}	0.11	0.14	0.16	0.14	0.08	0.16	0.12	0.11
$f_{msgweight}$	0.16	0.19	0.18	0.16	0.03	0.14	0.04	0.08

8.2.3 Sentence Selection in the Discussions

We asked respondents to create their own summary of each of the discussions. They needed to select 12 lines for the first discussion and 10 lines for the second (the same amounts as for the automatically generated summaries). During the first test phase of the evaluation it was not clear to some people that the amount of lines was not an upper bound. In response the instructions were corrected.

As mentioned before, the main reason for asking participants to create their own summary was to enable them to better judge the machine generated summary. The hand-created summaries themselves are not as central to this research as the grading of the machine summary. Nevertheless, this step does deliver interesting data. This is the reason that we decided to analyse and discuss the results.

We observed a number of interesting phenomena in the hand-made summaries:

- Many participants ordered the sentences in a way which was not consistent with the ordering in the original thread. This was done to ease reading. Our automatic summarizer does not do this. This makes a case for investigating whether re-ordering of sentences (especially question/answer pairs) is a useful addition.
- Some people changed the original sentences even though they were explicitly asked not to do this. This included changes like deleting or adding a dot at the end of a sentence or removing a smiley from a sentence.
- On submission we explicitly checked the number of sentences that was pasted in the textbox. This was done with a simple screen-line count check. Some people managed to get past this. First, as mentioned in the introduction some people used less sentences (and padded their summary with empty screen-lines). Some also went over the twelve sentence limit by, presumably accidentally, putting two sentences on one screen-line. This was also caused by the fact that screen-lines in some posts contained two sentences which apparantly confused people into assuming that each screen-line was *equivalent* to a sentence.

For the first discussion only about half of the summaries (10) contained precisely twelve lines due to these issues. For the requested ten-line summaries of the second discussion the picture is similar (11 correct). This opts for a ‘better’ sentence selection interface. However, when leaving out summaries that are not precisely as long as requested, we still end up with the same selection of sentences as shown in the tables in this section. Although their ordering does differ.

The twelve most selected sentences for the first thread, taking into account all responses, is shown in table 8.5. In total respondents picked 60 unique sentences for inclusion in the first discussion (60% of the sentences in the thread), 11 of which appeared in only one of the summaries. The table shows that many people agree on a few important sentences. It is obvious what the central question of the author of the initial post is, since this is also the most frequently included sentence. Besides this, messages 2, 4 and 11 are well represented. With respect to the previous section the following can be stated: if a message has a high Positional Message Relevance (PMR) (e.g. message 1) this not always directly translates into a higher amount of selected sentences. This makes a case against using PMR alone, supporting the addition of the Participation-Talkativity factor.

Table 8.6 shows what sentences were selected by the automatic summarizer (in posting order) and the number of times these lines were included in one of the human-made summaries. It can be concluded that the discussion between RedBoll and wd200 (messages 5 and 8) is actually perceived as an unimportant sidebranch of the main topic. This suggests research into subtopic detection would be wise.

Results for the second discussion are shown in table 8.7. Respondents picked 30 unique sentences for inclusion (all of the sentences in the thread), 6 of which appeared in only one of the summaries. Interestingly, the contributions made by the initial posting author are not found at the top of the table. The contributions made by user dutchbird41 are apparantly judged as being the most relevant. Note that this user is prolific in the thread. This is reflected in a high Participation-Talkativity factor.

Table 8.8 shows the automatically selected sentences. Even though the summary does not contain the three most selected sentences, all of the selected sentences were included in some human summary. Of course this is also due to the fact that the machine correctly identified the most important messages in the previous step. This already restricts the sentence selection space and, as a result, decreases the probability of ‘incorrect’ selection. Additionally, this thread is also shorter, there is a chance of 33% (indeed, equal to the summary size relative to the size of the entire thread) of selecting the right sentences.

8.2.4 Judgement of the Automatic Summaries

The most important component of the evaluation was the judgements of the automatically generated summaries. The summaries themselves can be found in appendix D. Recall that the first discussion was of the *Problem-Solution* type, whereas the second was of *Statement-Discussion* type. The summary of the first discussion was shorter, percent-wise, but also contained more repetition of arguments. Also since the first discussion type is generally more concrete, we expected the automatic summary of the first discussion to be graded higher than the second. Nevertheless, figure 8.8 suggests the opposite. The first automatic discussion summary scores

Table 8.5: *Sentences (12) of the first discussion that were included in the summaries the most (as indicated by all 18 respondents). The first columns shows the message number (counted from the top starting at one), author name and the inclusion counts (descending sort order).*

Msg	*User*	*Count*	*Sentence*
1	wd200	12	Mijn vraag is hoeveel werk is er te vinden voor een ict beheerder op MBO4 niveau ?
11	CyberTijn	11	Ieder fatsoenlijk bedrijf geeft z'n mensen de mogelijkheden om trainingen te volgen en certificaten te halen om zijn / haar werk beter te kunnen doen en om door te stromen naar een hogere functie.
4	m0nk	10	Momenteel is de markt goed voor de werkzoekende.
2	N3oC	10	Voor vergoedingen, salarissen en secundaire arbeidsvoorwaarden kijk eens in "Wat verdient een ICTer gemiddeld? (deel 7)".
7	Aikon	9	Salaris moet je rond de 1600~1800 denken.
4	m0nk	8	Als je bij een detacheerder in dienst gaat zal je hoogst waarschijnlijk bij een opdrachtgever neergezet worden (afhankelijk van je opleiding en ervaring).
4	m0nk	7	Er zijn trouwens momenteel zat vacatures voor junior systeembeheerders/helpdesk/werkplekbeheer.
11	CyberTijn	7	tussen de 1600 en 1700 is niet onrealistisch
12	hypz	7	Staar je niet blind op de vacatures om monsterboard waar ze vaak een hele lijst met software kennis en veel ervaring eisen, maar plaats gewoon je cv eens op monsterboard en houd je telefoon ff een dagje aan.
11	CyberTijn	6	Als je in de uitvoerende tak van ICT gaat zitten (dus zonder het woord "manager" in je functietitel) zul je altijd door moeten blijven leren.
9	Red Boll	5	Een HBO diploma in je achterzak geeft je wel meer mogelijkheden, zeker later in je carriere...
12	hypz	5	Inderdaad, via detacheerders ben je zo aan het werk, en het is niet verkeerd om mee te beginnen al zal het salaris daar niet zo hoog zijn.

Table 8.6: *Sentences selected by the automatic summarizer sorted in posting order. The third column shows the number of times a certain sentence was included in one of the 18 human made summaries.*

Msg	*User*	*Count*	*Sentence*
1	wd200	2	Ik heb zelf vorig jaar MBO4 ict beheerder afgerond.
		1	Ik ben wel door aan het studeren om het hbo(HTS-A) maar dit gaat niet zo lekker.
		12	Mijn vraag is hoeveel werk is er te vinden voor een ict beheerder op MBO4 niveau?
		2	wie is hier gaan werken na het mbo en hoe bevalt het?
5	RedBoll	0	Vraagje: Welke certificeringen heb je nu eigenlijk behaald in die 4(?) *jaar?*
		0	Deze opleiding is "nog van voor mijn tijd".
8	wd200	0	Geen certi.
		0	Ja itil en ecdl op school afgesloten als vak zijnde
11	CyberTijn	3	Maak je geen zorgen, een baan vinden in de ICT met alleen MBO is een eitje
		6	Als je in de uitvoerende tak van ICT gaat zitten (dus zonder het woord 'manager' in je functietitel) zul je altijd door moeten blijven leren.
12	hypz	4	- wat voor werkgevers zijn mogelijk (peak, call2?)
		5	Inderdaad, via detacheerders ben je zo aan het werk, en het is niet verkeerd om mee te beginnen al zal het salaris daar niet zo hoog zijn.

Table 8.7: *Sentences (10) of the second discussion that were included in the summaries the most (as indicated by all 18 respondents). The first columns shows the message number (counted from the top starting at one), author name and the inclusion counts (descending sort order).*

Msg	*User*	*Count*	*Sentence*
3	dutchbird41	14	Wanneer er sprake is van een BEDRIJFS ongeval dan is natuurlijk de WERKGEVER verantwoordelijk ... de premie voor deze verzekering komt dus voor rekening van de baas!
6	dutchbird41	13	Denk dat niemand er bezwaar tegen zal maken als de "beslissing" over de oorzaak van het verzuim wordt bepaald door de (huis)arts.
3	dutchbird41	12	In alle andere gevallen lijkt mij dat de WERKNEMER de (zelf gekozen) risico's verzekert en daarvoor ook de premie betaalt.
2	Paolo	10	En hoe zit het dan met andere risico-factoren?
1	sunny	10	Met name in je vrije tijd aan wedstrijdsporten (voetbal bijvoorbeeld) mee doen, vergroot de kans op het krijgen van blessures enorm.
1	sunny	9	Dat men er over denkt om de werknemers hier geheel of gedeeltelijk zelf de lasten van te laten dragen is zeker niet onterecht.
4	Paolo	9	Dit soort discussies leiden tot niets, daarom de verantwoordelijkheid voor alle ziekteverzuim duidelijk bij een van beide partijen leggen.
4	Paolo	8	Maar u beseft toch ook wel dat er eindeloze discussies gaan ontstaan.
4	Paolo	8	Is ziekteverzuim te wijten aan overmatig alcoholgebruik of is te hoge werkdruk de oorzaak?
2	Paolo	8	Een gedeelde verwantwoordelijkheid met het criterium 'eigen schuld' zie ik niet zitten.

Table 8.8: *Sentences selected by the automatic summarizer sorted in posting order. The third column shows the number of times a certain sentence was included in one of the 18 human-made summaries.*

Msg	*User*	*Count*	*Sentence*
1	sunny	5	Tussen sportief bezig zijn en sporten zit een enorm verschil vooral met betrekking tot blessures.
		9	Dat men er over denkt om de werknemers hier geheel of gedeeltelijk zelf de lasten van te laten dragen is zeker niet onterecht.
2	Paolo	10	En hoe zit het dan met andere risico-factoren?
		8	Een gedeelde verwantwoordelijkheid met het criterium 'eigen schuld' zie ik niet zitten.
3	dutchbird41	5	Met zelfgekozen bedoel ik sport, roekeloos autorijden etc. Van ingrijpen in prive-leven is geen sprake ... iedere werknemer mag worden verondersteld een gezond stel hersens te hebben en moet dus heel goed in staat zijn onnodige risico's te vermijden.
4	Paolo	8	Is ziekteverzuim te wijten aan overmatig alcoholgebruik of is te hoge werkdruk de oorzaak?
		9	Dit soort discussies leiden tot niets, daarom de verantwoordelijkheid voor alle ziekteverzuim duidelijk bij een van beide partijen leggen.
6	dutchbird41	4	Voor beide partijen bindend en dat voorkomt de door u gevreesde eindeloze discussies.
		2	Geen schuld, dan de aanvulling met 30% tot 100% ... onnodig risico genomen, dus eigen schuld, dan tevreden met toch nog een beloning (voor onzorgvuldig gedrag) van 70%

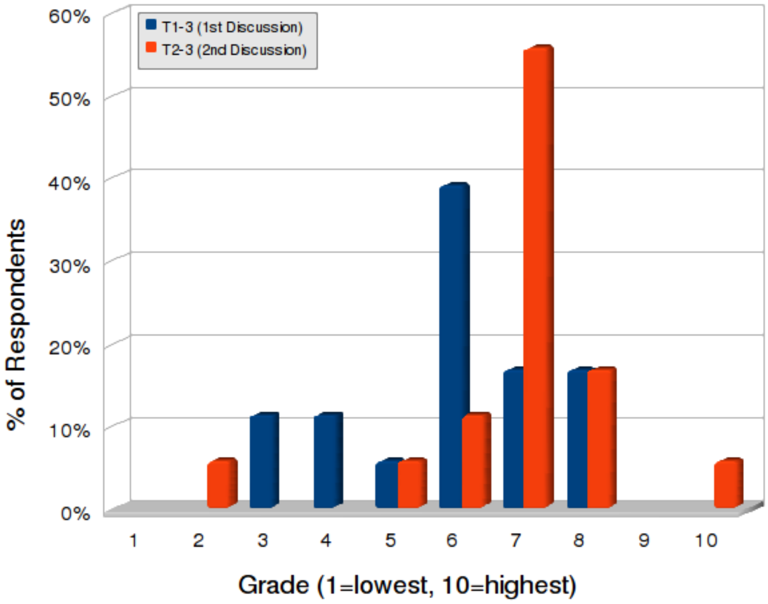


Figure 8.8: *(T1-3, T2-3) Summary grades.*

a 5.89 average (with a median of 6), the second a 6.83 (with a median of 7). The difference between the two is statistically significant according to a one-tailed paired[1] t-test (p=0.02).

This might also be tied to the message ranking evaluation result discussed earlier. Compared to the human rankings, the machine did better here for the second discussion (100% correct) than for the first (60%). An other reason is that the second discussion summary is larger (10 lines for 8 messages, 33%) compared to the first (12 lines for 12 messages, 12%). It suggests that a larger summary can yield a significant increase in perceived quality. However, since the thread types and content are different and because of repetitions in the threads, it is difficult to draw any hard conclusions. This phenomenon requires further investigation.

Regardless of the difference in expectation, the grades by themselves are surprisingly good. Consider that reading the discussion and making an extractive summary took about 10 to 15 minutes for each discussion for the human participants. The developed prototype does this in several seconds.

An important aspect of a summary is the coverage and coherence described in chapter 6. Coverage is shown in figure 8.9. We see that the first discussion summary is rated above average by only 33% of the respondents, while the second discussion summary is rated likewise for 67%. This shows a consistent pattern with the grading.

However, the coherence of both summaries, see figure 8.10), is much closer. Coherence of the first summary is rated slightly higher. In both cases the coherence is perceived as average or

[1] Even though these are different discussions, the grades provided by the respondents are still paired. As such, a paired test was used.

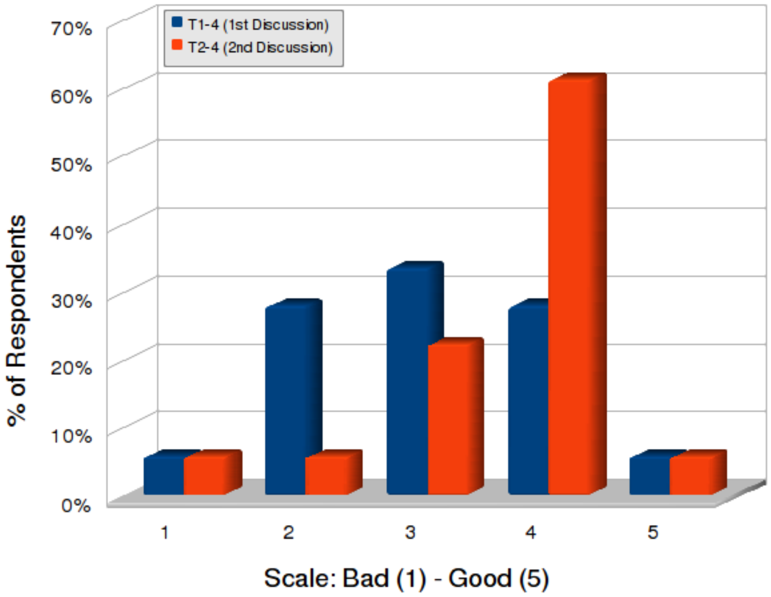


Figure 8.9: *(T1-4, T2-4) Summary coverage.*

higher by over 80% of the respondents.

We also wondered how useful the summaries were for assisting in a task. For the first discussion the task was envisioned as finding the question and answer, for the second discussion it was finding the main focus of the discussion. Figure 8.11 shows that utility for both tasks is rated average or higher by over 70% of the respondents, but the utility of the summary for finding the question and answer is rated higher.

Finally, the usefulness of including anaphoric references within the summary was tested. Users were able to judge this for both summaries independently. Figure 8.12 shows the results. For both they are judged as average or higher by 70% of the respondents. However, their presence is appreciated more in the summary of the first discussion.

Based on this we would expect there to be some positive correlation between the coherence of the summaries (T1-5, T2-5) and the usefulness of the references (T1-7, T2-7). This correlation indeed exists although it is not strong. Between T1-5 and T1-7 the correlation is 0.29, between T2-5 and T2-7 it is 0.44.

8.2.5 Opinions

The usefulness of an automatic summarizer (question M1) is judged on a five point scale to be useful (4) or very useful (5) by 95% of the respondents (5% is indifferent). There is a positive correlation of 0.14 between the reading frequency of discussion (E1) and the usefulness (M1).

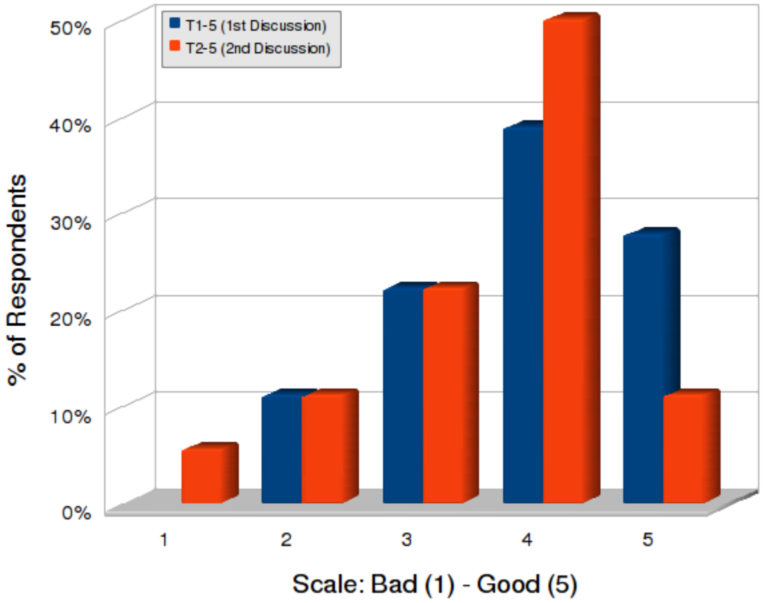


Figure 8.10: *(T1-5, T2-5) Summary coherence.*

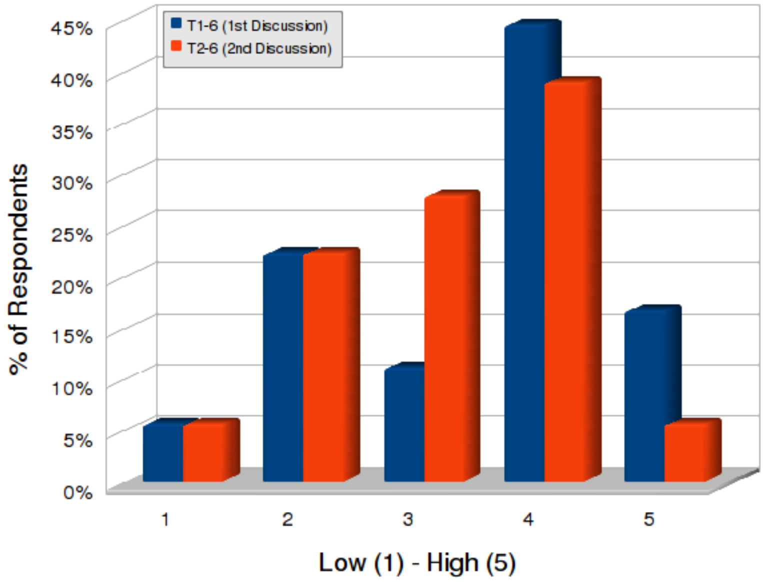


Figure 8.11: *(T1-6, T2-6) Summary task aid.*

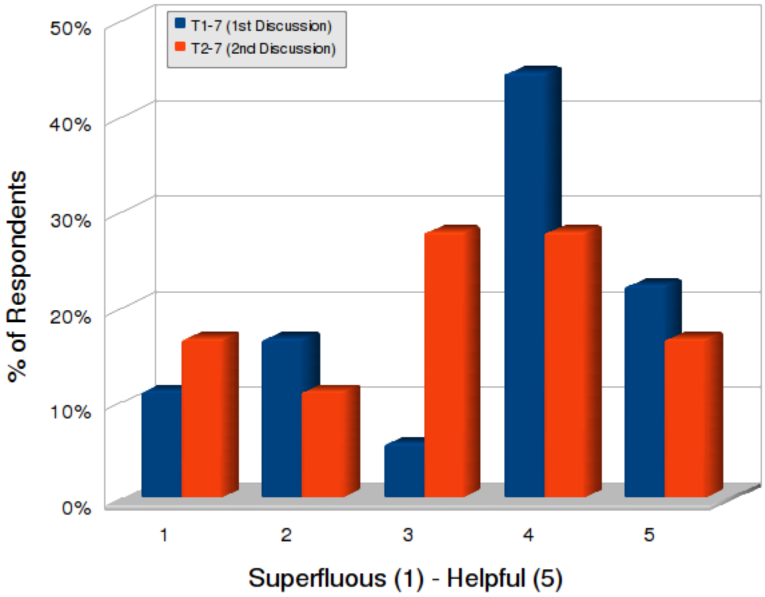


Figure 8.12: *(T1-7, T2-7) Usefulness of references.*

For the posting frequency (E2) and the usefulness (M1) the correlation is 0.40. Suggesting that people that post more frequently would also appreciate an automatic summarizer more.

Figure 8.13 suggests that most people would like an automatic summarizer to be built into a forum, a webbrowser plug-in being the second preference. Very few (5%) would like the summarizer to be a completely separate website.

Half of the respondents would use a summarizer every now and then. The other half would use it either regularly or always as can be seen in figure 8.14. This is promising and shows that there is genuine interest in the developed technology as well as a potential user base.

Finally, we asked the users whether they think an option for leaving out either objective or subjective information in a summary would be useful. This was done using a five-point scale. The result, figure 8.15, mostly shows an equal pattern for both. About half of the respondent would find such a feature useful, with a subjective preference being slightly favoured (~55% valued 4 or 5) over an objective one (50% valued 4 or 5).

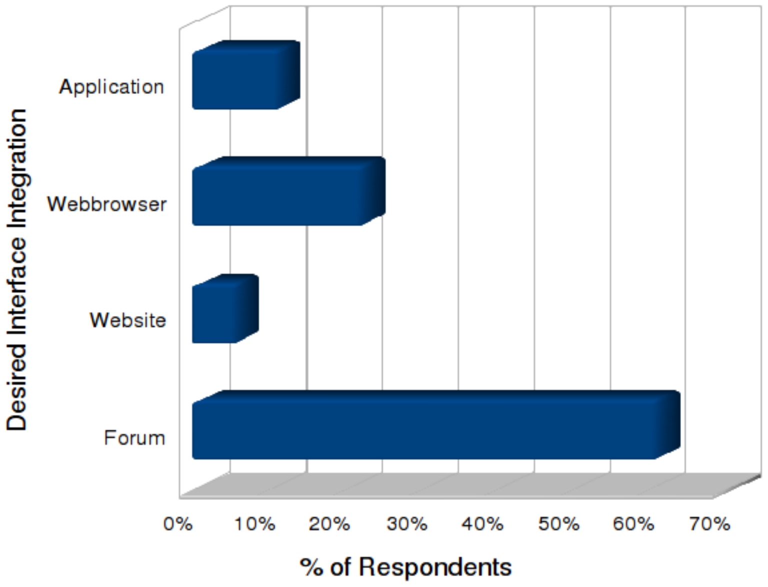


Figure 8.13: *(M2) Summarizer user interface integration.*

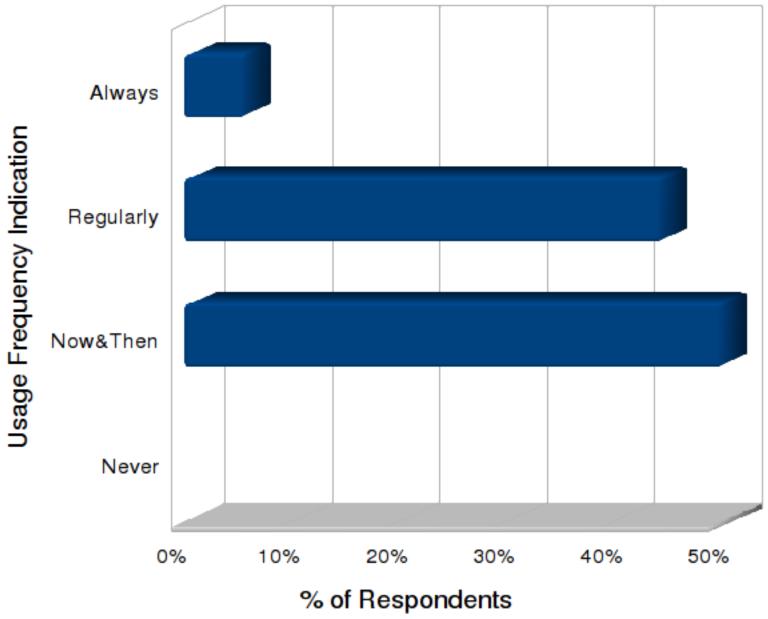


Figure 8.14: *(M3) Summarizer usage indication.*

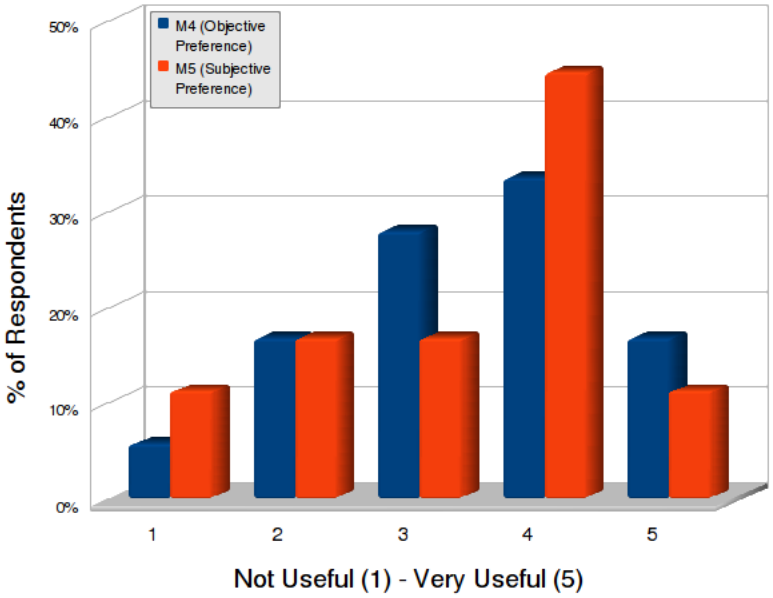


Figure 8.15: *(M4, M5) Usefulness of objective/subjective summarization preference.*

Chapter 9

Conclusion

'The only perfect science is hindsight.'

(Anonymous)

STARTING from the data, this research has forayed into many different aspects of Natural Language Processing (NLP). Many of our higher-level methods have been built upon a solid foundation laid by prior research. The developed prototype system has shown acceptable levels of performance.

Let us first look at our initial research questions and answer them:

1. How to automatically build summaries of threads?
2. What are structural characteristics of threads?
   And how can these characteristics be exploited for summarization purposes?
   a) What technologies and methods are necessary for this exploitation?
   b) How should these technologies and methods be combined?
3. What is the performance and usefulness of a thread summarization system?
   a) What is the performance of the individual components? (systematic evaluation)
   b) How do users rate the performance of the entire system? (user evaluation)
   c) What do different types of users think of the usefulness of such a system?
      i. Are automatically built summaries a useful addition to the search process?
      ii. Does the objective-subjective summarization preference add value?

The first question (1) is answered largely by this thesis as a whole. It shows one possible way in which such a system can be built by using a combination of heuristics and traditional NLP techologies (2a, b). Core parts of the current methodology rely on existing metadata in threads (2). We have developed heuristics for recognising the referential structure between messages in threads, for filtering messages and for recognising the importance of contributions of authors. Sentence selection was aided by finding sentence types and linking questions and answer sentences. We also made a brief detour into the field of subjectivity detection. Finally, everything was put together into one summarization algorithm. All ideas being implemented in a prototype system.

Parts of the prototype system could be evaluated on external datasets have shown to be relatively on-par with the state-of-the-art for Dutch (3a) performance-wise. Results of the evaluation of the prototype system show promise. Averaged, the grading of the generated summaries is well above a six on a ten point scale, which we interpret as being 'acceptable' (3b). Scores for coherence are high, suggesting that presenting the summary as a dialogue with anaphoric references expanded is a good approach. Coverage is rated less good, which suggests that further research into message and sentence selection is necessary. Respondents indicated that they find an automatic discussion summarizer useful (3c.i, 95%) and that a preference for subjective or objective content in the summary is a useful feature (3c.ii, 50-55%).

Nevertheless, there is plenty of room for improvements and several aspects deserve more attention. We investigated subjectivity, but found it remains a difficult and diffuse topic. While experimenting with various solutions, we had the feeling we were overfitting our models to data instead of finding real patterns. The solution we finally chose was the most robust we could think of, even though it is admittedly simplistic. Since subjectivity is highly context dependent this is really one area in which more semantic information regarding the context is necessary.

In our opinion, the ideas behind question-answer coupling are sound, but the implementation could be improved. Especially by looking closely at the data it would be possible to extend this functionality. We believe that the current approach is very dependent on the presence of proper domain data in the lists. Looking for a more generic, less domain data dependent, solution would be useful. It might also be worthwhile to look into selection of a possible best answer to a question and making this an additional factor in the message selection process.

Another area that of improvement is the selection of sentences. The fallback heuristic, which interleaves lines from the top and bottom of a post, is very basic. There is much more one could do here, especially by analysing the structure of paragraphs. Finding more sentence types to prioritise in different ways would also be useful.

The approach used for message filtering works relatively well and correctly identifies messages with poor formatting charactistics by using the Message Formatting Score (MFS). However, the usage of Average Word Length (AWL) and Average Sentence Length (ASL) is much less sensible and in hindsight was not such a fruitful detour. We observed that messages picked out by a deviations of these statistics are almost always also identified based on the MFS.

The anaphora resolution should be evaluated separately with the help of datasets. In the same vein as was done for partial parsing and named entity recognition. We conjecture that improving this aspect would also improve the coherence of the summaries.

Even though we have looked at many NLP technologies, we believe it is worthwhile to keep investigating other technologies. For example, those from the fields of question-answering and dialogue analysis.

We also think this research paints a clear picture of what is important in forum discussions. A good next step would be to investigate this further by creating annotations on larger datasets with many participants (a very modest attempt at human annotation was already done in the prototype evaluation). This can be used to refine the heuristics, like the Positional Message Relevance and Participation-Talkativity factor, and possibly to find new patterns. There are other different types of user evaluations that would be useful. For example, how to build an intuitive interface and how to integrate this into the browsing experience of the user.

These are only the first steps of a much longer journey. Nevertheless, we can conclude that creating automatic summaries of on-line discussions in Internet fora in the way presented in this thesis is a feasible idea. It can be used as an effective assistive technology for people that use fora.

Chapter 10

Future Work

‘Prediction is very difficult,
especially about the future.’

(Niels Bohr)

We think that there is a large amount of possible follow-up research to this thesis. Especially since the basic ideas behind it seem feasible. What follows are several of our ideas for future work.

- Subtopics (also called conversational branches) could be detected within a thread. This would make the end summary a collection of multiple subsummaries. It will be especially useful for (longer) threads in which the topic shifts, providing more coherent summaries. Zhou and Hovy already performed something similar for Internet Relay Chat discussions [99]. One of the threads used for evaluation showed that there are differences in importance between subtopics (see section 8.2.3). We feel that proper handling for this would be an important contributions to the current system.
- Adding handling for the sub-sentence level. This could be performed either on sentence parts or even on words such as presented by Witbrock and Mittal [97]. This would require proper sentence rewrite support.
- Using a speech act tagger might be a good approach improve question finding. This does require a corpus with annotated questions. Zechner uses a decision tree specifically for this task [98]. It may also be useful to look specifically at literature that treats the type of question-answer pairings in dialogues to improve this part of the system [3].
- To be able to better find question/answer pairs it might be worthwhile to look into the function of quoteblocks in this context. Specifically lines right below a quoteblock with a question are likely to be an answer. The difficulty is that some authors give very elaborate multi-paragraph answers. Where the gist of their argument is not always in the first (few) sentences below the quoteblock. However, by labeling the sentences by their function in

the message one could conceivably end up with a good model of where lines of significant importance with respect to the quoteblock can be found.

- Using word overlap for detecting question/answer pairs in a fashion that is less dependent on external data than the current approach. This would depend on a good overlap measure, the challenge being largely in paraphrasings of words in the original question sentence.
- Improvements to subjectivity detection could be made by building large and rich knowledge sources containing the effects of relations between words (as described in chapter 5).
- Subjectivity could be better integrated into the summarization algorithm. It could override even the normal ordering preference of sentence types. A good approach for this would need to be determined.
- During evaluation when participants were asked to make their own extractive summaries, many ordered the sentences in a way that was not consistent with the ordering in the original thread. This makes a case for investigating whether re-ordering of sentences based on the data (especially question/answer pairs) is a useful addition. This does raise coherence problems, although these could be mitigated by rewriting sentence parts.
- The system makes no distinction between Problem-Solution and Statement-Discussion threads. Because it is quite difficult to do automatically. Statement-Discussion threads tend to have more participants, more and longer messages with longer lines and words. However, none of these differences are significant enough for threads to be classified on the basis of these characteristics alone. Possible cue phrases could be of some help here. It could be helpful to have an extra system input parameter which specifies a preference for either a narrow summary (including more lines of fewer distinct authors) or a wide one (including fewer lines of many distinct authors). The current system tries to balance these two automatically expressed in the message weight. Such a parameter would offer extra flexibility and could be beneficial depending on the information needs of the user.
- The balance between the Positional Message Relevance (PMR) and the Participation-Talkativity (PT) factor is currently fixed to 2/3 and 1/3 respectively. Making this balance customizable is an option. This would enable favouring of the contributions of prolific authors in a thread. It would be a new input to the summarization process: a scale named ‘spread’. When set to highest (default) the balance as currently fixed would be used. When set to lowest the balance would be reversed with 1/3 for PMR and 2/3 for PT.
- Currently the PMR only takes into account the height of the thread graph. We believe the width and branching factor are also important. The fixed multiplication factor (of two) used in the PMR could be tuned automatically to the thread width or branching factor to yield a better and automatic fit to the input data.
- Looking at a site like Slashdot (which has moderated commenting) and see what the typical characteristics are of messages that are modded up or down.
- Summarizing exchange of mobile messages (SMS). This would be especially useful because of the relatively small screens that mobile devices typically have.

- Summarizing (Internet) phone conversations. This would require speech recognition and special handling for phenomena only seen in spoken dialogues.
- A good interface for letting evaluation respondents construct their own extractive summaries remains a difficult problem (see section 8.2.3). It would be interesting to investigate if, and how, a good interface for this problem could be developed. For example one in which participants can click and drag sentences from the discussion to a drop zone where they can be further arranged.
- We believe that a large-scale evaluation would be useful once the prototype system has been further developed. Possibly this could be done by setting up arrangements with discussion board owners.
- Currently the prototype system is optimized for threads of around 10 to 50 messages of medium size (approximation based on the reference threads: five paragraphs of about three lines each). It becomes progressively slower as the number of messages in the thread increases. There are many ways in which the prototype could be optimised. Some of the resource intensive operations we identifief are: anaphora resolution, the final summarization process, and anything that uses text overlap comparisons like finding the source of a citation.
- Adding support for Internet slang handling. There are some common abbreviations used on Internet fora that could be expanded into full representations, for example: AFAIK (As Far As I Know), IMHO (In My Humble Opinion). Expanding these automatically could make the summaries more readable. This expansion would be language dependent. A list of these abbreviations is already included in the prototype, but currently unused.

Bibliography

[1] State of the Art Report: Recognizing Subjective Content in Text and Conversation. AMI Consortium, November 2007.

[2] op den Akker, R., Hospers, M., Kroezen, E., Nijholt, A., and Lie, D. A Rule-based Reference Resolution Method for Dutch Discourse Analysis, 2002.

[3] op den Akker, R., and van Schooten, B. Follow-up Utterances in QA Dialogue. *Traitement Automatique des Langues 46*, 3 (2005), 181–206. Appeared in January 2007.

[4] Alterman, R. Understanding and Summarization. *Artificial Intelligence Review 5*, 4 (1991), 239–254.

[5] Asher, N., and Lascarides, A. *Logics of Conversation.* Cambridge University Press, Cambridge, UK, 2003.

[6] Barzilay, R., and Elhadad, M. Using Lexical Chains for Text Summarization. In *Proceedings of ACL* (Madrid, ESP, July 1997).

[7] Bird, S., Klein, E., and Loper, E. Natural Language Processing in Python. `http://nltk.sourceforge.net/index.php/Book`, 2008. (Draft Version 0.9.2).

[8] Bogers, T. Dutch Named Entity Recognition: Optimizing Features, Algorithms, and Output. Master's thesis, University of Tilburg, 2004.

[9] Bosma, W. E. Query-based Summarization using Rhetorical Structure Theory. In *Proceedings of CLIN* (Leiden, NL, December 2004), pp. 29–44.

[10] Bosma, W. E. *Discourse Oriented Summarization.* PhD thesis, University of Twente, 2008.

[11] Bouma, G., van Noord, G., and Malouf, R. Alpino: Wide-coverage Computational Analysis of Dutch. In *Proceedings of CLIN* (November 2000).

[12] Bruce, R. F., and Wiebe, J. M. Recognizing Subjectivity: A Case Study of Manual Tagging. *Natural Language Engineering 5* (1999), 187–205.

[13] Carenini, G., Ng, R. T., and Zhou, X. Summarizing Email Conversations with Clue Words. In *Proceedings of WWW* (Banff, AB, CA, May 2007), pp. 91–100.

[14] Choi, Y., Cardie, C., Riloff, E., and Patwardhan, S. Identifying Sources of Opinions with Conditional Random Fields and Extraction Patterns. In *Proceedings of HLT/EMNLP* (Vancouver, BC, CA, October 2005), pp. 355–362.

[15] Coleman, M., and Liau, T. L. A Computer Readability Formula Designed for Machine Scoring. *Journal of Applied Psychology 60*, 2 (1975), 283–284.

[16] Conrad, J. G., and Schilder, F. Opinion Mining in Legal Blogs. In *Proceedings of ICAIL* (Stanford, CA, USA, June 2007), pp. 231–236.

[17] Dalli, A., Yunqing, X., and Wilks, Y. Fasil Email Summarisation System. In *Proceedings of COLING* (Geneva, CHE, August 2004).

[18] van Deemter, K., and Kibble, R. On Coreferring: Coreference in MUC and Related Annotation Schemes. *Computational Linguistics 26*, 2 (2000), 629–637.

[19] DuBay, W. H. The Principles of Readability. Tech. rep., Impact Information, 2004.

[20] Erkan, G., and Radev, D. R. Lexrank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal of AI Research 22* (2004), 457–479.

[21] Evans, D. K., Klavans, J. L., and McKeown, K. R. Columbia Newsblaster: Multilingual News Summarization on the Web. In *Proceedings of HTL/NAACL* (Boston, MA, USA, May 2004).

[22] van Eynde, F. *Part of Speech Tagging en Lemmatisering van het Corpus Gesproken Nederlands.* Centre for Computerlinguistics, Catholic University of Leuven, February 2004.

[23] Farell, R. Summarizing Electronic Discourse. *International Journal of Intelligent Systems in Accounting, Finance & Management 11* (2002), 23–38.

[24] Farell, R., Fait-weather, P. G., and Snyder, K. Summarization of Discussion Groups. In *Proceedings of CIKM* (Atlanta, GA, USA, November 2001).

[25] Feng, D., Shaw, E., Kim, J., and Hovy, E. Learning to Detect Conversation Focus of Threaded Discussions. In *Proceedings of ACL* (New York, NY, USA, June 2006).

[26] Fournier, B. The Impossible Task of Dialog Analysis in Chatboxes. Capita Selecta Paper (HMI Group, University of Twente), 2007.

[27] Francis, W. N., and Kûcera, H. Brown Corpus Manual. `http://icame.uib.no/brown/bcm.html`, 1979.

[28] Ghose, A., Ipeirotis, P. G., and Sundararajan, A. Opinion Mining Using Econometrics: A Case Study on Reputation Systems. In *Proceedings of ACL* (Prague, CZ, June 2007), pp. 416–423.

[29] Goldstein, J., Mittal, V., Carbonell, J., and Kantrowitz, M. Multi-document Summarization by Sentence Extraction. In *Proceedings of ANLP/NAACL Workshop on Automatic Summarization* (Seattle, WA, USA, May 2000), vol. 4, pp. 40–48.

[30] Grishman, R. *The Oxford Handbook of Computational Linguistics: Information Extraction.* Oxford University Press, Oxford, UK, 2004, ch. 30, pp. 545–559.

[31] Hatzivassiloglou, V., and McKeown, K. R. Predicting the Semantic Orientation of Adjectives. In *Proceedings of ACL* (Madrid, ESP, July 1997), pp. 174–181.

[32] Hatzivassiloglou, V., and Wiebe, J. M. Effects of Adjective Orientation and Gradability on Sentence Subjectivity. In *Proceedings of COLING* (Saarbrücken, DE, July 2000), vol. 1, pp. 299–305.

[33] Hendrickx, I., Hoste, V., and Daelemans, W. Semantic and Syntactic features for Anaphora Resolution for Dutch. In *Proceedings of the CICLing* (Haifa, IL, February 2008), pp. 351–361.

[34] Hoeksema, J., and den Ouden, D.-B. Positief- en Negatief-polaire Bepalingen van Graad: Een Empirisch Onderzoek. *Tabu 35* (2005), 129–144.

[35] Hong, Y., and Hatzivassiloglou, V. Towards Answering Opinion Questions: Separating Facts from Opinions and Identifying the Polarity of Opinion Sentences. In *Proceedings of EMNLP* (Sapporo, JP, July 2003), pp. 129–136.

[36] Hoste, V., and Daelemans, W. Learning Dutch Coreference Resolution. In *Proceedings of CLIN'04* (December 2005).

[37] Hoste, V., and van den Bosch, A. A Modular Approach to Learning Dutch Coreference Resolution. In *Proceedings of WAR I* (Bergen, NO, 2007), pp. 51–75.

[38] Hovy, E. *The Oxford Handbook of Computational Linguistics: Text Summarization.* Oxford University Press, Oxford, UK, 2004, ch. 32, pp. 583–598.

[39] Hovy, E., Hermjakob, U., and Ravichandran, D. Qtargets used in Webclopedia. `http://www.isi.edu/natural-language/projects/webclopedia/Taxonomy`, 2002.

[40] Hovy, E., Lin, C.-Y., Zhou, L., and Fukumoto, J. Automated summarization evaluation with basic elements. In *Proceedings of LREC* (Genoa, IT, May 2006).

[41] Hu, M., Sun, A., and Lim, E.-P. Comments-Oriented Blog Summarization by Sentence Extraction. In *Proceedings of CIKM* (Lisboa, PT, November 2007).

[42] Johannessen, J. B. Negative Polarity Verbs in Norwegian. *Working papers in Scandinavian syntax 71* (2003), 33–73.

[43] Jurafsky, D., and Martin, J. H. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition.* Prentice-Hall, Upper Saddle River, NJ, USA, 2000.

[44] Kaji, N., and Kitsuregawa, M. Automatic Construction of Polarity-tagged Corpus from HTML Documents. In *Proceedings of COLING/ACL* (Sydney, AUS, July 2006), pp. 452–459.

[45] Kamps, J., Marx, M., Mokken, R. J., and de Rijke, M. Using WordNet to Measure Semantic Orientation of Adjectives. In *Proceedings of LREC* (Lisboa, PT, May 2004), vol. IV, pp. 1115–1118.

[46] Kazama, J., and Torisawa, K. Exploiting Wikipedia as External Knowledge for Named Entity Recognition. In *Proceedings of EMNLP-CoNLL* (Prague, CZ, June 2007), pp. 698–707.

[47] Kim, J., Chem, G., Feng, D., Shaw, E., and Hovy, E. Mining and Assessing Discussions on the Web through Speech Act Analysis. In *Proceedings of ISWC'06* (Athens, GA, USA, November 2006).

[48] Kim, J., Chem, G., Feng, D., Shaw, E., and Hovy, E. Modeling and Assessing Student Activities in On-Line Discussions. In *Proceedings of AAAI EDM'06* (Boston, MA, USA, July 2006).

[49] Kiss, T., and Strunk, J. Unsupervised Multilingual Sentence Boundary Detection. *Computational Linguistics 32*, 4 (2006), 485–525.

[50] Klaas, M. Toward Indicative Discussion Fora Summarization. Tech. Rep. UBC-CS TR-2005-04, University of Britisch Columbia, 2005.

[51] Kleinberg, J. M. Authoritative Sources in a Hyperlinked Environment. *ACM 46*, 5 (1999), 604–632.

[52] Kurose, J. F., and Ross, K. W. *Computer Networking: A Top-Down Approach Featuring the Internet*, 2nd ed. Addison Wesley, 2003.

[53] Lam, D., Rohall, S. L., Schmandt, C., and Stern, M. K. Exploiting e-mail structure to improve summarization. In *Proceedings of CSCW (Interactive Posters)* (New Orleans, LA, USA, November 2002).

[54] Lang, K. Newsweeder: Learning to Filter Netnews. In *Proceedings of ICML* (Tahoe City, CA, USA, July 1995), pp. 331–339.

[55] Lappin, S., and Leass, H. J. An Algorithm for Pronominal Anaphora Resolution. *Computational Linguistics 20*, 4 (1994), 535–561.

[56] Leiner, B. M., Cerf, V. G., Clark, D. D., Kahn, R. E., Kleinrock, L., Lynch, D. C., Postel, J., Roberts, L. G., and Wolff, S. A Brief History of the Internet. `http://www.isoc.org/internet/history/brief.shtml`, 2003.

[57] Lin, C.-Y. Looking for a Few Good Metrics: ROUGE and its Evaluation. In *Proceedings of NTCIR Workshop* (Tokyo, JP, June 2004).

[58] Mani, I. Summarization evaluation: An overview. In *Proceedings of NAACL* (Pittsburgh, PA, USA, June 2001).

[59] Mani, I., and Bloedorn, E. Multi-document summarization by graph search and matching. In *Proceedings of AAAI* (1997), pp. 622–628.

[60] Mann, W. C., and Thompson, S. A. Rhetorical Structure Theory: A Theory of Text Organization. Tech. Rep. ISI/RS-87-190, Information Sciences Institute, Los Angeles, CA, USA, 1987.

[61] Manning, C., and Schütze, H. *Foundations of Statistical Natural Language Processing.* MIT Press, Cambridge, MA, USA, 1999.

[62] Marcu, D. From Discourse Structures to Text Summaries. In *Proceedings of ACL* (Madrid, ES, July 1997), pp. 82–88.

[63] McKeown, K., Shrestha, L., and Rambow, O. Using Question-Answer Pairs in Extractive Summarization of Email Conversations. *Lecture Notes in Computer Science 4394* (2007), 542–550.

[64] de Meulder, F., Daelemans, W., and Hoste, V. A Named Entity Recognition System for Dutch. In *Proceedings of CLIN'01* (Enschede, NL, November 2002), vol. 12, pp. 77–88.

[65] Mishne, G., and Glance, N. Predicting Movie Sales from Blogger Sentiment. In *Proceedings of AAAI* (Boston, MA, USA, July 2006).

[66] Mitkov, R. Multilingual Anaphora Resolution. *Machine Translation* (1999).

[67] Mulder, M. P. Schijtlijster: A Lexical Grammatical Implementation of Affect. Master's thesis, University of Twente, Enschede, NL, 2003.

[68] Nenkova, A., and Vanderwende, L. The Impact of Frequency on Summarization. Tech. Rep. MSR-TR-2005-101, Microsoft Research, Redmond, WA, USA, 2005.

[69] Nigam, K., and Hurst, M. Towards a Robust Metric of Opinion. In *Proceedings of AAAI* (San Hose, CA, USA, July 2004).

[70] Osman, D. J., and Yearwood, J. L. Opinion Search in Web Logs. In *Proceedings of ADC* (Ballarat, VIC, AUS, January 2007).

[71] Ounis, I., de Rijke, M., Macdonald, C., Mishne, G., and Soboroff, I. Overview of TREC-2006 Blog Track. In *Proceedings of TREC* (Gaitherburg, MD, USA, November 2006).

[72] Radev, D. R., and McKeown, K. Generating Natural Language Summaries from Multiple On-Line Sources. *Computational Linguistics 24*, 3 (1998), 469–500.

[73] Radev, D. R., Otterbacher, J., Winkel, A., and Blair-Goldensohn, S. NewsInEssence: Summarizing Online News Topics. *Communications of the ACM 48*, 10 (2005), 95–98.

[74] Rambow, O., Shrestha, L., Chen, J., and Lauridsen, C. Summarizing Email Threads. In *Proceedings of HTL/NAACL Short Papers* (Boston, MA, USA, May 2004), pp. 105–108.

[75] Rienks, R. *Meetings in Smart Environments: Implications of Progressing Technology.* PhD thesis, University of Twente, 2007.

[76] Riloff, E., Patwardhan, S., and Wiebe, J. Feature Subsumption for Opinion Analysis. In *Proceedings of EMNLP* (Sydney, AUS, July 2006), pp. 440–448.

[77] Sallis, P., and Kassabova, D. Computer-Mediated Communication: Experiments with E-mail Readability. *Information Sciences 123* (2000), 43–53.

[78] Salvetti, F., Lewis, S., and Reichenbach, C. Automatic Polarity Classification of Movie Reviews. *Colorado Research in Linguistics 17*, 1 (2004).

[79] Sang, E. T. K. Language-independent named entity recognition. `http://www.cnts.ua.ac.be/conll2002/ner/`, 2005.

[80] Schuth, A., Marx, M., and de Rijke, M. Extracting the Discussion structure in Comments on News-Articles. In *Proceedings of CIKM/WIDM* (Lisboa, PT, November 2007), vol. 123, pp. 97–104.

[81] Soricut, R., and Marcu, D. Sentence Level Discourse Parsing using Syntactic and Lexical Information. In *Proceedings of HTL/NAACL* (Edmonton, CA, May 2003).

[82] Spärck Jones, K. Automatic summarising: The state of the art. *Information Processing and Management 43*, 6 (2007), 1449–1481.

[83] Sporleder, C., van Erp, M., Porcelijn, T., van den Bosch, A., and Arntzen, P. Identifying named entities in text databases from the natural history domain. In *Proceedings of LREC* (Trento, IT, May 2006).

[84] Stegeman, L. Hammer Tagger. `http://www.vf.utwente.nl/~infrieks/stt/stt.html`, 2007.

[85] Stehouwer, J. H. Comparing a TBL Tagger with an HMM Tagger: Time Efficiency, Accuracy, Unknown Words. Capita Selecta Paper (HMI Group, University of Twente), 2006.

[86] Taboada, M., and Mann, W. C. Rhetorical Structure Theory: looking back and moving ahead. *Discourse Studies 8* (2006), 423–459.

[87] Tigelaar, A. S. A QA System for the MRSA Web Portal: Evaluating Retrieval Techniques for a Domain-Specific Corpus. Capita Selecta Paper (HMI Group, University of Twente), 2007.

[88] Timmerman, S. Automatic Recognition of Structural Relations in Dutch Text. Master's thesis, University of Twente, 2007.

[89] Toutanova, K., Brockett, C., Gamon, M., Jagarlamudi, J., Suzuki, H., and Vanderwende, L. The PYTHY Summarization System: Microsoft Research at DUC2007. In *Proceedings of HTL/NAACL DUC Workshop* (Rochester, NY, USA, April 2007).

[90] Turney, P. D. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In *Proceedings of ACL* (Philadelphia, PA, USA, July 2002), pp. 417–424.

[91] Vossen, P. Introduction to EuroWordNet. *Computers and the Humanities 32*, 2–3 (1998), 73–89.

[92] Wan, S., and McKeown, K. Generating Overview Summaries of Ongoing Email Thread Discussions. In *Proceedings of COLING* (Geneva, CHE, August 2004).

[93] Weimer, M., Gurevych, I., and Mühlhäuser, M. Automatically Assessing the Post Quality in Online Discussions on Software. In *Proceedings of ACL'07 Demo and Poster Sessions* (June 2007), pp. 125–128.

[94] Wiebe, J. Learning Subjective Adjectives from Corpora. In *Proceedings of AAAI* (Austin, TX, USA, July 2000), pp. 735–740.

[95] Wijnen, H.-J. Sentiment Polarity Tagging Based on Nearby Words. In *Proceedings of TSConIT* (Enschede, NL, 2005).

[96] Wilson, T., Wiebe, J., and Hwa, R. Just how mad are you? Finding Strong and Weak Opinion Clauses. In *Proceedings of AAAI* (San Hose, CA, USA, July 2004).

[97] Witbrock, M. J., and Mittal, V. O. Ultra-Summarization: A Statistical Approach to Generating Highly Condensed Non-Extractive Summaries. In *Proceedings of SIGIR* (Berkeley, CA, USA, August 1999).

[98] Zechner, K. Automatic Summarization of Open-Domain Multiparty Dialogues in Diverse Genres. *Computational Linguistics 28*, 4 (2002), 447–485.

[99] Zhou, L., and Hovy, E. Digesting Virtual 'Geek' Culture: The Summarization of Technical Internet Relay Chats. In *Proceedings of ACL* (Ann Arbor, MI, USA, June 2005), pp. 298–305.

[100] Zhou, L., and Hovy, E. On the summarization of dynamically introduced information: Online discussions and blogs. In *Proceedings of AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs* (Boston, MA, USA, March 2006).

Terms and Definitions

Administrator	Handles technical issues surrounding a Site (or a specific Forum).
Board	See Forum.
First Post	See Initial Post.
Forum	Location on a Site where users can start a discussion Thread concerning specific topics. Usually devoted to some specific topic (e.g. motherboards).
Group	See Forum.
Initial Post	First message that serves as the start of a new Thread. This message is usually accompanied with a title that is also used as the Thread title.
Member	A special Forum user. Less powerful than an Administrator, but more powerful than a normul user. Members can have elevated priviliges such as access to private fora on a site.
Message	Coherent text posted by an author in an existing Thread or as start of a new Thread.
MFS	Message Formatting Score. Assigned to a message representing how well formatted the message is (0.0 - 1.0).
Moderator	Takes care of approving new messages and removing irrelevant messages posted in a Forum.
Multi-quoting	The practice of quoting (parts of) multiple previous messages in one post.
PMR	Positional Message Relevance. A formula capturing the importance of messages in a thread by looking at their position in the thread reply structure tree.

Posts
See Messages.

Precision
Percentage of retrieved documents relevant to a search.

Problem-Solution
A thread wherein a main question is posed, replies are posted and (optionally) follow-up questions are asked. This process (usually) leads to one or more concrete solutions.

PT factor
Participation-Talkativity factor. Expresses both the participation (number of posts) and the talkativity (number of unique words) of an author in a thread as a weighted combination.

Quoteblock
A block of text within a post containing (parts of) text of a previous post (also called a citation). This is used as an anchor for the reply context.

Recall
Percentage representing the number of known relevant documents retrieved, relative to the total number of documents retrieved.

Site
Place where a discussion boards is hosted (e.g. `http://gathering.tweakers.net`).

Statement-Discussion
A thread wherein an opinion statement is voiced, replies are posted and stances are revised. There is usually no real tangible outcome, other than changed points of view or a consensus.

Thread
Technically: (A public exchange of) one or more related messages within a forum concerning a specific topic.
In this research: Technical definition augmented with the requirement that the thread consists of two or more message by different authors.

Topic
See Thread.

Troll
A person that posts irrelevant (and off-topic) messages.

User
A Forum user with no registered account. Has at least read access to (parts of) a Site and can (sometimes) also post in (a limited number of) fora. Since no registration is required, users like this can remain essentially anonymous.

Appendix A

Unified Tagset: Definition and Accuracy

The table on the next page shows how the Spoken Dutch Corpus (CGN) and Brown Tagsets (last two columns) map onto the Unified Tagset (3rd column).

For CGN-72 numeric sequences are shortened by using ranges. For example the mapping to VNW7, VWN8 and VWN9 is shown as VWN7-9. For the Brown corpus, tags with a common prefix are shortened by using parentheses. So JJT and JJS are shortened to JJ(T|S). If the initial part of a tag can also appear without one of the postfixes this is expressed as an empty option in the postfix list, for example BED(|Z) matches both BED and BEDZ.

The Brown corpus also allows for expressing additional characteristics using plus or minus symbols after a tag, as well as allowing $ as a special symbol. With the exception of possessive pronouns and infinitive verbs these symbols are stripped from the tag. They are unused since they provide even finer grained detail which is not needed for the Unified tagset.

Concept	*Characteristic*	*Unified*	*CGN-72*	*Brown*
Adjective	Base	A	ADJ1, 4, 7, 9, 12	JJ
	Comperative	A-C	ADJ2, 5, 8, 10	JJR
	Superlative	A-S	ADJ3, 6, 11	JJ(T|S)
Adverb		V-B	BW	R(B|P|N), RB(R|T), W(RB|QL), QL(|P)
Conjunction		J	VG1-2	C(C|S)
Article		D	LID	DT, AB(L|N), A(T|P)
Noun	Common Singular	N-C-1	N1-2, N4	N(N|R)
	Common Plural[1]	N-C-*	N3	N(NS|RS)
	Proper Singular	N-P-1	N5-6, N8	NP
	Proper Plural	N-P-*	N7	NPS
Pronoun	Personal	P-P	VNW1-6	PPS(|S), PPO
	Reflexive	P-R	VNW7-9	PPL(|S)
	Possessive	P-O	VNW11	PP$(\|$), WP$
	Question	P-Q	VNW14-15, 17	WP(S|O), WDT
	Determiner	P-D	VNW19-21	DT(I|S|X), ABX
	Other	P	VNW12-13, 16, 18, 22-27	PN, EX
Verb	Base	V	WW1-3, 13	VB, DO, HV, BE, MD
	Infinitive	V-I	WW4-6	VB(|G|N)+TO, HV+TO, MD+TO
	Completed	V-C	WW7-9	VB(D|N), DOD, HV(D|N), BED(|Z), BEN
	Progressing	V-P	WW10-12	VB(G|Z), DOZ, HV(G|Z), BE(G|Z|M|R)
Preposition		R	VZ1-3	IN, TO
Punctuation		L	LET	. : , - ( )
Numeral	Cardinal	C-R	TW2	CD
	Ordinal	C-T	TW1	OD
Interjection		I	TSW	UH
Other		O	SPEC	FW, NIL

Table A.1: *Tagger accuracy per tag and overall (ordered high to low).*

Tag	*Accuracy*	*Tag*	*Accuracy*
L	100.00%	V-C	96.67%
I	99.76%	N-C-*	96.61%
P-R	99.74%	P-O	96.39%
R	99.32%	P-D	95.73%
C-R	99.05%	J	95.46%
C-T	99.01%	P-Q	95.24%
P-P	98.93%	A-C	94.95%
D	98.78%	O	94.23%
V-B	98.26%	V-I	93.41%
N-C-1	97.94%	N-P-1	90.51%
A-S	96.81%	P	87.90%
A	96.72%	N-P-*	87.27%
V	96.69%	V-P	84.39%
		Total	97.49%

Table A.1 below shows the tagging accuracy of the Hammer trigram tragger on the Unified tagset for each individual tag as well as the overall result. Training was performed on one million lines from the Spoken Dutch Corpus (CGN), testing was done a separate set of 100.000 lines from that corpus (these training and test sets were provided with the hammer tagger, they were simply converted to the Unified tagset). The results are not as thorough as those provided by e.g. a ten fold cross validation, but are only intended to give a rough indication of the expected tagging accuracy.

We have ommitted a confusion matrix here for reasons of brevity. However, for the bottom four tags (color shaded in table A.1) we provide some insight into the source of their lower accuracy:

- Words that should be tagged V-P (progressing verb) get mistagged as A (adverb) in about 14% of the cases.
- Words that should be tagged N-P-* (plural proper noun) mostly get mistagged as N-C-* (plural common noun) [4%] or N-P-1 (singular proper noun) [4%].
- P (other pronoun) words get mistagged as P-D (determiner pronoun) in about 6% of all cases.
- N-P-1 (singular proper noun) words get mistagged as O (other) in about 6% of all cases.

Except the first, most of these errors do not carry serious implications for the rest of the system. It would be interesting to find out the reasons behind the mistaggings. However, the tagger accuracy is quite high and we use the tagger as a tool in our assignment, and we decided not to pursue this further.

Appendix B

Partial Parsing: Rules and Accuracy

Table B.1 gives an overview of the rules used to find noun phrases. Whilst some of these might apply to English as well, they are based on a Dutch grammar provided by Rieks op den Akker. Hence all the examples in the second column are in Dutch.

The format used is that of regular expressions working at two-levels. The first one is at the tag-level. This makes an expression such as '<A.*>' match the tags A, A-S and A-C. The second level is for tag sequences: '<C-T>?' matches one or zero occurences of the C-T tag.

Of course we would also like to have an indication of the accuracy of this simple set of rules. For this, we turn to the Alpino Treebank [11]. We only use the noun-phrase information available in the treebank for evaluation. Unfortunately while starting evaluations we noticed some differences between our definition of a noun phrase (as expressed in the rules above) and that of Alpino.

Firstly, we do not consider stand-alone pronouns (P-R, P-P, etc.) to be noun phrases. We really only consider something a noun phrase if it is or contains a noun. For our application this is a very useful definition. Hence, during evaluation we did not consider it an error (specifically a false negative) if a stand-alone pronoun is not recognised as a noun phrase.

A second issue is that Alpino finds noun phrases at higher nesting levels. Consider the phrase 'Jan bought a book from Chomsky'. This contains the leaf noun phrases 'Jan', 'a book' and 'Chomsky'. Alpino however also detects the higher level noun phrase 'a book from Chomsky' and in such cases does *not* properly mark the first sub noun phrase. Therefore, Alpino would detect 'Jan', 'a book from Chomsky' and 'Chomsky'. So, it is not enough for us to only consider the leaf noun phrases present in the treebank. As a solution we also did not consider it an error (specifically a false position) during evaluation if any of the first three words at the beginning of what Alpino recognises as a higher-level noun-phrase were tagged as a noun phrase. In the example this would mean it would be counted as a true positive if we would mark 'a book' as a noun phrase (even though Alpino does not agree with us at the leaf level on that, only implicitly at the level above it).

Table B.1: *Partial parsing noun-phrase rules.*

Expression	*Examples (Dutch)*
<D><A.*><N.*>+	het mooie boek, de grote McDonalds
<D><N.*>+	het boek, een boek, de boeken, de schrijver Hermans
<P-D><(C|N).*>+	die Jan, deze auto, die 65nm CPU
<A.*><N.*>	kleine mensen, dure kaartjes
<P-O><N.*>+	mijn boek, onze Laura
<P-O><A.*><N.*>+	mijn mooie boek, onze kleine Laura
<P-P><N.*>+	je auto
<P><N.*>+	wat stipjes
<D><C-R><N.*>+	het eerste kind, de tweede McLaren
<N.*><C-T><N.*><C-T>?	woensdag 31 oktober (2007)
<C-T><N.*>+	drie kaartjes, vier Ferrari's
<C-T><A.*><N.*>+	drie mooie boeken
<C-T>	drie, 8
<N-C.*><N-P.*>	opera forum, bakker Jan, tante Marie
<N.*><C-T>	zaal 1
<N.*>+	ABN AMRO, Herbert Jan Dijksma, auto shell

Table B.2: *Partial parser performance (on Alpino treebank parts).*

Corpus Part	*Description*	*# Lines*	*Precision*	*Recall*
cgn_ex	CGN Annotation Guidelines	271	88.10%	96.17%
g_suite	Alpino Development Corpus	885	89.24%	95.08%
cdbl	Eindhoven Corpus (Newspaper part)	7153	85.03%	95.60%

Evaluation results can be seen in table B.2. We have tested on three (larger) parts of the Alpino treebank. As can be seen, precision and recall scores of these simple rules are remarkably good. Averaged and rounded, the F-score would be about 91% (with $\alpha = 0.5$). Certainly good enough for our purposes. Because we did not deem it necessary, we did not investigate the precise source of the remaining errors. However, we think some of this may be due to tagging errors, as the maximum performance of the partial parser is always impaired to some extent by the tagger.

Appendix C

Semantic Tags

The following is a list of semantic tags and named entity that are assigned to words.

Tags for General Semantic Units are assigned based on regular expressions. Note that for quantities many prefixes are allowed (kilo-, mega-, etc.) as defined in the metric system. Currently, imperial quantities are not supported.

Named Entities are based on thread level recognition (such as that of author names) and lists (gazetteers).

Table C.1: *Semantic and named entity tags.*

Description	*Example(s)*
DateTime	3 May 2006, 11:23
Address-Email	user@evolution.tld
Address-URI	http://www.gnome.org
Address-File	C:\autoexec.bat
Quantity-Percentage	100%
Quantity-Length	1 m, two kilometer
Quantity-Mass	1 kg, two Newton
Quantity-Time	1 min, two hours
Quantity-Money	$ 1, two euro
Quantity-Information	1 KB, two megabytes
Quantity-Temperature	1 K, two degrees celsius
Quantity-Electric	1 V, 2 Weber
Quantity-Luminosity	1 ca, two lumen
Quantity-Radioactivity	1 Bq, two sievert
Emoticon-State-Happy	:)
Emoticon-State-Sad	:(
Emoticon-State-Confused	:s
Emoticon-State-Annoyed	:/
Emoticon-State-Undecided	:\
Emoticon-State-Sarcastic	;)
Emoticon-State-Surprise	:O
Emoticon-Action-Laugh	XD
Emoticon-Action-Cry	:,(
Person-Author	John, Jack53 (author in thread!)
Person-Other	Ellen van den Berg, Minister
Organisation	University of Twente, 3M
Location	Netherlands, Alaska, Eindhoven

Appendix D

Evaluation Setup

This appendix contains detailed information with respect to the set-up of the evaluation. Participants were asked a variety of questions. The actual questions are shown in several tables. Each table lists the identifier of each question, the question itself in Dutch and English and the possible choices (in English) as well as a rationale for the question.

The evaluation consisted of several steps described in section 8.1. What follows is a short overview which step is described in which table (or section):

1. Introductory screen
   Table D.1 on the following page.
2. First discussion (part 1)
   Section D.1 on page 120 (details and screenshots in section 8.1 on page 67).
3. First discussion (part 2)
   Table D.2 on page 117 (note that this table also contains the question for step 5).
4. Second discussion (part 1)
   Section D.2 on page 124.
5. Second discussion (part 2)
   Table D.2 on page 117 (only question six (T2-6) was different because of the different type of discussion).
6. Your opinion
   Table D.3 on page 118.
7. General information
   Table D.4 on page 119.

Table D.1: *Participant experience and competency questions.*

ID	*Field*	*Description*
E1	Dutch:	Hoe vaak bezoekt u discussies op een forum (lezen van berichten)?
	English:	How often do you visit discussions on a forum (reading messages)?
	Choices:	Seldom, Couple of times a year, Monthly, Weekly, Daily.
	Rationale:	Finding the participant's experience level with regard to fora.
E2	Dutch:	Hoe vaak draagt u bij aan discussies op een forum (plaatsen van berichten)?
	English:	How often do you contribute to discussions on a forum (posting messages)?
	Choices:	Seldom, Couple of times a year, Monthly, Weekly, Daily.
	Rationale:	Finding the participant's experience level with regard to fora.
E3	Dutch:	Van hoeveel verschillende fora maakt u regelmatig gebruik (lezen en plaatsen van berichten)?
	English:	How many different fora do you regularly use (reading and posting messages)?
	Choices:	0, 1-2, 3-4, 5-6, 7+.
	Rationale:	Finding the participant's experience level with regard to fora.
E4	Dutch:	Hoe behendig denkt u dat u zelf bent in het vinden van informatie op het Internet?
	English:	Hoe agile do you think you are in finding information on the Internet?
	Choices:	5-point scale: Not very agile - Very agile.
	Rationale:	Finding the participant's experience level in general searching.
E5	Dutch:	Hoe behendig denkt u dat u zelf bent in het vinden van informatie op een discussie forum?
	English:	How agile do you think you are in finding information on a discussion forum?
	Choices:	5-point scale: Not very agile - Very agile.
	Rationale:	Finding the participant's experience level in searching on fora.
E6	Dutch:	Is Nederlands uw moedertaal?
	English:	Is Dutch your first language?
	Choices:	Binary: Yes/No.
	Rationale:	Finding the participant's language competency.
E7	Dutch:	Hoe vaardig vindt u uzelf in de Nederlandse taal?
	English:	At what level of competentency would you rate yourself with respect to the Dutch language?
	Choices:	5-point scale: Not very competent - Very competent.
	Rationale:	Finding the participant's language competency.

Table D.2: *Automatically generated summary judgement.*

ID	*Field*	*Description*
T1-3	Dutch:	Wat voor cijfer zou u deze samenvatting geven?
T2-3	English:	How would you grade this summary?
	Choices:	10-point grade scale: 1 - 10.
	Rationale:	Finding out the perceived summary quality.
T1-4	Dutch:	Wat vind u van de dekking van deze automatisch gegenereerde samenvatting?
T2-4	English:	What do you think of the coverage of this automatically generated summary?
	Choices:	5-point scale: Bad - Good.
	Rationale:	Finding out the perceived summary coverage.
T1-5	Dutch:	Wat vind u van de samenhang van deze automatisch gegenereerde samenvatting?
T2-5	English:	What do you think of the coherence of this automatically generated summary?
	Choices:	5-point scale: Bad - Good.
	Rationale:	Finding out the perceived summary coherence.
T1-6	Dutch:	Hoe vindbaar is een mogelijk antwoord op de hoofdvraag in de discussie in deze samenvatting?
	English:	How easy is it to find the possible answer to the main question in the discussion in this summary?
	Choices:	5-point scale: Difficult to find - Easy to find.
	Rationale:	Finding out whether the question and possible answers are well represented in the summary.
T2-6	Dutch:	Hoe vindbaar is de rode draad van de discussie in deze samenvatting?
	English:	How easy is it to find the main focus of the discussion in this summary?
	Choices:	5-point scale: Difficult to find - Easy to find.
	Rationale:	Finding out whether the summary captures the focus of the discussion.
T1-7	Dutch:	Wat vindt u van de tussen blokhaken weergegeven referenties?
T2-7	English:	What do you think of the references (indicated between square brackets)?
	Choices:	5-point scale: Superfluous - Helpful.
	Rationale:	Finding whether the anaphora resolution adds anything to the summary.

Table D.3: *Participant's opinion.*

ID	*Field*	*Description*
M1	Dutch:	Denkt u dat een automatische samenvatter een zinvolle toepassing is?
	English:	Do you think an automatic summarizer is a useful application?
	Choices:	5-point scale: Not useful - Very useful.
	Rationale:	Finding out the participant's opinion.
M2	Dutch:	Hoe kan een automatische samenvatter volgens u het beste geintegreerd worden in de gebruikersinterface?
	English:	How should an automatic summarizer be integrated in the user interface according to you?
	Choices:	Within the forum website, As separate website, As part of my webbrowser (plug-in), As separately installable application.
	Rationale:	Finding out how a summarizer had best be integrated in the user experience.
M3	Dutch:	Hoe vaak zou u een automatische samenvatter gebruiken als hulpmiddel bij discussies?
	English:	How often would you use an automatic summarizer as an aid?
	Choices:	Never, Every now and then, Regularly, All the time.
	Rationale:	Finding out how often a potential user would use such an application.
M4	Dutch:	Denkt u dat het zinvol is om subjectieve informatie uit een samenvatting te kunnen laten?
	English:	Do you think it would be useful to be able to leave subjective information out of a summary?
	Choices:	5-point scale: Not useful - Very useful.
	Rationale:	Finding out whether the bias option towards objectivity is a useful feature.
M5	Dutch:	Denkt u dat het zinvol is om objectieve informatie uit een samenvatting te kunnen laten?
	English:	Do you think it would be useful to be able to leave objective information out of a summary?
	Choices:	5-point scale: Not useful - Very useful.
	Rationale:	Finding out whether the bias option towards subjectivity is a useful feature.

Table D.4: *Participant general information.*

ID	*Field*	*Description*
A1	Dutch:	Wat is uw geslacht?
	English:	What is your gender?
	Choices:	Male, Female, Do not want to disclose.
	Rationale:	Demographic.
A2	Dutch:	Wat is uw leeftijd?
	English:	What is your age?
	Choices:	<21, 21–30, 31–40, 41–50, 51–60, 61–70, > 71, Do not want to disclose.
	Rationale:	Demographic.
A3	Dutch:	Heeft u nog opmerkingen of suggesties?
	English:	Do you have any remarks or suggestions?
	Choices:	This is a free-form text field.
	Rationale:	-

D.1 First discussion

First discussion (in Dutch). Topic: 'Werken in de ICT met MBO4 (ict-beheerder)'. Messages in chronological order. Note that discussions were presented to the user in their authentic forum format. This table is an abstract representation. The automatically generated summary can be found below the messages in a separate table.

Source: `http://gathering.tweakers.net/forum/list_messages/1289050`

This is a discussion about someone (wd200) who has completed his senior secondary vocational education and has started higher education. Since he is not really doing well he is interested in how much work there is at the level of his prior completed education. This is a thread of *Problem-Solution* type.

User	*Message*
wd200	Ik heb zelf vorig jaar MBO4 ict beheerder afgerond. Ik ben wel door aan het studeren om het hbo(HTS-A) maar dit gaat niet zo lekker. Mijn vraag is hoeveel werk is er te vinden voor een ict beheerder op MBO4 niveau ? zijn er mensen die na het mbo zijn gaan werken ? - wat voor werkgevers zijn mogelijk (peak, call2 ?) - wat is een beetje een gangbaar start salaris ? - wat zijn werkend de doorstudeer mogelijkheden ? wie is hier gaan werken na het mbo en hoe bevalt het ? wat voor werk doe je ? Hoe ziet een dag werken er uit ? (vergoedingen ?)
N3oC	Voor vergoedingen, salarissen en secundaire arbeidsvoorwaarden kijk eens in "Wat verdient een ICTer gemiddeld? (deel 7)". Ik denk dat er genoeg werk in te vinden is, zet je CV eens op monsterboard oid, zat mensen die je uit willen nodigen (zelf van ICT Beheerder niv. 4 naar HBO informatica en mensen wilden MBOers graag hebben)
ICT Assist	*knip werving* Welkom op GoT. Zoals je in Werk & Inkomen - Policy kunt lezen is elke vorm van werving uitgesloten op het forum.

User	*Message*
m0nk	Momenteel is de markt goed voor de werkzoekende. Ik heb 2 weken geleden mjin cv op nvbank gezet en ik werd helemaal gek gebeld. Als je bij een detacheerder in dienst gaat zal je hoogst waarschijnlijk bij een opdrachtgever neergezet worden (afhankelijk van je opleiding en ervaring). Soms zijn het opdrachten van 6 maanden, soms korter en uiteraard soms langer. Vaak kan je bij je werkgever verschillende cursussen volgen en examens doen (let wel op je studieschuld). Hoe je werkdag eruit ziet hangt er vanaf bij welke klant je zit :) opstaan-werken-thuiskomen, net zoals elke andere baan... :P Vergoedingen/Salaris zijn bij elke werkgever anders. Er zijn trouwens momenteel zat vacatures voor junior systeembeheerders/helpdesk/werkplekbeheer. Kijk maar eens op de verschillende werkaanbod sites.
Red Boll	*[Citaat (oorspronkelijk geplaatst door wd200):* *Ik heb zelf vorig jaar MBO4 ict beheerder afgerond.]* Vraagje: Welke certificeringen heb je nu eigenlijk behaald in die 4(?) jaar? Deze opleiding is "nog van voor mijn tijd". ;)
Red Boll	*[Citaat: Er zijn trouwens momenteel zat vacatures voor junior systeembeheerders/helpdesk/werkplekbeheer. Kijk maar eens op de verschillende werkaanbod sites.]* De junior systeembeheerders zitten steeds meer in India/Polen... Dat wordt lastig instromen voor de nieuwe collega's ben ik bang :| Via detachering moet je echter zo aan de slag kunnen lijkt me...
Aikon	Gooi je cv eens online en je merkt 't vanzelf. Ik heb ook mbo ict4, en werd helemaal gek gebeld. Overal waar ik ging sollicteren ben ik aangekomen en kon dus uit 4 banen kiezen. Salaris moet je rond de 1600~1800 denken. Doorstudeermogelijkheden hangen natuurlijk van het bedrijf af, kan je bespreken. Mijn certificaten worden bijv. betaald.
wd200	*[Citaat (oorspronkelijk geplaatst door Red Boll):* *[...] Vraagje: Welke certificeringen heb je nu eigenlijk behaald in die 4(?) jaar?* *Deze opleiding is "nog van voor mijn tijd". ;)]* Geen certi. Ja itil en ecdl op school afgesloten als vak zijnde

User	*Message*
Red Boll	Weet je al of je gaat door leren of gaat werken? Lastige keuze... Een HBO diploma in je achterzak geeft je wel meer mogelijkheden, zeker later in je carriere... Een eerste opstap baan in de IT vinden lijkt me niet zo'n uitdaging voor je. Heb je al een bepaalde job in gedachte die je leuk lijkt?
MuddyMagical	Ik ben zelf met MBO4 bij TTP begonnen. Ik kan gewoon mijn MCSE, Cisco, etc. halen. Als je wilt kan ik je wel in contact brengen...
CyberTijn	*[Citaat (oorspronkelijk geplaatst door wd200): Letterlijk het eerste bericht in de discussie]* - wat voor werkgevers zijn mogelijk (peak, call2 ?) Bakken met detacheerder die zitten te springen om nieuwe medewerkers. Maak je geen zorgen, een baan vinden in de ICT met alleen MBO is een eitje - wat is een beetje een gangbaar start salaris ? tussen de 1600 en 1700 is niet onrealistisch - wat zijn werkend de doorstudeer mogelijkheden ? Ieder fatsoenlijk bedrijf geeft z'n mensen de mogelijkheden om trainingen te volgen en certificaten te halen om zijn / haar werk beter te kunnen doen en om door te stromen naar een hogere functie. Als je in de uitvoerende tak van ICT gaat zitten (dus zonder het woord "manager" in je functietitel) zul je altijd door moeten blijven leren. Ontkom je niet aan. - wie is hier gaan werken na het mbo en hoe bevalt het ? 3 jaar geleden ben ik met diploma van het MBO het bedrijfsleven in gegaan. Begonnen op een servicedesk, anderhalf jaar gedaan, en prima bevallen. - wat voor werk doe je ? Na anderhalf jaar servicedesk doorgestroomd naar 1e-lijns netwerkbeheer - hoe ziet een dag werken er uit ? Opstaan, naar werk rijden, koffie drinken, beetje werken, lunchen, beetje werken, diner, nog beetje werken, naar huis rijden, genieten van je vrije avond zonder huiswerk :)

User	*Message*
hypz	*[Citaat (oorspronkelijk geplaatst door wd200): Letterlijk het eerste bericht in de discussie]* - wat voor werkgevers zijn mogelijk (peak, call2 ?) Inderdaad, via detacheerders ben je zo aan het werk, en het is niet verkeerd om mee te beginnen al zal het salaris daar niet zo hoog zijn. Staar je niet blind op de vacatures om monsterboard waar ze vaak een hele lijst met software kennis en veel ervaring eisen, maar plaats gewoon je cv eens op monsterboard en houd je telefoon ff een dagje aan. Gegarandeerd dat je als mbo ict'er de volgende dag plat gebeld word door bedrijven die je willen hebben. - wat is een beetje een gangbaar start salaris ? Ik ben zelf vorig jaar begonnen voor 1400 bruto als werkplekbeheerder, is inmiddels 1500 geworden, en vanaf volgende maand 1600 als het goed is. - wat zijn werkend de doorstudeer mogelijkheden ? Zoveel als je zelf wil, ieder fatsoenlijk bedrijf bied je voldoende mogelijkheden om bij te blijven leren, meestal in de vorm van certificaten etc. wie is hier gaan werken na het mbo en hoe bevalt het ? Ik heb een jaartje HBO gedaan, maar ben gestopt omdat ik ging verhuizen en het niet zo zag zitten weer een andere school op te zoeken etc, en ben aan het werk gegaan en heb er geen spijt van. wat voor werk doe je ? Werkplekbeheer bij een grote energieleverancier. Hoe ziet een dag werken er uit ? In de auto naar het werk rijden, auto parkeren, laptop opstarten, kopje koffie halen, mail checken, got forum checken, fok lezen. Planning etc opstarten, en checken wat er allemaal voor incidenten zijn die ik die dag moet / ga doen. Incidenten oplossen etc, daarna de tickets weer bijwerken / afsluiten enzo, dan weer verder tot de dag om is. Op zo'n dag kom ik ook geregeld bij de koffie automaat, en maak gezellig een praatje bij de mensen waar ik langs kom. Het is best relaxed werk over het algemeen. Soms is best wel stressvol, maar over het algemeen valt dat best mee moet ik zeggen. Ik zit nog 2 maanden bij het bedrijf waar ik momenteel gedetacheerd ben, probeer in die tijd nog wat certificaten te halen zodat ik op een wat interresantere opdracht kan gaan zitten hierna. (vergoedingen ?) Reiskosten als je met eigen auto rijd.

User	*Message (Summarized & Included)*
wd200	Ik [wd200] heb zelf vorig jaar MBO4 ict beheerder afgerond. Ik [wd200] ben wel door aan het studeren om het hbo(HTS-A) maar dit gaat niet zo lekker. Mijn [wd200's] vraag is hoeveel werk is er te vinden voor een ict beheerder [MBO4 ict beheerder] op MBO4 niveau? wie is hier gaan werken na het mbo en hoe bevalt het?
RedBoll	Vraagje: Welke certificeringen heb je [wd200] nu eigenlijk behaald in die 4(?) Deze opleiding is "nog van voor mijn [Red Boll's] tijd".
wd200	Geen certi. Ja itil en ecdl op school afgesloten als vak zijnde
CyberTijn	Maak je [wd200] geen zorgen, een baan vinden in de ICT met alleen MBO is een eitje Als je [wd200] in de uitvoerende tak van ICT gaat zitten (dus zonder het woord 'manager' in je [wd200's] functietitel) zul je [wd200] altijd door moeten blijven leren.
hypz	- wat voor werkgevers zijn mogelijk (peak, call2?) Inderdaad, via detacheerders ben je zo aan het werk, en het is niet verkeerd om mee te beginnen al zal het salaris [een gangbaar start salaris] daar niet zo hoog zijn.

D.2 Second Discussion

Second discussion (in Dutch). Topic: 'Ziek door eigen schuld? Dan ook zelf voor de kosten opdraaien'. Messages in chronological order. Note that discussions were presented to the user in their authentic forum format. This table is an abstract representation.

Source: `http://www.stand.nl/forum/showthread.php?t=30136`

This is a discussion started by someone (sunny) that supports the stance that people who are injured (or sick) because of their own fault should pay for their expenses themselves. This is a thread of Statement-Discussion type.

User	*Message*
sunny	Tussen sportief bezig zijn en sporten zit een enorm verschil vooral met betrekking tot blessures. Met name in je vrije tijd aan wedstrijdsporten (voetbal bijvoorbeeld) mee doen, vergroot de kans op het krijgen van blessures enorm. Om de werkgever hiervoor volledig te laten opdraaien lijkt mij asociaal. Dat men er over denkt om de werknemers hier geheel of gedeeltelijk zelf de lasten van te laten dragen is zeker niet onterecht.
Paolo	En hoe zit het dan met andere risico-factoren? Roken bijvoorbeeld, of het drinken van alcohol. Moeten die ook meegenomen worden? Leidt het criterium 'eigen schuld' niet tot een ongewenst ingrijpen van de werkgever in het priveleven van de werknemer? Mijns inziens zijn er maar twee mogelijkheden, of de werkgever helemaal verantwoordelijk, of de werknemer. Een gedeelde verwantwoordelijkheid met het criterium 'eigen schuld' zie ik niet zitten.
dutchbird41	[Citaat (oorspronkelijk geplaatst door Paolo): Letterlijk het bericht hierboven] Wanneer er sprake is van een BEDRIJFS ongeval dan is natuurlijk de WERKGEVER verantwoordelijk ... de premie voor deze verzekering komt dus voor rekening van de baas! In alle andere gevallen lijkt mij dat de WERKNEMER de (zelf gekozen) risico's verzekert en daarvoor ook de premie betaalt. Met zelfgekozen bedoel ik sport, roekeloos autorijden etc. Van ingrijpen in prive-leven is geen sprake ... iedere werknemer mag worden verondersteld een gezond stel hersens te hebben en moet dus heel goed in staat zijn onnodige risico's te vermijden.

User	*Message*
Paolo	[Citaat (oorspronkelijk geplaatst door dutchbird41): Letterlijk het bericht hierboven] Maar u beseft toch ook wel dat er eindeloze discussies gaan ontstaan. Is ziekteverzuim te wijten aan overmatig alcoholgebruik of is te hoge werkdruk de oorzaak? Te weinig bewegen is een bewezen risico-factor, dus ook verantwoordelijkheid werknemer? etc. Dit soort discussies leiden tot niets, daarom de verantwoordelijkheid voor alle ziekteverzuim duidelijk bij een van beide partijen leggen.
aadje	[Titel: :(Mogen wij nog bepalen hoe wij leven] Het wordt steeds erger hier in Nederland. nog effe dan word men door anderen geleeft' ik vraag mijn hoeveel provicie krijg Hr Hermes. van verzekeringen lijk wel steeds meer verborgen reclame. nkb
dutchbird41	*[Citaat (oorspronkelijk geplaatst door Paolo): Letterlijk tweede bericht hierboven]* Denk dat niemand er bezwaar tegen zal maken als de "beslissing" over de oorzaak van het verzuim wordt bepaald door de (huis)arts. Voor beide partijen bindend en dat voorkomt de door u gevreesde eindeloze discussies. Geen schuld, dan de aanvulling met 30% tot 100% ... onnodig risico genomen, dus eigen schuld, dan tevreden met toch nog een beloning (voor onzorgvuldig gedrag) van 70%.
handstoffer	*[Citaat (oorspronkelijk geplaatst door Paolo): Letterlijk derde bericht hierboven]* Van dit soort reactie krijg ik altijd zin in een borrel
Mr. Ed	*[Citaat (oorspronkelijk geplaatst door dutchbird41): Letterlijk tweede bericht hierboven]* Ik dacht dat (huis)artsen een zwijgplicht hadden omtrent hun patienten.......

User	*Message (Summarized & Included)*
sunny	Tussen sportief bezig zijn en sporten zit een enorm verschil vooral met betrekking tot blessures. Dat men er over denkt om de werknemers hier geheel of gedeeltelijk zelf de lasten van te laten dragen is zeker niet onterecht.
Paolo	En hoe zit het [de lasten] dan met andere risico-factoren? Een gedeelde verwantwoordelijkheid met het criterium ' eigen schuld ' zie ik [Paolo] niet zitten.
dutchbird41	Met zelfgekozen bedoel ik [dutchbird41] sport, roekeloos autorijden etc. Van ingrijpen in prive-leven is geen sprake ... iedere werknemer mag worden verondersteld een gezond stel hersens te hebben en moet dus heel goed in staat zijn onnodige risico's te vermijden.
Paolo	Is ziekteverzuim te wijten aan overmatig alcoholgebruik of is te hoge werkdruk de oorzaak? Dit soort discussies leiden tot niets, daarom de verantwoordelijkheid voor alle ziekteverzuim duidelijk bij een van beide partijen leggen.
dutchbird41	Voor beide partijen bindend en dat [er bezwaar] voorkomt de door u [Paolo] gevreesde eindeloze discussies [Dit soort discussies]. Geen schuld, dan de aanvulling met 30% tot 100% ... onnodig risico genomen, dus eigen schuld, dan tevreden met toch nog een beloning (voor onzorgvuldig gedrag) van 70%

Appendix E

Evaluation Results

This appendix contains details with regard to the results of the evaluation. Only the English translation of the questions is shown. On some scales the options have been abbreviated. The detailed set-up can be found in appendix D.

Step 1

Q.	Description	Seldom	Several/Year	Monthly	Weekly	Daily
E1	How often do you visit discussions on a forum (to read messages)?	1	2	4	2	7
E2	How often do you contribute to discussions on a forum (posting messages)?	6	4	3	2	1

Q.	Description	0	1-2	3-4	5-6	7+
E3	How many different fora do you regularly use (reading and posting messages)?	2	7	5	1	1

Q.	Description	Not agile	2	3	4	Very agile
E4	Hoe agile do you think you are in finding information on the Internet?	0	0	1	10	7
E5	How agile do you think you are in finding information on a discussion forum?	0	6	4	7	1

Q.	Description	No	Yes
E6	Is Dutch your first language?	2	16

Q.	Description	Not competent	2	3	4	Very competent
E7	At what level of competentency would you rate yourself with respect to the Dutch language?	0	0	1	10	7

Step 2

Tallied results for message ordering (T1-1) can be found in section 8.2.2, table 8.1 on page 75. Results for sentence selection (T1-2 and T2-2) can be found in section 8.2.3.

Step 3

Q.	Description	1	2	3	4	5	6	7	8	9	10
T1-3	How would you grade this summary?	0	0	2	2	1	7	3	3	0	0

Q.	Description	Bad	2	3	4	Good
T1-4	What do you think of the coverage of this automatically generated summary?	1	5	6	5	1
T1-5	What do you think of the coherence of this automatically generated summary?	0	2	4	7	5

Q.	Description	Difficult	2	3	4	Easy
T1-6	How easy is it to find the possible answer to the main question in the discussion in this summary?	1	4	2	8	3

Q.	Description	Superfluous	2	3	4	Helpful
T1-7	What do you think of the references (indicated between square brackets)?	2	3	1	8	4

Step 4

Tallied results for message ordering (T2-1) can be found in section 8.2.2, table 8.3 on page 75. Results for sentence selection (T1-2 and T2-2) can be found in section 8.2.3.

Step 5

Q.	Description	1	2	3	4	5	6	7	8	9	10
T2-3	How would you grade this summary?	0	1	0	0	1	2	10	3	0	1

Q.	Description	Bad	2	3	4	Good
T2-4	What do you think of the coverage of this automatically generated summary?	1	1	4	11	1
T2-5	What do you think of the coherence of this automatically generated summary?	1	2	4	9	2

Q.	Description	Difficult	2	3	4	Easy
T2-6	How easy is it to find the main focus of the discussion in this summary?	1	4	5	7	1

Q.	Description	Superfluous	2	3	4	Helpful
T2-7	What do you think of the references (indicated between square brackets)?	3	2	5	5	3

Step 6

Q.	Description	Not useful	2	3	4	Very useful
M1	Do you think an automatic summarizer is a useful application?	0	0	1	9	8

Q.	Description	Forum	Website	Webbrowser	Application
M2	How should an automatic summarizer be integrated in the user interface according to you?	11	1	4	2

Q.	Description	Never	Now/Then	Regularly	Never
M3	How often would you use an automatic summarizer as an aid during discussions?	0	9	8	1

Q.	Description	Not useful	2	3	4	Very useful
M4	Do you think it would be useful to be able to leave subjective information out of a summary?	1	3	5	6	3
M5	Do you think it would be useful to be able to leave objective information out of a summary?	2	3	3	8	2

Step 7

Q.	Description	Male	Female	Private
A1	What is your gender?	13	5	0

Q.	Description	<21	21-30	31-40	41-50	51-60	61-70	>71	Private
A2	What is your age?	3	13	2	0	0	0	0	0

This thesis was typeset using LyX (a LaTeX pre-processor) which is free software. Other such software that was used during the creation of this thesis and the prototype software is: GNU Emacs, Inkscape, Scribus, Python, Blender3D and the GNOME desktop, among others. Free software is a matter of liberty, not price. It encompasses the users' freedom to run, copy, distribute, study, change and improve the software. Please support the free software movement. You can find out more at `http://www.gnu.org/`.