KTH Numerical Analysis
and Computer Science

Evaluation of Automatic Text Summarization

A practical implementation

MARTIN HASSEL

Licentiate Thesis
Stockholm, Sweden 2004

TRITA NA 2004-08
ISSN 0348-2952
ISRN KTH/NA/R--04/08--SE
ISBN 91-7283-753-5

KTH Numerisk analys och datalogi
SE-100 44 Stockholm
SWEDEN

Akademisk avhandling som med tillstånd av Kungl Tekniska högskolan framlägges till offentlig granskning för avläggande av teknologie licentiatexamen måndagen den 17 maj 2004 kl 13.15 i E2, Osquars Backe 2 / Lindstedtsvägen 3, Kungl Tekniska högskolan, Valhallavägen 79, Stockholm.

Tryck: Universitetsservice US AB

Abstract

Summaries are an important tool for familiarizing oneself with a subject area. Text summaries are essential when forming an opinion on if reading a document in whole is necessary for our further knowledge aquiring or not. In other words, summaries save time in our daily work. To write a summary of a text is a non-trivial process where one, on one hand has to extract the most central information from the original text, and on the other has to consider the reader of the text and her previous knowledge and possible special interests. Today there are numerous documents, papers, reports and articles available in digital form, but most of them lack summaries. The information in them is often too abundant for it to be possible to manually search, sift and choose which knowledge one should aquire. This information must instead be automatically filtred and extracted in order to avoid drowning in it.

Automatic Text Summarization is a technique where a computer summarizes a text. A text is given to the computer and the computer returns a shorter less redundant extract of the original text. So far automatic textsummarization has not yet reached the quality possible with manual summarization, where a human interprets the text and writes a completely new shorter text with new lexical and syntactic choices. However, automatic text summarization is untiring, consistent and always available.

Evaluating summaries and automatic text summarization systems is not a straightforward process. What exactly makes a summary beneficial is an elusive property. Generally speaking there are at least two properties of the summary that must be measured when evaluating summaries and summarization systems - the Compression Ratio, i.e. how much shorter the summary is than the original, and the Retension Ratio, i.e. how much of the central information is retained. This can for example be accomplished by comparison with existing summaries for the given text. One must also evaluate the qualitative properties of the summaries, for example how coherent and readable the text is. This is usually done by using a panel of human judges. Furthermore, one can also perform task-based evaluations where one tries to discern to what degree the resulting summaries are beneficent for the completion of a specific task.

This licentiate thesis thus concerns itself with the many-faceted art of evaluation. It focuses on different aspects of creating an environment for evaluating information extraction systems, with a centre of interest in automatic text summarization. The main body of this work consists of developing human language technology evaluation tools for Swedish, which has been lacking these types of tools. Starting from manual and time consuming evaluation of the Swedish text summarizer SweSum using a Question-Answering schema the thesis moves on to a semi-automatic evaluation where an extract corpus, collected using human informants, can be used repeatedly to evaluate text summarizers at low cost concerning time and effort.

Thus, the licentiate thesis describes the first summarization evaluation resources and tools for Swedish, and aims at bringing if not order, then at least overview into chaos.

Sammanfattning

Sammanfattningar är ett viktigt redskap för att vi snabbt skall kunna orientera oss inom ett område. De är nödvändiga för att vi skall kunna bestämma oss för om ett dokument är väsentligt för vårt vidare kunskapsinhämtande eller inte. Med andra ord sparar sammanfattningar tid för oss i vårt dagliga arbete. Att skriva en sammanfattning av en text är en inte helt trivial process där man dels skall extrahera den mest centrala informationen från originaltexten, dels ta hänsyn till den som läser sammanfattningen och dennes bakgrundskunskap och eventuella intressen. Idag finns det myriader av dokument, tidningar, rapporter, artiklar etc. tillgängliga i digital form, men ofta saknas det sammanfattningar av denna information. Information som är för omfattande för att det skall vara görligt att manuellt söka, sålla och välja ut vad man skall tillgodogöra sig. Denna information måste istället automatiskt filtreras och extraheras för att man inte skall drunkna i den.

Automatisk textsammanfattning är tekniken där en dator sammanfattar en text. En text skickas till datorn och datorn returnerar ett kortare icke-redundant extrakt av originaltexten. Automatisk textsammanfattning har ännu inte nått upp till den kvalitet som manuell textsammanfattning har, där en människa tolkar texten och skriver en ny kortare text med egna ord och nya syntaktiska konstruktioner. Automatisk textsammanfattning är dock outtröttlig, konsekvent och ständigt tillgänglig.

Bedömning av sammanfattningar och utvärdering av automatiska textsammanfattningssystem är en komplex process. Vad som egentligen gör en sammanfattning användbar är en svårfångad egenskap. Enkelt uttryckt är det åtminstone två egenskaper hos sammanfattningen som måste mätas vid utvärdering av sammanfattningssystem – komprimeringsgraden, dvs. hur mycket av originaltexten som har kastats bort, och bibehållandegraden, dvs. hur mycket av den centrala informationen som har bevarats. Detta kan t.ex. åstadkommas genom jämförelser mot manuellt framställda sammanfattningar av den givna texten. Sammanfattningens kvalitativa egenskaper måste också bedömas, såsom hur sammanhängande och förståelig texten är. Detta sker typiskt med hjälp av en panel av domare. Vidare kan även uppgiftsbaserade utvärderingar utföras, där man ser till vilken grad de resulterande sammanfattningarna är användbara till att utföra en viss given uppgift.

Denna licentiatavhandling behandlar den mångfacetterade konst som utvärdering är. Den inriktar sig främst på olika aspekter av skapandet av en miljö för utvärdering av system för informationsextraktion, med huvudsakligt fokus på automatisk textsammanfattning. Huvuddelen av detta arbete har bestått i att bygga svenska språkteknologiska utvärderingsverktyg, något som tidigare varit en bristvara. Med en startpunkt i tidskrävande manuell utvärdering av den svenska automatiska textsammanfattaren SweSum och med vidare utveckling mot halvautomatisk utvärdering där en extraktkorpus, insamlad med hjälp av mänskliga informanter, kan användas för att effektivt och repetitivt utvärdera textsammanfattare.

Licentiatavhandlingen beskriver således de första svenska utvärderingsresurserna för automatisk textsammanfattning, och syftar till att bringa om inte ordning, så i alla fall översikt över kaos.

List of Papers and How to read this thesis

This licentiate thesis is based on the work contained in the following three published papers and one technical report. Each chapter corresponds to one paper, except for the Summary which is an introduction to the problem area and a description of the whole thesis. This type of thesis is called in Swedish *Sammanläggningsavhandling*, which means that it consists of a number of published papers with an introductory chapter.

Summary.
Introduction. The introductory chapter describes the different contributions of the thesis and the relation between them. This chapter is enough to obtain a fair picture of the thesis.

Paper 1. Hassel, M: Internet as Corpus - Automatic Construction of a Swedish News Corpus NODALIDA '01, May 2001.

Paper 2. Carlberger, J., H. Dalianis, M. Hassel, O. Knutsson: Improving Precision in Information Retrieval for Swedish using Stemming NODALIDA '01, May 2001.

Paper 3. Dalianis, H. and M. Hassel: Development of a Swedish Corpus for Evaluating Summarizers and other IR-tools, Technical report, TRITA-NA-P0112, IPLab-188, NADA, KTH, June 2001.

Paper 4. Hassel, M: Exploitation of Named Entities in Automatic Text Summarization for Swedish, NODALIDA '03, May 2003.

Acknowledgements

I would like to thank, most and foremost, my partner in life – Basia – who has stood by me, always ready with help, support and encouragement, and of course my supervisor Dr. Hercules Dalianis whose sound advice and support to me only has been matched by his efforts and determination to put me through this ordeal. Without these two people this licentiate thesis would never be in existence.

I would also like to thank my roomie Ola Knutsson for making many of my days brighter with stimulating discussions, always offering proof-reading and comments as well as insights that make me shift my view on a problem when needed.

Also, gratitude is offered to Prof. Kerstin Severinson-Eklundh, Prof. Koenraad de Smedt and my father – Rolf Hassel – for commenting on drafts of this thesis and also to fellow researchers and students who have contributed with time, effort and expertise to the papers included in this thesis and the experiments they describe.

Contents

Chapter 1

Summaries and the Process of Summarization

1.1 Introduction

Text summarization (or rather, automatic text summarization) is the technique where a computer automatically creates an abstract, or summary, of one or more texts. The initial interest in automatic shortening of texts was spawned during the sixties in American research libraries. A large amount of scientific papers and books were to be digitally stored and made searchable. However, the storage capacity was very limited and full papers and books could not be fit into databases those days. Therefore summaries were stored, indexed and made searchable. Sometimes the papers or books already had summaries attached to them, but in cases were no readymade summary was available one had to be created. Thus, the technique has been developed for many years (see Luhn 1958, Edmundson 1969, Salton 1988) and in recent years, with the increased use of the Internet, there have been an awakening interest for summarization techniques. Today the situation is quite the opposite from the situation in the sixties. Today storage is cheap and seemingly limitless. Digitally stored information is available in abundance and in a myriad of forms to an extent as to making it near impossible to manually search, sift and choose which information one should incorporate. This information must instead be filtered and extracted in order to avoiding drowning in it.

1.1.1 The World According to ISO

According to the documentation standard ISO 215:1986, a summary is a "brief restatement within the document (usually at the end) of its salient findings and conclusions, and is intended to complete the orientation of a reader who has studied the preceding text" while an abstract is, according to the same standard, a "Short representation of the content of a document without interpretation or criticism". In this paper, however, they will be used somewhat interchangeably. In the field of

automatic text summarization it is customary to differentiate between extraction based, or cut-and-paste, summaries where the summary is composed of more or less edited fragments from the source text (this is the task of text extraction), as opposed to abstraction based summaries ("true abstracts") where the source text is transcribed into some formal representation and from this regenerated in a shorter more concise form, see Hovy and Lin (1997). A good overview of the field can be found in Mani and Maybury (1999).

1.1.2 In Defense of the Abstract

Why do we need automatic text summarization, indeed, why do we need summaries or abstracts at all? In the words of the American National Standards Institute (ANSI 1979) – "A well prepared abstract enables readers to identify the basic content of a document quickly and accurately, to determine its relevance to their interests, and thus to decide whether they need to read the document in its entirety". Actually the abstract is highly beneficial in several information acquisition tasks, some examples are given in (Borko and Bernier 1975):

- Abstracts promote current awareness
- Abstracts save reading time
- Abstracts facilitate selection
- Abstracts facilitate literature searches
- Abstracts improve indexing efficiency
- Abstracts aid in the preparation of reviews

Furthermore, human language is highly redundant, probably to facilitate error recovery in highly noisy channels. Mathematician and electrical engineer Claude E. Shannon has, for example, using a training data of 583 million words to create a trigram language model and corpus of 1 million words for testing, shown a 75% redundancy of English on letter level (Shannon 1951). Shannon initially defined redundancy as "the discovery of long-windedness" and accordingly it is not the amount of information that is increased, but the probability that the information reaches the recipient.

Fittingly, entropy experiments have also shown that humans are just as good at guessing the next letter – thus discerning the content of the text on a semantic level – after seeing 32 letters as after 10,000 letters (Burton and Licklider 1955). Other experiments (Morris et al. 1992) concerning reading comprehension of extraction based summaries compared to full documents have shown that extracts containing 20% or 30% of the source document are effective surrogates of the source document. Performance on 20% and 30% extracts is no different than informative abstracts.

Then, how does one go about constructing an abstract? Cremmins (1996) give us the following guidelines from the American National Standard for Writing Abstracts:

- State the purpose, methods, results, and conclusions presented in the original document, either in that order or with an initial emphasis on results and conclusions.
- Make the abstract as informative as the nature of the document will permit, so that readers may decide, quickly and accurately, whether they need to read the entire document.
- Avoid including background information or citing the work of others in the abstract, unless the study is a replication or evaluation of their work.
- Do not include information in the abstract that is not contained in the textual material being abstracted.
- Verify that all quantitative and qualitative information used in the abstract agrees with the information contained in the full text of the document.
- Use standard English and precise technical terms, and follow conventional grammar and punctuation rules.
- Give expanded versions of lesser known abbreviations and acronyms, and verbalize symbols that may be unfamiliar to readers of the abstract.
- Omit needless words, phrases, and sentences.

In automatic abstracting or summarization, however, one often distinguishes between informative and indicative summaries, where informative summaries intend to make reading of source unnecessary, if possible. Indicative summaries, on the other hand, act as an appetizer giving an indication of the content of the source text, thus making it easier for the reader to decide whether to read the whole text or not.

1.2 Automatic Text Summarization

Summarization approaches are often, as mentioned, divided into two groups, text extraction and text abstraction. Text extraction means to identify the most relevant passages in one or more documents, often using standard statistically based information retrieval techniques augmented with more or less shallow natural language processing and heuristics. These passages, often sentences or phrases, are then extracted and pasted together to form a non-redundant summary that is shorter than the original document with as little information loss as possible. Sometimes the extracted fragments are post-edited, for example by deleting subordinate clauses or joining incomplete clauses to form complete clauses (Jing and McKeown 2000, Jing 2000).

Text abstraction, being the more challenging task, is to parse the original text in a deep linguistic way, interpret the text semantically into a formal representation, find new more concise concepts to describe the text and then generate a new shorter text, an abstract, with the same information content. The parsing and interpretation of a text is an old research area that has been investigated for many years. In this area we have a wide spectrum of techniques and methods ranging from word

by word parsing to rhetorical discourse parsing as well as more statistical methods or a mixture of all.

1.2.1 Application Areas

The application areas for automatic text summarization are extensive. As the amount of information on the Internet grows abundantly, it is difficult to select relevant information. Information is published simultaneously on many media channels in different versions, for instance, a paper newspaper, web newspaper, WAP[1] newspaper, SMS[2] message, radio newscast, and a spoken newspaper for the visually impaired. Customisation of information for different channels and formats is an immense editing job that notably involves shortening of original texts.

Automatic text summarization can automate this work completely or at least assist in the process by producing a draft summary. Also, documents can be made accessible in other languages by first summarizing them before translation, which in many cases would be sufficient to establish the relevance of a foreign language document. Automatic text summarization can also be used to summarize a text before an automatic speech synthesizer reads it, thus reducing the time needed to absorb the key facts in a document. In particular, automatic text summarization can be used to prepare information for use in small mobile devices, such as a PDA,[3] which may need considerable reduction of content.

1.2.2 Approaches to Automatic Text Summarization

Automatic Text Summarization is a multi-facetted endeavor that typically branches out in several dimensions. There is no clear-cut path to follow and summarization systems usually tend to fall into several categories at once. According to (Sparck-Jones 1999, Lin and Hovy 2000, Baldwin et al. 2000), among others, we can roughly make the following inconclusive division.

Source Text (Input):

- Source: single-document vs. multi-document
- Language: monolingual vs. multilingual
- Genre: news vs. technical paper
- Specificity: domain-specific vs. general
- Length: short (1-2 page docs) vs. long (> 50 page docs)
- Media: text, graphics, audio, video, multi-media

[1] *Wireless Application Protocol*, a secure specification that allows users to access information instantly via handheld wireless devices such as mobile phones, pagers and communicators.

[2] *Short Message Service*, the transmission of short text messages to and from a mobile phone, fax machine and/or IP address. Messages must be no longer than 160 alpha-numeric characters.

[3] *Personal Digital Assistant* small mobile hand-held device that provides computing and information storage and retrieval capabilities, often contains calendar and address book functionality.

Purpose:

- Use: generic vs. query-oriented
- Purpose: what is the summary used for (e.g. alert, preview, inform, digest, provide biographical information)?
- Audience: untargeted vs. targeted (slanted)

Summary (Output):

- Derivation: extract vs. abstract
- Format: running text, tables, geographical displays, timelines, charts, etc.
- Partiality: neutral vs. evaluative

The generated summaries can also be divided into different genres depending on their intended purpose, for example: headlines, outlines, minutes, biographies, abridgments, sound bites, movie summaries, chronologies, etc. (Mani and Maybury 1999). Consequently, a summarization system falls into at least one, often more than one, slot in each of the main categories above and thus must also be evaluated along several dimensions using different measures.

1.3 Summarization Evaluation

Evaluating summaries and automatic text summarization systems is not a straightforward process. What exactly makes a summary beneficial is an elusive property. Generally speaking there are at least two properties of the summary that must be measured when evaluating summaries and summarization systems: the Compression Ratio (how much shorter the summary is than the original);

$$CR = \frac{length\ of\ Summary}{length\ of\ Full\ Text} \tag{1.1}$$

and the Retension Ratio (how much information is retained);

$$RR = \frac{information\ in\ Summary}{information\ in\ Full\ Text} \tag{1.2}$$

Retention Ratio is also sometimes referred to as Omission Ratio (Hovy 1999). An evaluation of a summarization system must at least in some way tackle both of these properties.

A first broad division in methods for evaluation automatic text summarization systems, as well as many other systems, is the division into intrinsic and extrinsic evaluation methods (Spark-Jones and Galliers 1995).

1.3.1 Intrinsic Evaluation

Intrinsic evaluation measures the system in of itself. This is often done by comparison to some gold standard, which can be made by a reference summarization system or, more often than not, is man-made using informants. Intrinsic evaluation has mainly focused on the coherence and informativeness of summaries.

1.3.1.1 Summary Coherence

Summaries generated through extraction-based methods (cut-and-paste operations on phrase, sentence or paragraph level) sometimes suffer from parts of the summary being extracted out of context, resulting in coherence problem (e.g. dangling anaphors or gaps in the rhetorical structure of the summary). One way to measure this is to let subjects rank or grade summary sentences for coherence and then compare the grades for the summary sentences with the scores for reference summaries, with the scores for the source sentences, or for that matter with the scores for other summarization systems.

1.3.1.2 Summary Informativeness

One way to measure the informativeness of the generated summary is to compare the generated summary with the text being summarized in an effort to assess how much information from the source is preserved in the condensation. Another is to compare the generated summary with a reference summary, measuring how much information in the reference summary is present in the generated summary. For single documents traditional precision and recall figures can be used to assess performance as well as utility figures (section 1.3.1.5) and content based methods (section 1.3.1.6).

1.3.1.3 Sentence Precision and Recall

Sentence recall measures how many of the sentences in the reference summary that are present in the generated summary and in a similar manner precision[4] can be calculated. Precision and recall are standard measures for Information Retrieval and are often combined in a so-called F-score (Van Rijsbergen 1979). The main problems with these measures for text summarization is that they are not capable of distinguishing between many possible, but equally good, summaries and that summaries that differ quite a lot content wise may get very similar scores.

1.3.1.4 Sentence Rank

Sentence rank is a more fine-grained approach than precision and recall (P&R), where the reference summary is constructed by ranking the sentences in the source

[4]Precision is in this case defined as the number of sentences in the generated summary that are present in the reference summary.

text by worthiness of inclusion in a summary of the text. Correlation measures can then be applied to compare the generated summary with the reference summary. As in the case of P&R this method mainly applies to extraction based summaries, even if standard methods of sentence alignment with abstracts can be applied (Marcu 1999, Jing and McKeown 1999).

1.3.1.5 The Utility Method

The utility method (UM) (Radev et al. 2000) allows reference summaries to consist of extraction units (sentences, paragraphs etc.) with fuzzy membership in the reference summary. In UM the reference summary contains all the sentences of the source document(s) with confidence values for their inclusion in the summary. Furthermore, the UM methods can be expanded to allow extraction units to exert negative support on one another. This is especially useful when evaluating multi-document summaries, where in case of one sentence making another redundant it can automatically penalize the evaluation score, i.e. a system that extracts two or more "equivalent" sentences gets penalized more than a system that extracts only one of the aforementioned sentences and a, say, less informative sentence (i.e. a sentence that has a lower confidence score).

This method bears many similarities to the Majority Vote method (Hassel 2003) in that it, in contrast to P&R and Percent Agreement, allows summaries to be evaluated at different compression rates. UM is mainly useful for evaluating extraction based summaries, more recent evaluation experiments has led to the development of the Relative Utility metric (Radev and Tam 2003).

1.3.1.6 Content Similarity

Content similarity measures (Donaway et al. 2000) can be applied to evaluate the semantic content in both extraction based summaries and true abstracts. One such measure is the Vocabulary Test (VT) where standard Information Retrieval methods (see Salton and McGill 1983) are used to compare term frequency vectors calculated over stemmed or lemmatized summaries (extraction based or true abstracts) and reference summaries of some sort. Controlled thesauri and "synonym sets" created with Latent Semantic Analysis (Landauer et al. 1998) or Random Indexing (Kanerva et al. 2000, Sahlgren 2001) can be used to reduce the terms in the vectors by combining the frequencies of terms deemed synonymous, thus allowing for greater variation among summaries. This is especially useful when evaluating abstracts.

The disadvantage of these methods is, however, that they are quite sensitive to negation and word order differences. With LSA[5] or RI[6] one must also be aware of the fact that these methods do not necessarily produce true synonym sets, these

[5] Latent Semantic Analyzis; sometimes also referred to as Latent Semantic Indexing.
[6] Random Indexing.

sets typically also include antonyms, hyponyms and other terms that occur in similar semantic contexts (on word or document level for RI and document level for LSA). These methods are however useful for extraction based summaries where little rewriting of the source fragments is done, and when comparing fragmentary summaries, such as key phrase summaries.

1.3.1.7 BLEU Scores

The idea here is that, as well as there may be many "perfect" translations of a given source sentence, there may be several equally good summaries for a single source document. These summaries may vary in word or sentence choice, or in word or sentence order even when they use the same words/sentences. Yet humans can clearly distinguish a good summary from a bad one.

The recent adoption of BLEU/NIST[7] scores (Papineni et al. 2001, NIST 2002) by the MT community for automatic evaluation of Machine Translation, Lin and Hovy (2003) have applied the same idea to the evaluation of summaries. They used automatically computed accumulative n-gram matching scores (NAMS) between ideal summaries and system summaries as a performance indicator. Only content words were used in forming n-grams and n-gram matches between the summaries being compared where treated as position independent. For comparison, IBM's BLEU evaluation script was also applied to the same summary set. However, this showed that direct application of the BLEU evaluation procedure does not always give good results.

1.3.2 Extrinsic Evaluation

Extrinsic evaluation on the other hand measures the efficiency and acceptability of the generated summaries in some task, for example relevance assessment or reading comprehension. Also, if the summary contains some sort of instructions, it is possible to measure to what extent it is possible to follow the instructions and the result thereof. Other possible measurable tasks are information gathering in a large document collection, the effort and time required to post-edit the machine generated summary for some specific purpose, or the summarization system's impact on a system of which it is part of, for example relevance feedback (query expansion) in a search engine or a question answering system.

Several game like scenarios have been proposed as surface methods for summarization evaluation inspired by different disciplines, among these are The Shannon Game (information theory), The Question Game (task performance), The Classification/Categorization Game and Keyword Association (information retrieval).

[7] Based on the superior F-ratios of information-weighted counts and the comparable correlations, a modification of IBM's formulation of the score was chosen as the evaluation measure that NIST will use to provide automatic evaluation to support MT research.

1.3.2.1 The Shannon Game

The Shannon Game, which is a variant of Shannon's measures in Information Theory (Shannon 1948), is an attempt to quantify information content by guessing the next token, e.g. letter or word, thus recreating the original text. The idea has been adapted from Shannon's measures in Information Theory where you ask three groups of informants to reconstruct important passages from the source article having seen either the full text, a generated summary, or no text at all. The information retention is then measured in number of keystrokes it takes to recreate the original passage. Hovy (see Hovy and Marcu 1998) has shown that there is a magnitude of difference across the three levels (about factor 10 between each group). The problem is that Shannon's work is relative to the person doing the guessing and therefore implicitly conditioned on the reader's knowledge. The information measure will infallibly change with more knowledge of the language, the domain, etc.

1.3.2.2 The Question Game

The purpose of the Question Game is to test the readers' understanding of the summary and its ability to convey key facts of the source article. This evaluation task is carried out in two steps. First the testers read the source articles, marking central passages as they identify them. The testers then create questions that correspond to certain factual statements in the central passages. Next, assessors answer the questions 3 times: without seeing any text (baseline 1), after seeing a system generated summary, and after seeing original text (baseline 2). A summary successfully conveying the key facts of the source article should be able to answer most questions, i.e. being closer to baseline 2 than baseline 1. This evaluation scheme has for example been used in the TIPSTER SUMMAC text summarization evaluation Q&A[8] task, where Mani et al. (1998) found an informativeness ratio of accuracy to compression of about 1.5.

1.3.2.3 The Classification Game

In the classification game one tries to compare classifiability by asking assessors to classify either the source documents (testers) or the summaries (informants) into one of N categories. Correspondence of classification of summaries to originals is then measured. An applicable summary should be classified into the same category as its source document. Two versions of this test were run in SUMMAC (Mani et al. 1998).

[8] Question and Answering; a scenario where a subject is set to answer questions about a text given certain conditions, for example a summary of the original text.

1.3.2.4 Keyword Association

Keyword association is an inexpensive, but somewhat shallower, approach that relies on keywords associated (either manually or automatically) to the documents being summarized. For example Saggion and Lapalme (2000) presented human judges with summaries generated by their summarization system together with five lists of keywords taken from the source article as presented in the publication journal. The judges were then given the task to associate the each summary with the correct list of keywords. If successful the summary was said to cover the central aspects of the article since the keywords associated to the article by the publisher were content indicative. Its main advantage is that it requires no cumbersome manual annotation.

1.3.3 Evaluation Tools

In order to allow a more rigorous and repeatable evaluation procedure, partly by automating the comparison of summaries, it is advantageous to build an extract corpus containing originals and their extracts, i.e. summaries strictly made by extraction of whole sentences from an original text. Each extract, whether made by a human informant or a machine, is meant to be a true summary of the original, i.e. to retain the meaning of the text as good as possible. Since the sentence units of the original text and the various summaries are known entities, the construction and analysis of an extract corpus can almost completely be left to computer programs, if these are well-designed. A number of tools have been developed for these purposes.

1.3.3.1 Summary Evaluation Environment

Summary Evaluation Environment (SEE; Lin 2001) is an evaluation environment in which assessors can evaluate the quality of a summary, called the peer text, in comparison to a reference summary, called the model text. The texts involved in the evaluation are pre-processed by being broken up into a list of segments (phrases, sentences, clauses, etc.) depending on the granularity of the evaluation. For example, when evaluating an extraction based summarization system that works on the sentence level, the texts are pre-processed by being broken up into sentences.

During the evaluation phase, the two summaries are shown in two separate panels in SEE and interfaces are provided for assessors to judge both the content and the quality of summaries. To measure content, the assessor proceeds through the summary being evaluated, unit by unit, and clicks on one or more associated units in the model summary. For each click, the assessor can specify whether the marked units express all, most, some or hardly any of the content of the clicked model unit. To measure quality, assessors rate grammaticality, cohesion, and coherence at five different levels: all, most, some, hardly any, or none. Quality is assessed both for each unit of the peer summary and for overall quality of the peer summary (coherence, length, content coverage, grammaticality, and organization of the peer

text as a whole). Results can, of course, be saved and reloaded and altered at any time.

A special version of SEE 2.0 has for example been used in the DUC-2001 (Harman and Marcu 2001) intrinsic evaluation of generic news text summarization systems (Lin and Hovy 2002). In DUC-2001 the sentence was used as the smallest unit of evaluation.

1.3.3.2 MEADeval

MEADeval (Winkel and Radev 2002) is a Perl toolkit for evaluating MEAD- and DUC-style extracts, by comparison to a reference summary (or "ideal" summary). MEADeval operates mainly on extract files, which describe the sentences contained in an extractive summary: which document each sentence came from and the number of each sentence within the source document – but it can also perform some general content comparison. It supports a number of standard metrics, as well as some specialized (see table 1.1).

A strong point of Perl, apart from platform independency, is the relative ease of adapting scripts and modules to fit a new summarization system. MEADeval has, for example, been successfully applied to summaries generated by a Spanish lexical chain summarizer and the SweSum[9] summarizer in a system-to-system comparison against model summaries (see Alonso i Alemany and Fuentes Fort 2003).

Extracts only	General text
precision	unigram overlap
recall	bigram overlap
normalized precision[10]	cosine[11]
normalized recall[12]	simple cosine[13]
kappa[14]	
relative utility[15]	
normalized relative utility	

Table 1.1: Metrics supported by MEADeval.

[9]SweSum mainly being a Swedish language text summarizer, also supports plug-in lexicons and heuristics for other languages, among these Spanish.

[10]Like precision, but normalized by the length (in words) of each sentence.

[11]The 2-norm (Euclidean Distance) between two vectors.

[12]Like recall, but normalized by the length (in words) of each sentence.

[13]Cosine without adjustments for Inverse Document Frequency (IDF).

[14]The simple kappa coefficient is a measure of interrater agreement compared to what could be expected due to chance alone.

[15]The Relative Utility and Normalized Relative Utility metrics are described in Radev and Tam (2003), also see section 1.3.1.5.

1.3.3.3 ISI ROUGE - Automatic Summary Evaluation Package

ROUGE, short for Recall-Oriented Understudy for Gisting Evaluation, by Lin (2003) is a very recent adaption of the IBM BLEU (see section 1.3.1.7) for Machine Translation that uses unigram co-occurrences between summary pairs. According to in-depth studies based on various statistical metrics and comparison to the results DUC-2002 (Hahn and Harman 2002), this evaluation method correlates surprisingly well with human evaluation (Lin and Hovy 2003).

ROUGE is recall oriented, in contrast to the precision oriented BLEU script, and separately evaluates 1, 2, 3, and 4-grams. Also, ROUGE does not apply any length penalty (brevity penalty), which is natural since text summarization involves compression of text and thus rather should reward shorter extract segment as long as they score well for content. ROUGE has been verified for extraction based summaries with a focus on content overlap. No correlation data for quality has been found so far.

1.3.3.4 KTH eXtract Corpus and Tools

At the Royal Institute of Technology (KTH), Hassel has developed a tool for collection of extract based summaries provided by human informants and semi-automatic evaluation of machine generated extracts (Hassel 2003, Dalianis et al. 2004) in order to easily evaluate the SweSum summarizer (Dalianis 2000). The KTH eXtract Corpus (KTHxc) contains a number of original texts and several manual extracts for each text. The tool assists in the construction of an extract corpus by guiding the human informant creating a summary in such a way that only full extract units (most often sentences) are selected for inclusion in the summary. The interface allows for the reviewing of sentence selection at any time, as well as reviewing of the constructed summary before submitting it to the corpus.

Once the extract corpus is compiled, the corpus can be analysed automatically in the sense that the inclusion of sentences in the various extracts for a given source text can easily be compared. This allows for a quick adjustment and evaluation cycle in the development of an automatic summarizer. One can, for instance, adjust parameters of the summarizer and directly obtain feedback of the changes in performance, instead of having a slow, manual and time consuming evaluation.

The KTH extract tool gathers statistics on how many times a specific extract unit from a text has been included in a number of different summaries. Thus, an ideal summary, or reference summary, can be composed using only the most frequently chosen sentences. Further statistical analysis can evaluate how close a particular extract is to the ideal one. The tool also has the ability to output reference summaries constructed by Majority Vote in the format SEE (described in section 1.3.3.1) uses for human assessment.

Obviously, the KTHxc tool could easily be ported to other languages and so far corpus collection and evaluation has been conducted for Swedish as well as

Danish. The University of Bergen has initiated a similar effort for Norwegian and has developed some similar tools (Dalianis et al. 2004).

1.3.4 Famous Last Words

Most automatic text summarization systems today are extraction based systems. However, some recent work directed towards post-editing of extracted segments, e.g. sentence/phrase reduction and combination, thus at least creating the illusion of abstracting in some sense, leads to the situation where evaluation will have to tackle comparison of summaries that do not only differ in wording but maybe also in specificity and bias.

Furthermore, in automatic text summarization, as well as in for example machine translation, there may be several equally good summaries (or in the case of MT - translations) for one specific source text, effectively making evaluation against one rigid reference text unsatisfactory. Also, evaluation methods that allow for evaluation at different compression rates should be favored as experiments have shown that different compression rates are optimal for different text types or genres, or even different texts within a text type or genre. The automatic evaluation methods presented in this paper mainly deal with content similarity between summaries. Summary quality must still be evaluated manually.

Today, there is no single evaluation scheme that provides for all these aspects of the evaluation, so a mixture of methods described in this paper should perhaps be used in order to cover as many aspects as possible thus making the results comparable with those of other systems, shorten the system development cycle and support just-in-time comparison among different summarization methods. Clearly some sort of standardized evaluation framework is heavily in need in order to ensure replication of results and trustworthy comparison among summarization systems.

However, it is also important to keep users in the loop, at least in the end stages of system evaluation. One must never forget the target of the summaries being produced.

1.4 Overview of the Papers Included in this Thesis

1.4.1 Paper 1.

Internet as Corpus - Automatic Construction of a Swedish News Corpus
(Hassel 2001a)
In order to evaluate automatic summarizers or information extraction and retrieval tools, but also to train these tools to make their performance better, one needs to have a corpus. For English and other widespread languages there are freely available corpora, but this is not the case for Swedish. Therefore we needed to collect a well balanced corpus mainly consisting of news text in Swedish. We used the Internet as our source. In total we automatically collected approximately 200,000 news articles between May 2000 to June 2002 containing over 10 million words. The news

texts collected were news flashes and press releases from large Swedish newspapers like Svenska Dagbladet, Dagens Nyheter and Aftonbladet, business and computer magazines as well as press releases from humanitary organizations like Amnesty International, RFSL[16] and authorities like Riksdagen.[17]

The KTH News Corpus has since been used to train a Named Entity tagger (Dalianis and Åström 2001), that also is used in the SweSum text summarizer (see Paper 4). The KTH News Corpus has also been used to evaluate the robustness of the SweSum summarizer by summarizing Swedish news text collected by the Business Intelligence tool NyhetsGuiden, "NewsGuide" (see Hassel 2001b), that actually is an interface to the KTH News Corpus tool.

Furthermore, the corpus is currently used in experiments concerning the training of a Random Indexer (see Karlgren and Sahlgren 2001, Sahlgren 2001) for experiments where "synonym sets", or rather *semantic sets*, are used to augment the frequency count. Gong and Liu (2001) have carried out similar experiments, but using LSA for creating their sets, where they used the semantic sets to pin point the topically central sentences, using each set only once in an attempt to avoid redundancy. LSA, however, is a costly method for building these types of sets, and if RI performs equally or better there would be much benefit.

1.4.2 Paper 2.

Improving Precision in Information Retrieval for Swedish using Stemming (Carlberger, Dalianis, Hassel, and Knutsson, 2001)
An early stage of the KTH News Corpus[18], as from October 2000, was used to evaluate a stemmer for Swedish and its inpact on the performance of a search engine, SiteSeeker[19]. The work was completely manual in the sense that three persons first had to each annotate ≈33[20] randomly selected texts from the KTH News Corpus each, and for each text construct a relevant/central question with corresponding answer, in total 100 texts and 100 answers. The next step was to use the search engine formulating queries to retrieve information, i.e to find answers to our allotted questions, with and without stemming in hope that one would find answers to the questions. Here the questions where swapped one step so that no one formulated queries concerning their own questions. The last step was to assess the answers of the partner, again a swap was made, to find out the precision and recall. This time consuming and manual evaluation schema showed us that precision and relative recall improved with 15 respectively 18 percent for Swedish in information retrieval using stemming.

[16]Riksförbundet För Sexuellt Likaberättigande; A gay, lesbian, bisexual and transgendered lobby organization.
[17]The Swedish Parliament.
[18]The corpus at this stage contained about 54,000 texts.
[19]http://www.euroling.se/siteseeker/
[20]One person actually had to do 34 questions/queries/evaluations at each corresponding stage, but this "extra load" was shifted with each stage.

The Swedish stemmer was also scheduled to be used as a postprocessor for SweSum, after using a stop list to find all keywords in a text and consequently improve the performance of SweSum. This track was however abandoned and an upcoming effort aims at connecting SweSum to a Granska server (a more modern incarnation of Domeij et al. 1999) for lemmatization and PoS tagging.

1.4.3 Paper 3.

Development of a Swedish Corpus for Evaluating Summarizers and other IR-tools (Dalianis and Hassel 2001)
SweSum - the first automatic text summarizer for Swedish news text was constructed in 2000 (Dalianis 2000). SweSum works in the text extractor paradigm. This is to extract the most significant, in our case, sentences from a text and from them create a new shorter non redundant text. This paradigm is the most common among automatic text summarizers.

We first made an intrinsic (see section 1.3.1) qualitative subjective evaluation of SweSum using the techniques described in Firmin and Chrzanowski (1999). Our informants were students at our Human Language Technology course. The students had to judge the summarized texts, by ocular inspection, and decide if the text in question was perceived as well formed in terms of coherence and content. That is, the students rated the SweSum generated summaries for summary coherence (see section 1.3.1.1) and summary informativeness (see section 1.3.1.2). We found that the coherence of the text was intact at 30 percent compression rate and that the information content was intact at 25 percent compression rate.

The following year we improved our experiment by making a more objective extrinsic (see section 1.3.2) evaluation of our text summarizer, SweSum. This time we used our 100 annotated news texts and corresponding queries (see Paper 2). Again we let students attending our Human Language Technology course execute SweSum with increasing compression rate on the 100 manually annotated texts, in an effort to find answers to the predefined questions in a Question Game-like scenario (see section 1.3.2.2). The results showed that at 40 percent summarization/compression rate the correct answer rate was 84 percent. Both these methods needed a large human effort, a more efficient evaluation framework was clearly in demand.

1.4.4 Paper 4.

Exploitation of Named Entities in Automatic Text Summarization for Swedish (Hassel 2003)
In Dalianis and Åström (2001) a Named Entity recognizer called SweNam was constructed for Swedish NER.[21] Named Entity recognition is the method that from a text extracts names of persons, organizations, locations and possibly also dates

[21] Named Entity Recognition

and time. SweNam was trained on the KTH News corpus. We were keen on finding out if Named Entity recognition could improve automatic text summarization. Therefore we connected the original SweSum with SweNam were SweNam acted as a preprocessor to SweSum.

We were not completely happy with our extrinsic Question and Answering scheme (see Paper 3) in evaluating our text summarizer and wanted to bring evaluation a step further to a more intrinsic evaluation. Therefore we created the KTH eXtract Corpus (KTHxc) - a corpus of manual extracts from original texts that could be used as a Gold Standard. A Gold Standard summary, or ideal extract summary, can then repeatedly be compared with automatic summaries generated by SweSum. A group of human informants were presented news articles one at a time in random order so they could select sentences for extraction. The submitted extracts were allowed to vary between 5 and 60 percent of the original text length. The advantage of having extracts was that we could directly compare what humans selected as informative or good sentences to include in an extract summary with what the machine, i.e. SweSum, selected. Different settings in and incarnations of SweSum can thus be easily compared. Even though the continuous growth of the corpus is necessary in order to avoid overfitting, the effort of collecting the corpus and the repeated use of it in evaluation is still less than previous attempts.

The results of one such evaluation showed that Named Entities tend to prioritize sentences with a high information level on the categories used. They tend to prioritize elaborative sentences over introductory and thus sometimes are responsible for serious losses of sentences that give background information. Our finding were that Named entity recognition must be used with consideration so it will not make the summary too information intense and consequently difficult to read. Also, it may actually in extreme cases lead to condensation of redundancy in the original text.

1.5 Main Contribution of the Licentiate Thesis

This licentiate thesis has created the starting point of a framework for evaluation of text summarization, information extraction and retrieval tools for Swedish. We have created a corpus in Swedish, the KTH News Corpus, that has been used by several researchers, both for evaluation and training of human language technology tools (besides SweSum; a spelling and grammar checker, taggers, parsers and text clusterers as well as a search enginge spiffed up with stemming). Part of this corpus[22] has been manually tagged with topically relevant questions and corresponding answers, keywords and Named Entities. A second corpus, the KTH eXtract Corpus, consisting of full texts and sentence selection statistics representing summaries of the full texts has also been contructed, so far for Swedish and Danish. This corpus is mainly aimed at the evaluation of sentence extraction systems. We have specifically carried out the work on human language technology tools for Swedish,

[22] 100 randomly chosen news texts.

which has been lacking these types of tools, but also we have spread the technique to Danish and to some extent to Norwegian.

We have slowly but steadily moved from manual and time consuming evaluation of the summarizer to a semi automatic evaluation where we have a ready extract corpus that can be used repeatedly to evaluate text summarizers.

1.6 Concluding Remarks and Future Directions

We have obtained a higher knowledge on the nature of text summarizers, how they behave and their specific problems. We have discussed and analyzed their architecture, the use of keywords as well as Named Entities for topic identification and also the problems of text segmentation (i.e. finding clause and sentence boundaries). The question of porting a summarizer engine to other languages[23] has also been adressed within the frameworks of the Majordome project and the ScandSum network. We have also taken a closer look at how text summarizers should be evaluated, both extrinsically and intrinsically. This is the work comprised and summarized in this licentiate thesis.

Regarding architecture, we are planning to go towards a more fine grained analysis of the original text structure down to word and clause level, and not to use intact extracted sentences as the neccessary resulting unit.

We also plan to make the process of evaluation nearly completely automatic, such that one can change the architecture of the summarizer and directly assess the impact on performance of that change. We will also transfer this methodology to create an extract corpus to other languages such that one can evaluate a summarizer without having more than basic knowledge in the language (naturally assuming the expressed reliability of the corpus).

[23] The lexicon based version of SweSum, which uses language specific keyword and abbreviation lexicons as well as heuristics, has so far been successfully ported to English, Spanish, French, German, Danish, Norwegian and most recently Farsi (Dalianis et al. 2004, Mazdak 2004).

1.7 Bibliography

Alonso i Alemany, L. and M. Fuentes Fort (2003). Integrating Cohesion and Coherence for Automatic Summarization. In *Proceedings of the 11th Meeting of the European Chapter of the Association for Computational Linguistics*, Budapest, Hungary.

ANSI (1979). American National Standard for Writing Abstracts. Technical report, American National Standards Institute, Inc., New York, NY. ANSI Z39.14.1979.

Baldwin, B., R. Donaway, E. Hovy, E. Liddy, I. Mani, D. Marcu, K. McKeown, V. Mittal, M. Moens, D. Radev, K. Sparck-Jones, B. Sundheim, S. Teufel, R. Weischedel, and M. White (2000). An Evaluation Road Map for Summarization Research. http://www-nlpir.nist.gov/projects/duc/papers/summarization.roadmap.doc.

Borko, H. and C. Bernier (1975). *Abstracting Concepts and Methods*. Academic Press, New York.

Burton, N. and J. Licklider (1955). Long-range constraints in the statistical structure of printed English. *American Journal of Psychology*, 68:650–655.

Carlberger, J., H. Dalianis, M. Hassel, and O. Knutsson (2001). Improving Precision in Information Retrieval for Swedish using Stemming. In *Proceedings of NODALIDA'01 - 13th Nordic Conference on Computational Linguistics*, Uppsala, Sweden.

Cremmins, E. T. (1996). *The Art of Abstracting*. Information Resources Press, Arlington, VA, 2nd edition.

Dalianis, H. (2000). SweSum - A Text Summarizer for Swedish. Technical report, KTH NADA, Sweden.

Dalianis, H. and M. Hassel (2001). Development of a Swedish Corpus for Evaluating Summarizers and other IR-tools. Technical report, KTH NADA, Sweden.

Dalianis, H., M. Hassel, K. de Smedt, A. Liseth, T. C. Lech, and J. Wedekind (2004). Porting and evaluation of automatic summarization. In Holmboe, H. (editor), *Nordisk Sprogteknologi 2003: Årbog for Nordisk Språkteknologisk Forskningsprogram 2000-2004*. Museum Tusculanums Forlag.

Dalianis, H. and E. Åström (2001). SweNam - A Swedish Named Entity recognizer. Its construction, training and evaluation. Technical report, TRITA-NA-P0113, IPLab-189, KTH NADA.

Domeij, R., O. Knutsson, J. Carlberger, and V. Kann (1999). Granska - An efficient hybrid system for Swedish grammar checking. In *Proceedings of NoDaLiDa'99 - 12th Nordic Conference on Computational Linguistics*.

Donaway, R. L., K. W. Drummey, and L. A. Mather (2000). A Comparison of Rankings Produced by Summarization Evaluation Measures. In Hahn, U., C.-Y. Lin, I. Mani, and D. R. Radev (editors), *Proceedings of the Workshop on Automatic Summarization at the 6th Applied Natural Language Processing Conference and the 1st Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 69–78. Association for Computational Linguistics.

Edmundson, H. P. (1969). New Methods in Automatic Extracting. *Journal of the Association for Computing Machinery*, 16(2):264–285.

Firmin, T. and M. J. Chrzanowski (1999). An Evaluation of Automatic Text Summarization Systems. In Mani, I. and M. T. Maybury (editors), *Advances in Automatic Text Summarization*, pp. 325–336. MIT Press.

Gong, Y. and X. Liu (2001). Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New Orleans, LA.

Hahn, U. and D. Harman (editors) (2002). *Proceedings of the 2nd Document Understanding Conference*. Philadelphia, PA.

Harman, D. and D. Marcu (editors) (2001). *Proceedings of the 1st Document Understanding Conference*. New Orleans, LA.

Hassel, M. (2001a). Internet as Corpus - Automatic Construction of a Swedish News Corpus. In *Proceedings of NODALIDA'01 - 13th Nordic Conference on Computational Linguistics*, Uppsala, Sweden.

Hassel, M. (2001b). *newsAgent* - A Tool for Automatic News Surveillance and Corpora Building. NUTEK report, http://www.nada.kth.se/~xmartin/papers/Nutek.pdf.

Hassel, M. (2003). Exploitation of Named Entities in Automatic Text Summarization for Swedish. In *Proceedings of NODALIDA'03 - 14th Nordic Conference on Computational Linguistics*, Reykjavik, Iceland.

Hovy, E. (editor) (1999). *Multilingual Information Management: Current Levels and Future Abilities. Chapter 3 Cross-lingual Information Extraction and Automated Text Summarization.*

Hovy, E. and C.-Y. Lin (1997). Automated Text Summarization in SUMMARIST. In *Proceedings of the ACL97/EACL97 Workshop on Intelligent Scalable Text Summarization*, Madrid, Spain.

Hovy, E. and D. Marcu (1998). Automated Text Summarization Tutorial at COLING/ACL'98. http://www.isi.edu/~marcu/acl-tutorial.ppt.

ISO 215:1986 (1986). Documentation – Presentation of Contributions to Periodicals and Other Serials. ISO 215:1986. Technical report, International Organisation for Standardisation.

Jing, H. (2000). Sentence Reduction for Automatic Text Summarization. In *Proceedings of the 6th Applied Natural Language Processing Conference*, pp. 310–315, Seattle,WA.

Jing, H. and K. R. McKeown (1999). The Decomposition of Human-Written Summary Sentences. In Hearst, M., G. F., and R. Tong (editors), *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 129–136, University of California, Beekely.

Jing, H. and K. R. McKeown (2000). Cut and Paste-Based Text Summarization. In *Proceedings of the 6th Applied Natural Language Processing Conference and the 1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pp. 178–185, Seattle, WA.

Kanerva, P., J. Kristoferson, and A. Holst (2000). Random Indexing of text samples for Latent Semantic Analysis. In Gleitman, L. and A. Josh (editors), *Proceedings 22nd Annual Conference of the Cognitive Science Society*, Pennsylvania.

Karlgren, J. and M. Sahlgren (2001). Vector-based Semantic Analysis using Random Indexing and Morphological Analysis for Cross-Lingual Information Retrieval. Technical report, SICS, Sweden.

Landauer, T. K., P. W. Foltz, and D. Laham (1998). Introduction to Latent Semantic Analysis. *Discourse Processes*, 25:259–284.

Lin, C.-Y. (2001). Summary Evaluation Environment. http://www.isi.edu/~cyl/SEE.

Lin, C.-Y. (2003). ROUGE: Recall-oriented understudy for gisting evaluation. http://www.isi.edu/~cyl/ROUGE/.

Lin, C.-Y. and E. Hovy (2000). The automated acquisition of topic signatures for text summarization. In *Proceedings of the 18th COLING Conference*, Saarbrücken, Germany.

Lin, C.-Y. and E. Hovy (2002). Manual and Automatic Evaluation of Summaries. In *Proceedings of the Workshop on Multi-Document Summarization Evaluation of the 2nd Document Understanding Conference at the 4Oth Meeting of the Association for Computational Linguistics*, Philadelphia, PA.

Lin, C.-Y. and E. Hovy (2003). Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. In *Proceedings of 2003 Language Technology Conference (HLT-NAACL 2003)*, Edmonton, Canada.

Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal of Research Development*, 2(2):159–165.

Mani, I., D. House, G. Klein, L. Hirshman, L. Orbst, T. Firmin, M. Chrzanowski, and B. Sundheim (1998). The TIPSTER SUMMAC Text Summarization Evaluation. Technical Report MTR 98W0000138, The Mitre Corporation, McLean, Virginia.

Mani, I. and M. T. Maybury (editors) (1999). *Advances in Automatic Text Summarization*. MIT Press, Cambridge, MA.

Marcu, D. (1999). The Automatic Construction of Large-Scale Corpora for Summarization Research. In Hearst, M., G. F., and R. Tong (editors), *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 137–144, University of California, Berkely.

Mazdak, N. (2004). FarsiSum - A Persian Text Summarizer. Master's thesis in computational linguistics, Department of Linguistics, Stockholm University, Sweden.

Morris, A., G. Kasper, and D. Adams (1992). The Effects and Limitations of Automated Text Condensing on Reading Comprehension Performance. *Information Systems Research*, 3(1):17–35.

NIST (2002). Automatic Evaluation of Machine Translation Quality using N-gram Co-Occurrence Statistics. http://www.nist.gov/speech/tests/mt/doc/ngram-study.pdf.

Papineni, K., S. Roukos, T. Ward, and W.-J. Zhu (2001). BLEU: A Method for Automatic Evaluation of Machine Translation. Research Report RC22176, IBM.

Radev, D. R., H. Jing, and M. Budzikowska (2000). Centroid-Based Summarization of Multiple Documents: Sentence Extraction, Utility-Based Evaluation, and User Studies. In Hahn, U., C.-Y. Lin, I. Mani, and D. R. Radev (editors), *Proceedings of the Workshop on Automatic Summarization at the 6th Applied Natural Language Processing Conference and the 1st Conference of the North American Chapter of the Association for Computational Linguistics*, Seattle, WA.

Radev, D. R. and D. Tam (2003). Single-Document and Multi-Document Summary Evaluation via Relative Utility. In *Poster Session, Proceedings of the ACM CIKM Conference*, New Orleans, LA.

Saggion, H. and G. Lapalme (2000). Concept Identification and Presentation in the Context of Technical Text Summarization. In Hahn, U., C.-Y. Lin, I. Mani, and D. R. Radev (editors), *Proceedings of the Workshop on Automatic Summarization at the 6th Applied Natural Language Processing Conference and the 1st Conference of the North American Chapter of the Association for Computational Linguistics*, Seattle, WA, USA. Association for Computational Linguistics.

Sahlgren, M. (2001). Vector-Based Semantic Analysis: Representing word meanings based on random labels. In *Proceedings of Semantic Knowledge Acquisition and Categorisation Workshop at ESSLLI'01*, Helsinki, Finland.

Salton, G. (1988). *Automatic Text Processing*. Addison-Wesley Publishing Company.

Salton, G. and M. J. McGill (1983). *Introduction to Modern Information Retrieval*. McGraw-Hill Book Company.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27.3-4:379–423,623–656.

Shannon, C. E. (1951). Prediction and Entropy of Printed English. *The Bell System Technical Journal*, 30:50–64.

Sparck-Jones, K. (1999). Automatic Summarizing: Factors and Directions. In Mani, I. and M. T. Maybury (editors), *Advances in Automatic Text Summarization*, pp. 1–13. The MIT Press.

Spark-Jones, K. and J. R. Galliers (1995). *Evaluating Natural Language Processing Systems: An Analysis and Review*. Number 1083 in Lecture Notes in Artificial Intelligence. Springer.

Van Rijsbergen, C. J. (1979). *Information Retrieval, 2nd edition*. Dept. of Computer Science, University of Glasgow.

Winkel, A. and D. Radev (2002). MEADeval: An evaluation framework for extractive summarization. http://perun.si.umich.edu/clair/meadeval/.

Chapter 2

Paper 1

Internet as Corpus

Automatic Construction of a Swedish News Corpus

Martin Hassel
KTH NADA
Royal Institute of Technology
100 44 Stockholm, Sweden
xmartin@nada.kth.se

Abstract

This paper describes the automatic building of a corpus of short Swedish news texts from the Internet, its application and possible future use. The corpus is aimed at research on Information Retrieval, Information Extraction, Named Entity Recognition and Multi Text Summarization. The corpus has been constructed by using an Internet agent, the so called *newsAgent*, downloading Swedish news text from various sources. A small part of this corpus has then been manually tagged with keywords and named entities. The newsAgent is also used as a workbench for processing the abundant flows of news texts for various users in a customized format in the application *Nyhetsguiden*.

Keywords: News Text, Corpus Aquisition, Internet, Swedish

2.1 Introduction

Two years ago we built an automatic text summarizer called SweSum (Dalianis 2000) for Swedish text. We wanted to evaluate SweSum but there were no tagged Swedish corpus available to evaluate text summarizers or information retrieval tools processing Swedish as it is for the English speaking community, mainly through the TREC (Vorhees and Tice 2000), MUC and TIPSTER-SUMMAC evaluation conferences (Mani et al. 1998, Krenn and Samuelsson 1997). The purpose of this project[1] was to construct test bed for new natural language technology tools, i.e. *automatic text summarization, named entity tagging, stemming, information retrieval/extraction,* etc. In the process of building this system, Nyhetsguiden (Hassel 2001), we also made it capable of gathering the news texts into a corpus, a

[1]This project is supported by NUTEK (Swedish board for Industrial and Technical Development) FavorIT programme in cooporation with EuroSeek AB.

corpus we have used to train and evaluate such tools as mentioned above. As this corpus is aimed at research on information and language technology applied on redundant text, the system does not, contrary to (Hofland 2000), remove duplicated concordance lines.

2.2 Nyhetsguiden - A User Centred News Delivery System

The system has a modular design and consists of three parts, the user interface, the user database and the main application, newsAgent. Being modular, the system can be run as a distributed system or on a single web server. When run as a distributed system, at least newsAgent must be run on a computer with Internet access. The user interface (Nyhetsguiden) and the user database can reside on either an Internet or Intranet capable server depending on the desired public access to the system. newsAgent is the core of the system and is basically a web spider that is run in a console window. The spider is implemented in Perl, which makes it platform independent, that is, it can run on any platform running Perl (Unix/Linux, Windows, Macintosh, BeOS, Amiga, etc). On intervals of 3-5 minutes newsAgent searches the designated news sources (Appendix A) for new news texts, that is news texts not seen by the system before. When a new news text is encountered it is fetched, the actual news text and accompanying illustrations are extracted (by removing navigation panels, banners, tables of links, etc). The resulting document is then passed through the system and, depending on configuration; stored, summarized and routed to the end recipient.

2.3 Construction of a Corpus of Swedish News Texts

Traditionally it has been hard work constructing a corpus of news text. In Sweden there are no newspapers that on a yearly basis offer their paper in digital form,[2] as some foreign newspapers do (for example Wall Street Journal), meaning that obtaining this material has to be done on demand. Many Swedish newspapers are, when inquired, unwilling to release texts from their archives for research purposes, and even when they do, it is often the question of a small amount of news texts with an age of several years. This may potentially lead to the exclusion of contemporary words and giving unusually high, or low, occurrence frequencies to words related to phenomena limited to a certain period of time.

In the past, the solution would be to collect newspapers in their paper form and type or scan (using a Optical Character Recognition program) them in order to convert them to a format manageable by computers.

The World Wide Web is, on the other hand, today a large collection of texts written in different languages and thus giving an abundant resource for language

[2]We have as yet only been able to aquire 1995 years years issue of Svenska Dagbladet (SvD) also the Scarrie Swedish News Corpus (Dahlqvist 1998) contains all articles published in SvD and Uppsala Nya Tidning (UNT) during the same period.

studies already in a format, by necessity, manageable by computers. Many of the web pages are also frequently updated and thus give us a steady access to concurrent use of language in different fields. In this situation, neglecting the usability of Internet as a corpus would be foolish. In our case we used a tool called newsAgent that is a set of Perl scripts designed for gathering news texts, news articles and press releases from the web and routing them by mail according to subscribers defined information needs.

2.4 KTH News Corpus

The project with the KTH News Corpus was initiated in May 2000. We started out collecting news telegrams, articles and press releases from three sources but with the ease of adding new sources we settled for twelve steady news sources (Appendix A). The choice of these news sources was based partly on site and page layout, partly on the wish to somewhat balance the corpus over several types of news topics. Among the chosen news sources are both general news, "daily press", and specialized news sources. The reason for this is the possibility of comparing how the same event is described depending on targeted reader (wording, level of detail, etc). As of February 2001 we have gathered more than 100,000 texts amounting to over 200Mb with an increase of over 10,000 new texts each month. The increase in word forms during March was almost 230,000. The lengths of the texts vary between 5 and 500 sentences with a tendency towards the shorter and an average length of 193 words per text.

The texts are stored in HTML tagged format but only the news heading and the body of the news text is preserved. All other page layout and all navigation tables and banners are removed. Each text is tagged with Meta tags storing the information on time and date of publication, source and source URL. We stored the news in different categories (Appendix A) and thus giving the possibility to study the difference in use of language in, for example, news on cultural respectively sports event. We did this using the news sources own categorization of their news texts (finance, sports, domestic, foreign, etc), instead of a reader based categorization, such as described in Karlgren (2000). The corpus is structured into these categories by the use of catalogue structure, a Hypertext linked index and a search engine driven index thus giving several modes of orientation in the corpus.

For the purpose of evaluating a Swedish stemmer in conjunction with a search engine Carlberger et al. (2001), we manually tagged 100 texts TREC style and constructed questions and answers central to each text. We also tagged each text with named entities (names, places, organisations and date/time) and the five most significant keywords for future evaluation purposes.

Unfortunately copyright issues remain unsolved, we have no permission from the copyright holders except fair use, and so the corpus can only be used for research within our research group. The tool for gathering the corpus, newsAgent, is on the

other hand available for use outside our research group (with the exclusion of mail routing and FTP plug-ins).

2.4.1 Areas of Use

So far the corpus has been used for evaluation and training purposes. Knutsson (2001) has employed the corpus for evaluating error detection rules for Granska (Domeij et al. 1999), a program for checking for grammatical errors in Swedish unrestricted text. The tagged texts have besides, as mentioned above, being used for evaluation of a Swedish stemmer also been utilized in the evaluation of SweSum (Dalianis and Hassel 2001), an automatic text summarizer that among other languages handles Swedish unrestricted HTML tagged or untagged ASCII text and for the training and evaluation of a Named Entity Tagger, SweNam (Dalianis and Åström 2001).

In the near future parts of the corpus will be used and for expanding SweSum with Multi Text Summarization. Other possible areas of use are for producing statistics and lexicons, and for developing a Topic Detection Tracking (for example, see Wayne 2000) system for Swedish news.

2.4.2 The Future of the Corpus

I am now on the verge of rewriting the corpus tools since we now are more fully aware of its potential uses. Among planned improvements are:

- Internal representation in XML
- Automatic tagging of:
  - Parts-of-speech
  - Clause and sentence boundaries
  - Named Entities (persons, locations, etc.)
- Automatic summarization of each text
- Automatic running statistics
  - Average increase per month/week in number of:
    - Texts
    - Sentences
    - Words
    - Word forms
  - Average:
    - Text length (in sentences, words & characters)
    - Sentence length (in words & characters)
    - Word length

  - Total number of:
    * Texts
    * Sentences
    * Words
    * Word forms
- Hopefully a solution to the current copyright issues
- A more balanced choice of channels/sources

This will hopefully result in a tool that in a short period can build a corpus of plain, tagged and summarized versions of the same news text along with appropriate statistics.

2.5 Conclusions

A concluding remark is that a small piece of programming has grown to a complete system which we had great use of in training and evaluation of various natural language tools and that the newsAgent has been a incentive to push our research beyond foreseeable limits. As a part of our online service Nyhetsguiden we have also gained as much as fifty willing beta testers of our language technology tools. We are now on the verge to incorporate our new Named Entity Tagger into newsAgent. We also believe that this proves that it is feasible to acquire a substantial corpus, over a short period of time, from the Internet. One may argue that as long as copyright issues are not solved, the corpus has no legal use outside our research group. While this is true, the corpus has been of great use to us in our research and the corpus tools still remain for public use. The tools have proven to be practically service free run without major problems. Since the same news reports are, potentially, repeated over news sources and time, the resulting corpus will be of much use for research on Information Extraction/Retrieval and Topic Detection Tracking.

Acknowledgements

I would like to thank Hercules Dalianis and Ola Knutsson for comments on early versions of this paper.

2.6 Bibliography

Carlberger, J., H. Dalianis, M. Hassel, and O. Knutsson (2001). Improving Precision in Information Retrieval for Swedish using Stemming. In *Proceedings of NODALIDA'01 - 13th Nordic Conference on Computational Linguistics*, Uppsala, Sweden.

Dahlqvist, B. (1998). The SCARRIE Swedish News Corpus. In Sågwall Hein, A. (editor), *eports from the SCARRIE project*. Uppsala University.

Dalianis, H. (2000). SweSum - A Text Summarizer for Swedish. Technical report, KTH NADA, Sweden.

Dalianis, H. and M. Hassel (2001). Development of a Swedish Corpus for Evaluating Summarizers and other IR-tools. Technical report, KTH NADA, Sweden.

Dalianis, H. and E. Åström (2001). SweNam - A Swedish Named Entity recognizer. Its construction, training and evaluation. Technical report, TRITA-NA-P0113, IPLab-189, KTH NADA.

Domeij, R., O. Knutsson, J. Carlberger, and V. Kann (1999). Granska - An efficient hybrid system for Swedish grammar checking. In *Proceedings of NoDaLiDa'99 - 12th Nordic Conference on Computational Linguistics*.

Hassel, M. (2001). *newsAgent* - A Tool for Automatic News Surveillance and Corpora Building. NUTEK report, http://www.nada.kth.se/~xmartin/papers/Nutek.pdf.

Hofland, K. (2000). A self-expanding corpus based on newspapers on the Web. In *In Proceedings of Second Internation Conference on Language Resources and Evaluation. LREC-2000*, Athens, Greece.

Karlgren, J. (2000). *Assembling a Balanced Corpus from the Internet. In Stylistic Experiments for Information Retrieval.* PhD thesis, Department of Linguistics, Stockholm University, Sweden.

Knutsson, O. (2001). *Automatisk språkgranskning av svensk text.* Licentiate thesis, KTH NADA, Sweden.

Krenn, B. and C. Samuelsson (editors) (1997). *The Linguist's Guide to Statistics - Don't Panic*.

Mani, I., D. House, G. Klein, L. Hirshman, L. Orbst, T. Firmin, M. Chrzanowski, and B. Sundheim (1998). The TIPSTER SUMMAC Text Summarization Evaluation. Technical Report MTR 98W0000138, The Mitre Corporation, McLean, Virginia.

Vorhees, E. and D. Tice (2000). The TREC-8 Question Answering System Track. In *In the proceedings of Second International Conference on Language Resources and Evaluation. LREC-2000*, Athens, Greece.

Wayne, C. (2000). Multilingual Topic Detection and Tracking: Successful Research Enabled by Corpora and Evaluation. In *In the proceedings of Second International Conference on Language Resources and Evaluation. LREC-2000*, Athens, Greece.

Appendix A

News sources and categories used by newsAgent:

Aftonbladet	- Economics, cultural, sports, domestic & foreign news
Amnesty International	- Press releases and news on human rights
BIT.se (Sifo Group)	- Press releases from companies
Dagens Industri	- News on the industrial market
Dagens Nyheter	- Economics, cultural, sports, domestic & foreign news
Homoplaneten (RFSL)	- News concerning rights of the homosexual community
Tidningen Mobil	- News articles on mobile communication
International Data Group	- News articles on computers
Medströms Förlag	- News articles on computers
Senaste Nytt.com	- News flashes (discontinued)
Svenska Dagbladet	- News flashes
Svenska Eko-nyheter	- News flashes
Sveriges Riksdag	- Press releases from the Swedish Parliament

Chapter 3

Paper 2

Improving Precision in Information Retrieval for Swedish using Stemming

Johan Carlberger, Hercules Dalianis,
Martin Hassel, Ola Knutsson
KTH NADA
Royal Institute of Technology
100 44 Stockholm, Sweden
{jfc, hercules, xmartin, knutsson}@nada.kth.se

Abstract

We will in this paper present an evaluation[1] of how much stemming improves precision in information retrieval for Swedish texts. To perform this, we built an information retrieval tool with optional stemming and created a tagged corpus in Swedish. We know that stemming in information retrieval for English, Dutch and Slovenian gives better precision the more inflecting the language is, but precision also depends on query length and document length. Our final results were that stemming improved both precision and recall with 15 respectively 18 percent for Swedish texts having an average length of 181 words.

Keywords: Stemming, Swedish, Information Retrieval, Evaluation

3.1 Introduction

Stemming is a technique to transform different inflections and derivations of the same word to one common "stem". Stemming can mean both prefix and suffix, and in rare cases infix, removal. Stemming can, for example, be used to ensure that the greatest number of relevant matches is included in search results. A word's stem is its most basic form: for example, the stem of a plural noun is the singular; the stem of a past-tense verb is the present tense. The stem is, however, not to be confused with a word lemma, the stem does not have to be an actual word itself. Instead the stem can be said to be the least common denominator for the morphological variants. The motivation for using stemming instead of lemmatization, or indeed tagging of the text, is mainly a question of cost. It is considerably more expensive,

[1]This project is supported by NUTEK (Swedish board for Industrial and Technical Development) FavorIT programme in cooperation with EuroSeek AB.

in terms of time and effort, to develop a well performing lemmatizer than to develop a well performing stemmer. It is also more expensive in terms of computational power and run time to use a lemmatizer than to use a stemmer. The reason for this is that the stemmer can use ad-hoc suffix and prefix stripping rules and exception lists while the lemmatizer must do a complete morphological analysis (based on a actual grammatical rules and a dictionary). Another point of motivation is that a stemmer can deliberately "bring together" semantically related words belonging to different word classes to the same stem, which a lemmatizer cannot.

A problem concerning stemming is the issue of overstemming. If the stemmer removes too much in its quest for the stem the result is that, morphologically or semantically, unrelated words are conjoined under the same stem. For example, if both the words *tiden* ("time") and *tidning* ("newspaper") are stemmed to *tid*, a search for *tidning* also would return documents containing *tiden*. The graveness of this problem depends on both the set of stemming rules and the document collection (more precisely; the index terms used to index the document collection). This is due to the fact that both the set of rules and the set of index terms used influence the amount of index terms conjoined under the same stem.

Here follows a number of algorithms previously used to find the stem of a word, (these are not using static lexicons which also can be used but are not so general).

The so-called Porter stemmer (Porter 1980) for English, which removes around 60 different suffixes, uses rewriting rules in two steps. The Porter stemmer is quite aggressive when creating stems and does overstemming, but still the Porter stemmer performs well in precision/recall evaluations. KSTEM is another stemmer described in Krovetz (1993). KSTEM is not as aggressive as the Porter stemmer, and it does not create as many equivalence classes as the Porter stemmer does. KSTEM is also considered more accurate, but does not produce better results in evaluation experiments. A stemmer for Slovene is described in Popovic and Willett (1992). Since Slovene is morphologically more complicated than English, the Slovene stemmer removes around 5 200 different suffixes. A Porter stemmer for Dutch is described in Kraaij and Pohlmann (1994).

Based on the work in constructing a Swedish tagger (Carlberger and Kann 1999) we developed techniques to find the stems of Swedish words and we have used these techniques in our information retrieval work. Our stemming algorithm for Swedish uses about 150 stemming rules. We use a technique where we, with a small set of suffix rules, in a number of steps modify the original word into an appropriate stem. The stemming is done in (up to) four steps and in each step no more than one rule from a set of rules is applied. This means that 0-4 rules are applied to each word passing through the stemmer. Each rule consists of a lexical pattern to match with the suffix of the word being stemmed and a set of modifiers, or commands, see Figure 1.

The technique is quite general and can easily be adapted to inflectional languages other than Swedish.

*****	Don't remove or replace anything
-	Remove matched if a preceeding vowel is found
+	Remove matched
=	Remove matched if matching the whole word
.	Stop matching (break)
abc	Replace with abc

Figure 1. The set of commands applicable to words being stemmed.

In step 0 genitive-s and active-s are handled; these are basically -s stripping rules. Definite forms of nouns and adjectivies are handled in step 1, as well are preterite tense and past participle.

hals	***.**	Don't remove or replace and stop matching. ("neck")
abel	**-**	Remove matched if a preceding vowel is found.
sköt	**+.skjut**	Remove *sköt*, insert *skjut* ("shoot" or "push") and break

Figure 2. Example of exception rules.

In step 2 mainly plural forms of nouns and adjectives are handled. Noun forms of verbs are handled in step 3. In step 3 there are also some fixes to cover exceptions to the above rules, see Figure 2.

A word's stem does not have to be of the same part of speech as the word; in whatever sense you can talk about part of speech for the stem. The rules are designed so that word classes can be 'merged'. This means that, for example, *cykel* ("bicycle") and *cyklade* ("rode a bicycle") are both stemmed to *cykl*.

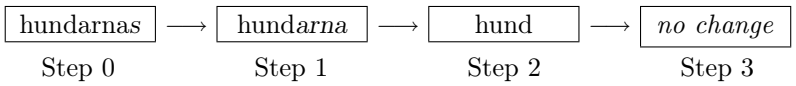

Figure 3. Stemming of the word *hundarnas* ("the dogs"' *genitive form plural*) to *hund* ("dog").

This technique, see Figure 3, works well when the stem is to be used as a internal representation for a set of morphological variants and semantically related words. The stems themselves are, however, often too cryptic to be presented to the user as bearing any information.

3.2 Precision and Recall in Information Retrieval

Regarding information retrieval, there have been experiments using stemming of texts before indexing, or query expansion of the query before retrieving the text collections to investigate the improvement on precision. These experiments have been made for English but also for Slovene and Dutch.

Xu and Croft (1998) describe that stemming at document indexing time is more computational efficient than at query time (query expansion). Query expansion and stemming in information retrieval are regarded as equivalent, but most experiments have been carried out with stemming both on the document collection and on the query, i.e. normalization of both the query and text. (One can also just use query expansion on the query and no stemming on the document collection. Query expansion means that all possible inflections of a word are generated)

Popovic and Willett (1992) found that there is no difference in precision using manual truncation of the query and automatic stemming; both methods gave the same results, at least for Slovene texts.

The first investigations by Harman (1991) indicated that there were no significant improvement in the retrieval using stemming, but in a later study by Krovetz (1993), an improvement of the retrieval (around 40 percent increase in precision) was proven specifically for shorter documents (average 45 words) with short queries (average 7 words). Longer texts (average 581 words) and with short queries (average 9 words) gave only 2 percent increase in precision.

According to Hull (1996), stemming is always beneficial in retrieving documents, around 1-3 percent improvement from no stemming, except on very small document collections.

Popovic and Willett (1992) showed that stemming on a small collection of 400 abstracts in Slovene and queries of average length of 7 words increased precision in information retrieval with 40 percent.

In the above experiments the relation between the number of documents (500 to 180,000 documents) in the document collection and the number of unique questions range between 0.1 percent and 10 percent of the document collection.

3.3 The KTH News Corpus

From the KTH News Corpus, described in detail in Hassel (2001), we selected 54,487 news articles from the period May 25, 2000 to November 4, 2000. From this sub-corpus we randomly selected 100 texts and manually tagged a question and answer pair central to each text; see Figure 4, for an example.

```
<top>
<num> Number: 35
<desc> Description: (Natural Language question)
    Vem är koncernchef på Telenor? (Who is CEO at Telenor?)
</top>

<top>
<num> Number: 35
<answer> Answer: Tormod Hermansen
<file> File: KTH NewsCorpus/Aftonbladet/Ekonomi/8621340_EKO__00.html
<person> Person: Tormod Hermansen
```

```
<location> Location: Norden
<organization> Organization: Telenor
<time> Time: onsdagen
<keywords> Keywords: Telenor; koncernchef; teleföretag; uppköp
</top>
```

Figure 4. Question and Answer tagging scheme.

3.4 Evaluation

Our information retrieval system uses traditional information retrieval techniques extended with stemming techniques and normalization of both the query and text. (The system can also be executed without using the stemming module).

We used a rotating Questioning and Answering evaluation schema to avoid training effects of running the information retrieval system. Each of three users answered 33, 33 and 34 questions respectively with and without stemming functionality. The three users were not allowed to do more than five trials on each question to find the answer and were not allowed to use longer queries than five words. No background knowledge was allowed, which means that only the words used in the natural language question were allowed. Boolean expressions and phrase searches were allowed but rarely used.

After going through all of the 100 questions and finding answers to these, that is 33 questions each, we rotated the work and we became evaluators of the previous persons' answers assessing how many of the found top ten answers were correct and how many were wrong. Of the 100 questions, the test persons found 96 answers, 2 questions did not give any answers at all and 2 other questions gave unreadable files. Each of the asked queries had an average length of 2.7 words. The texts containing the answer had an average length of 181 words. We found a 15 percent increase on precision on the first 10 hits for stemming compared to no stemming (see Table 1). We also compared with weighting the first hits higher than the last ones and we found no significant difference: 14 percent better with stemming and weighting. (We gave the first hit a weighting factor of 10 and the second hit a weighting factor of 9, decreasing the weighting factor until the last tenth hit giving it 1 and then we normalized everything to 1).

Precision/Recall at 10 first "hits"	Word-form	Stemming	Weighted Wordform	Weighted Stemming
Number of questions	96	96	96	96
Average precision	0.255	0.294	0.312	0.353
Increase of precision		15.2%		13.1%
Average relative recall	0.665	0.784		
Increase of relative recall		18.0%		

Table 1. No stemming versus stemming.

Regarding the recall, we calculated the relative recall. Maximum number of recalled texts per question is 21 (=10+10+1). This is calculated using the found unique or disjunctive texts when retrieving using both no stemming and stemming and also adding the tagged correct answer. We calculated the increase in recall taking the difference of the average relative recall, and we found an improvement of 18 percent on relative recall using stemming.

3.5 Conclusions

Stemming (and/or manual truncation) can give better precision (4-40 percent) in information retrieval for short queries (7-9 words) on short documents (500 words) than no stemming at all for languages as English, Dutch and Slovenian. Our experiments show that stemming for Swedish can give at least 15 percent increase in precision and 18 percent increase on relative recall depending on the set of rules and the document collection. We are convinced that the cost in creating a stemmer is proportional to the gain when using the stemmer. This indicates that using stemming on morphologically complicated languages will give great gain in precision.

3.6 Acknowledgments

We would like to thank the search engine team and specifically Jesper Ekhall at Euroseek AB for their support with the integration of our stemming algorithms in their search engine and for allowing us to use their search engine in our experiments.

3.7 Bibliography

Carlberger, J. and V. Kann (1999). Implementing an efficient part-of-speech tagger. In *Software Practice and Experience*.

Harman, D. (1991). How effective is suffixing? *Journal of the American Society for Information Science*, 42(1):1–15.

Hassel, M. (2001). Internet as Corpus - Automatic Construction of a Swedish News Corpus. In *Proceedings of NODALIDA'01 - 13th Nordic Conference on Computational Linguistics*, Uppsala, Sweden.

Hull, D. (1996). Stemming Algorithms - A Case Study for Detailed Evaluation. *Journal of the American Society for Information Science*, 47(1):70–84.

Kraaij, W. and R. Pohlmann (1994). Porter's stemming algorithm for Dutch. In Noordman, L. and W. de Vroomen (editors), *Informatie wetenschap 1994: Wetenschappelijke bijdragen aan de derde STINFON Conferentie*, pp. 167–180, Tilburg, The Netherlands.

Krovetz, R. (1993). Viewing Morphology as an Inference Process. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM*, pp. 191–202, Pittsburgh.

Popovic, M. and P. Willett (1992). The effectiveness of stemming for natural-language access to Slovene textual data. *Journal of the American Society for Information Science*, 43(5):384–390.

Porter, M. (1980). An algorithm for suffix stripping. *Program*, 14(3):130.

Xu, J. and W. Croft (1998). Corpus-based Stemming using Co-occurrence of Word Variants. *ACM Transactions on Information Systems*, 16(1):61–81.

Chapter 4

Paper 3

Development of a Swedish Corpus for Evaluating Summarizers and other IR-tools

Hercules Dalianis and Martin Hassel
KTH NADA
Royal Institute of Technology
100 44 Stockholm, Sweden
email: hercules, xmartin@nada.kth.se

Abstract

We are presenting the construction of a Swedish corpus aimed at research[1] on Information Retrieval, Information Extraction, Named Entity Recognition and Multi Text Summarization, we will also present the results on evaluating our Swedish text summarizer SweSum with this corpus. The corpus has been constructed by using Internet agents downloading Swedish newspaper text from various sources. A small part of this corpus has then been manually annotated. To evaluate our text summarizer SweSum we let ten students execute our text summarizer with increasing compression rate on the 100 manually annotated texts to find answers to questions. The results showed that at 40 percent summarization/compression rate the correct answer rate was 84 percent.

Keywords: Corpus, Evaluation, Text Summarization, Swedish

4.1 Introduction

Two years ago we built a text summarizer called SweSum[2] (Dalianis 2000) for Swedish text. We wanted to evaluate SweSum but there were no annotated Swedish corpus available to evaluate text summarizers or information retrieval tools processing Swedish as it is for the English speaking community, mainly through the TREC (Vorhees and Tice 2000), MUC and TIPSTER-SUMMAC evaluation conferences (Mani et al. 1998, Krenn and Samuelsson 1997).

The only annotated corpora so far for Swedish is the Stockholm-Umeå SUC (1 million words, manually morpho-syntactically annotated) balanced corpus for evaluation of taggers (Ejerhed et al. 1992) and the Swedish Parole corpus aimed

[1]This project is supported by NUTEK (Swedish board for Industrial and Technical Development) FavorIT programme in cooperation with Euroseek AB.

[2]SweSum is available online for testing at http://swesum.nada.kth.se and is also available for Norwegian, Danish, English, Spanish, French, German and Farsi.

at language studies (Språkdata 2000). The text material in the Parole corpus is morpho-syntactically tagged with a statistical tagger. The corpus is balanced, contains approximately 18.5 million words and is available from Språkdata, which is affiliated with Göteborgs Universitet.

One interesting approach to create an evaluation corpus for Swedish is the technique described by Marcu (1999). This technique requires a text and its abstract, from these two inparameters one can create an extract automatically which can be used to assess a text summarizer, but we had no Swedish texts with abstracts available.

Lacking the appropriate tools we managed to make a subjective evaluation of SweSum using the techniques described in Firmin and Chrzanowski (1999). They write that one can make qualitative, subjective, intrinsic evaluations of the text by investigating if the text is perceived as well formed in terms of coherence and content. Therefore we let a number of students within the framework of 2D1418 Språkteknologi (Human Language Technology), a 4-credit course at NADA/KTH, Stockholm, in the fall 1999, automatically summarize an identical set of ten texts each of news articles and movie reviews using our text summarizer SweSum. The purpose was to see how much a text could be summarized without loosing coherence or important information. We found that the coherence of the text was intact at 30 percent compression rate and that the information content was intact at 25 percent compression rate, see Dalianis (2000) (compression rate is defined as the number of words in the summary text divided by number of words in the source text). But to make an objective evaluation we needed an annotated corpus or at least a partly annotated corpus.

The only way to make this possible was to construct a Swedish annotated corpus ourselves, the other reason was that we also needed an annotated corpus to evaluate our Swedish stemming algorithm; see Carlberger et al. (2001). This was two of the reasons to create a Swedish corpus for evaluation of IR-tools.

4.2 Constructing the Corpus

Traditionally it has been hard work constructing a corpus of news text. In Sweden there are no newspapers that on a yearly basis offer their paper in digital form, as some foreign newspapers do. This means that obtaining news texts has to be done on demand. Many Swedish newspapers are, when inquired, unwilling to release texts from their archives for research purposes, and even when they do, it is often the question of a small amount of news texts with an age of several years. This may potentially lead to the exclusion of contemporary words and giving unusually high, or low, occurrence frequencies to words related to phenomena limited to a certain period of time.

In the past, the solution would be to collect newspapers in their paper form and type or scan them (using a Optical Character Recognition program) in order to convert them to a format manageable by computers.

The World Wide Web is, on the other hand, today a large collection of texts written in different languages and thus giving an abundant resource for language studies already in a format, by necessity, manageable by computers. Many of the web pages are also frequently updated and so give a steady access to concurrent use of language in different fields. In this situation, neglecting the usability of Internet as a corpus would be foolish. In our case we used a tool called newsAgent that is a set of Perl programs designed for gathering news articles and press releases from the web and routing them by mail according to subscribers defined information needs.

4.3 Downloading and Storing

The project with the KTH News Corpus was initiated in May 2000. We started out automatically collecting news telegrams, articles and press releases in Swedish from three sources but with the ease of adding new sources we soon settled for twelve steady news sources (Appendix A).

The choice of these news sources was based partly on site and page layout, partly on the wish to somewhat balance the corpus over several types of news topics. Among the chosen news sources are both general news, "daily press", and specialized news sources. The reason for this is the possibility of comparing how the same event is described depending on targeted reader (wording, level of detail, etc).

As of February 2001 we have gathered more than 100.000 texts amounting to over 200Mb with an increase of over 10.000 new texts each month. The increase in word forms during March was almost 230.000. The lengths of the texts vary between 5 and 500 lines with a tendency towards the shorter and an average length of 193 words per text.

The texts are stored in HTML tagged format but only the news heading and the body of the news text is preserved. All other page layout and all navigation tables and banners are removed. Each text is tagged with Meta tags storing the information on time and date of publication, source and source URL. Using the news sources own categorization of their news texts, instead of a reader based categorization (Karlgren 2000), we have stored the news in different categories (Appendix A). This gives the possibility to study the difference in use of language in, for example, news on cultural respectively sports events. The corpus is structured into these categories by the use of catalogue structure, a HyperText linked index and a search engine driven index thus giving several modes of orientation in the corpus.

Since the purpose of the corpus is research on Information Retrieval, Information Extraction, Named Entity Recognition and Multi Text Summarization the system does not, contrary to Hofland (2000), remove duplicated concordance lines.

4.4 Annotation

From the downloaded corpus we selected 54487 news articles from the period May 25, 2000 to November 4, 2000 and from these text we decided to manually annotate 100 news articles.

Three different persons constructed the Question and Answering (Q&A) schema, in total 100 questions and answers, (33,33 and 34 Q&A respectively each), by randomly choosing among the 54 487 news articles from KTH News corpus. Finding a suitable text, constructing a question from the text, finding the answer in the text, annotating the found text with: Filename, Person, Location, Organization, Time and five keywords. The 100 texts had an average length of 181 words each.

The reason to have the above tag-set was that the corpus is used and will be used to many tasks, namely, evaluation of an IR tool, (Carlberger et al. 2001), Text Summarization, Multi Text Summarization, Name Entity (NE) recognition and key word extraction. We constructed a Question and Answering annotation schema see Figure 1, following the annotation standard in Mani et al. (1998).

```
<top>
<num> Number: 35
<desc> Description: (Natural Language question)
    Vem är koncernchef på Telenor? (Who is CEO at Telenor?)
</top>

<top>
<num> Number: 35
<answer> Answer: Tormod Hermansen
<file> File: KTH NewsCorpus/Aftonbladet/Ekonomi/8621340_EKO__00.html
<person> Person: Tormod Hermansen
<location> Location: Norden
<organization> Organization: Telenor
<time> Time: onsdagen
<keywords> Keywords: Telenor; koncernchef; teleföretag; uppköp
</top>
```

Figure 1. Question and Answer tagging scheme.

4.5 Evaluation

Objective methods to evaluate text summarizers are described in Mani et al. (1998), one of these methods is to compare the produced summary (mainly extracts) with manually made extracts from the text to judge the overlap and consequently assess the quality of the summary. One other objective method to evaluate text summarizers is taken from the information retrieval area where a Question and Answering schema is used to reveal if the produced summary is the “right one”.

A text summarizer summarizes a text and one human assess if the summary contains the answer of a given question. If the answer is in the summarized text then the summary is considered good.

We let ten students within the framework of 2D1418 Språkteknologi (Human Language Technology), a 4-credit course at NADA/KTH, Stockholm, in the fall 2000, automatically summarize a set of ten news articles each using the text summarizer SweSum at increasing compression rates 20, 30 and 40 percent. If the 20, 30 and 40 percent summaries failed then the users could select their own key words to direct the summarizer at 20 percent compression rate to find the answers to the predefined questions. We then compared the given answers with the correct ones. The results are listed in Table 1 below.

Summary/ Compresssion rate	**20%**	**30%**	**40%**	**Keywords at 20%**	**Correct answers**
Number of texts	97	97	97	97	
Given and correct answers	50	16	15	4	85
Percent accumulated correct answers	52%	68%	84%	88%	

Table 1. Evaluation of the text summarizer SweSum.

From the evaluation at 20 percent compression rate we can conclude that we obtained 52 percent correct answers and at 40 percent compression rate we obtained totally 84 percent correct answers, only 12 summaries did not give any answer at all (some of the them did not become summarized due to technical problems).

We noted during the annotation phase that if we had constructed questions with a yes answer or a one-word answer instead of a long ambiguous complicated answer then we could had automated the evaluation process since the computer automatically could check if the manually given answer is correct or not.

4.6 Conclusions

We have constructed the first Swedish corpus for evaluating text summarizers and information retrieval tools. We found that our text summarizer SweSum at 40 percent compression rate gave 84 percent correct answers. From this evaluation we can conclude that our summarizer for Swedish is state-of-the-art compared to other summarizers for English (Mani et al. 1998). Comparing our current objective evaluation results we can also validate that our previous subjective evaluation results (Dalianis 2000) were correct, saying that 30 percent compression rate gave good summaries.

There is no perfect summarization every person has his preference when creating an abstract from a text. Except for the evaluation of the text summarizer SweSum, the corpus has been used for tree other evaluation purposes: First, for evaluating our Swedish stemming algorithm; see Carlberger et al. (2001) (we obtained 15

percent improvement in precision and 18 percent improvement on relative recall using stemming for Swedish), second for evaluating our Swedish Named Entity recognizer - SweNam (Dalianis and Åström 2001) (we obtained 92 percent precision and 46 percent recall) and third for evaluating error detection rules for Granska, a program for checking for grammatical errors in Swedish unrestricted text, see Knutsson (2001).

Unfortunately copyright issues remain unsolved so the corpus can only be used for research within our research group. The tool for gathering the corpus, newsAgent, is on the other hand available for use outside our research group (with the exclusion of mail routing and FTP plug-ins).

4.7 Bibliography

Carlberger, J., H. Dalianis, M. Hassel, and O. Knutsson (2001). Improving Precision in Information Retrieval for Swedish using Stemming. In *Proceedings of NODALIDA'01 - 13th Nordic Conference on Computational Linguistics*, Uppsala, Sweden.

Dalianis, H. (2000). SweSum - A Text Summarizer for Swedish. Technical report, KTH NADA, Sweden.

Dalianis, H. and E. Åström (2001). SweNam - A Swedish Named Entity recognizer. Its construction, training and evaluation. Technical report, TRITA-NA-P0113, IPLab-189, KTH NADA.

Ejerhed, E., G. Källgren, O. Wennstedt, and M. Åström (1992). *SUC - The Stockholm-Umeå Corpus*, version 1.0 (suc 1.0). CD-ROM produced by the Dept of Linguistics, University of Stockholm and the Dept of Linguistics, University of Umeå. ISBN 91-7191-348-3.

Firmin, T. and M. J. Chrzanowski (1999). An Evaluation of Automatic Text Summarization Systems. In Mani, I. and M. T. Maybury (editors), *Advances in Automatic Text Summarization*, pp. 325–336. MIT Press.

Hofland, K. (2000). A self-expanding corpus based on newspapers on the Web. In *In Proceedings of Second Internation Conference on Language Resources and Evaluation. LREC-2000*, Athens, Greece.

Karlgren, J. (2000). *Assembling a Balanced Corpus from the Internet. In Stylistic Experiments for Information Retrieval.* PhD thesis, Department of Linguistics, Stockholm University, Sweden.

Knutsson, O. (2001). *Automatisk språkgranskning av svensk text.* Licentiate thesis, KTH NADA, Sweden.

Krenn, B. and C. Samuelsson (editors) (1997). *The Linguist's Guide to Statistics - Don't Panic.*

Mani, I., D. House, G. Klein, L. Hirshman, L. Orbst, T. Firmin, M. Chrzanowski, and B. Sundheim (1998). The TIPSTER SUMMAC Text Summarization Evaluation. Technical Report MTR 98W0000138, The Mitre Corporation, McLean, Virginia.

Marcu, D. (1999). The Automatic Construction of Large-Scale Corpora for Summarization Research. In Hearst, M., G. F., and R. Tong (editors), *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 137–144, University of California, Berkely.

Språkdata (2000). The Swedish PAROLE Lexicon. http://spraakdata.gu.se/parole/lexikon/swedish.parole.lexikon.html.

Vorhees, E. and D. Tice (2000). The TREC-8 Question Answering System Track. In *In the proceedings of Second International Conference on Language Resources and Evaluation. LREC-2000*, Athens, Greece.

Chapter 5

Paper 4

Exploitation of Named Entities in Automatic Text Summarization for Swedish

Martin Hassel
KTH NADA
Royal Institute of Technology
100 44 Stockholm, Sweden
xmartin@nada.kth.se

Abstract

Named Entities are often seen as important cues to the topic of a text. They are among the most information dense tokens of the text and largely define the domain of the text. Therefore, Named Entity Recognition should greatly enhance the identification of important text segments when used by an (extraction based) automatic text summarizer. We have compared Gold Standard summaries produced by majority votes over a number of manually created extracts with extracts created with our extraction based summarization system, SweSum. Furthermore we have taken an in-depth look at how over-weighting of Named Entities affects the resulting summary and come to the conclusion that weighting of Named Entities should be carefully considered when used in a naïve fashion.

Keywords: Named Entities, Gold Standard Corpus, Evaluation, Text Summarization, Swedish

5.1 Background

The technique of automatic text summarization has been developed for many years (Luhn 1958, Edmundson 1969, Salton 1988). One way to do text summarization is by text extraction, which means to extract pieces of an original text on a statistical basis or with heuristic methods and put them together to a new shorter text with as much information as possible preserved (Mani and Maybury 1999).

One important task in text extraction is topic identification. There are many methods to perform topic identification Hovy and Lin (1997). One is word counting

at concept level that is more advanced than just simple word counting; another is identification of cue phrases to find the topic.

To improve our automatic text summarizer and to a larger extent capture the topic of the text we tried to use Named Entity Recognition. Named Entity recognition is the task of finding and classifying proper nouns in running text. Proper nouns, such as names of persons and places, are often central in news reports. Therefore we have integrated a Named Entity tagger with our existing summarizer, SweSum, in order to study its effect on the resulting summaries.

5.2 Introducing SweSum

The domain of SweSum (Dalianis 2000) is Swedish newspaper text. SweSum utilizes several different topic identification schemes. For example the bold tag is often used to emphasize contents of the text. Headings are also given a higher weight. In news paper text the most relevant information is always presented at the top. In some cases the articles are even written to be cuttable from from the bottom. Because of this we use Position Score Hovy and Lin (1997); sentences in the beginning of the text are given higher scores than later ones.

Sentences that contain keywords are scored high. A keyword is an open class word with a high Term Frequency (tf). Sentences containing numerical data are also considered carrying important information. All the above parameters are put in a naïve combination function with modifiable weights to obtain the total score of each sentence.

5.3 Working Hypothesis

Named Entities are often seen as important cues to the topic of a text. They are among the most information dense tokens of the text and largely define the domain of the text. Therefore, Named Entity Recognition should greatly enhance the identification of important text segments when used by an (extraction based) automatic text summarizer.

5.4 Enter SweNam

For Named Entity recognition and classifying SweNam (Dalianis and Åström 2001) is used. SweNam acts as a preprocessor for SweSum and tags all found Named Entities with one of the four possible categories - names of persons (given name and/or surname), locations (geographical as well as geopolitical), companies (names of companies, brands, products, organizations, etc) and time stamps (dates, weekdays, months, etc). The Named Entities found by SweNam are quite reliable, as it has shown a precision of 92 percent. However, the recall is as low as 46 percent, so far from all Named Entities are considered during the summarization phase.

All found entities are given an equal weight and entered, together with the parameters described above, into the combination function in weighting module in the summarizer, SweSum.

5.5 Creating a Gold Standard

For the evaluation we collected two sets of texts, each set consisting of 10 news texts. The first set (Group 1) consisted of ten news articles randomly chosen from Svenska Dagbladets web edition (http://www.svd.se/) over a couple of days. These where summarized using SweSum both with and without the use of Named Entity Recognition.

In order to evaluate and compare the two subsets of generated extracts from Group 1 we devised a system to collect manual extracts for the news articles from Group 1. Human test subjects where presented the news articles one at a time in random order in the form of one sentence per line. In front of each line was a checkbox with which the informant could select that particular sentence for extraction. The informant could then choose to generate an extract based on the selected sentences. This extract was then presented to the informant who had to approve the extract before it was entered into a database. Submitted extracts were allowed to vary between 5% and 60% of the original text length.

The result was that 11 informants submitted a total of 96 extracts for the ten texts of Group 1. Each news text received between 8 and 11 manual extracts and the mean length of submitted extracts was 37%. These manual extract constituted the foundation for the KTH eXtract Corpus.

There was, as expected, not very much agreement between the informants on which sentences to select for the extract. The level of agreement among the informants was calculated with a simple precision function. This is done per text and then a mean was calculated over all ten texts.

$$AgreementLevel = \frac{Vc}{Ns \times Nx} \times 100 \tag{5.1}$$

In the function above Vc is the number of votes that are represented in the generated extract, Ns is the number of sentences represented in the same extract and Nx is the number of man-made extracts made for the original text the votes and sentences account for. This means that when all informants choose not only the same number of sentences but also exactly the same set of sentences the function will result in a precision, or agreement, of 100%.

We where prepared for a low agreement among the human extractors as to which sentences are good summary sentences as previous studies have shown this (for an overview see Mani (2001). When taking all selected extraction units into account for each text there was only a mean agreement of 39.6%. This is however not so bad as it can seem at first glance. When generating a "gold standard" extract by presenting the most selected sentences up to a summary length of the mean length

of all man-made extracts for a given text the precision, or the agreement level, rose to 68.9%. Very few of the sentences chosen for the gold standard where selected by as few as one third or less of the informants. Of course, even fewer sentences where selected by all informants. In fact, not even all informants could agree upon extracting the title or not when one was present.

5.6 Evaluation

The extract summaries generated with SweSum where then manually compared on sentence level with the gold standard summaries generated by majority vote. We found that with Named Entity Recognition the summaries generated by SweSum and the gold standard only had 33.9% of their sentences in common (table 1). On the other hand, without Named Entity Recognition the summaries generated with SweSum shared as many as 57.2% of the sentences with the gold standard.

	With NER	**Without NER**
Shared sentences	33.9%	57.2%

Table 1. Gold standard compared to SweSum generated extracts.

Of course this does not say much about how good the summaries were, only how well the different runs with SweSum corresponded to what our informants wanted to see in the summaries. That is, the figures represent how well SweSum mimics human selection with and without the use of Named Entity Recognition.

5.6.1 Reference Errors

The difference in readability and coherence of the two types of SweSum generated summaries was quite interesting. When scrutinizing the extracts we decided to look at a typical problem with extraction-based summarization - reference errors due to removed antecedents. This error was divided into two severity levels, anaphors that refer to the wrong antecedent and anaphors that does not have any antecedent at all to point to.

In the subset of extracts generated using Named Entity Recognition there where a total of three reference errors (pronouns etc.) and 13 cases of completely lost context over the ten extract summaries (table 2). In the summaries generated not using Named Entity Recognition there were six reference errors and only two cases of completely lost context over the ten summaries.

	With NER	**Without NER**
Reference errors	3 errors	6 errors
Complete loss of context	13 cases	2 cases

Table 2. Referential errors in Group 1 extracts.

The extracts generated using Named Entity Recognition clearly showed a lot more coherency problems and loss of context.

To verify the above observations and to see how much NE affected the summarization result we collected a second set of texts (Group 2) and generated new summaries. The second set consisted of 10 news texts randomly chosen from KTH News Corpus (Hassel 2001). These were summarized with a high, low and no weight on Named Entities in SweSum. As shown in table 3 the observations for the Group 1 summaries were very much verified in Group 2. In this new set of extract summaries those generated using Named Entity Recognition showcased a total of 10 respectively 12 reference errors while the set of summaries generated not using Named Entity Recognition only contained 4 errors over the ten summaries.

NE weighting	**High weight**	**Low weight**	**No weight**
Reference errors	3 errors	3 errors	2 errors
Complete loss of context	7 cases	9 cases	2 cases

Table 3. Referential errors in Group 2 extracts.

Surprisingly enough the gold standard showed no reference error at all.

5.6.2 Loss of Background Information

Our conclusion is that weighting of Named Entities tend to prioritize singular sentences high in information centered on the categories used. The result is that it tends to prioritize elaborative sentences over introductory and thus sometimes is responsible for serious losses of sentences giving background information. Our guess is that elaborative sentences have more Named Entities per sentence than introductory due to the fact that introductory sentences focus on something newly introduced in the text. However we have no statistics to substantiate this claim. This often lessens the coherency of the summary (example 1). One solution to this would of course be to extract the paragraph with the highest-ranking sentences (Fuentes and Rodríguez 2002); another is to let sentence position highly outweigh Named Entities (Nobata et al. 2002).

> - Hennes tillstånd är livshotande, säger jourhavande åklagare **Åke Hansson**.
> Lisa **Eriksson** var knapphändig i sina uppgifter på tisdagen.
> Sjukvården i *Sundsvall* räckte inte till för att rädda flickan.
> Enligt läkare i *Uppsala* var hennes tillstånd i går fortfarande livshotande.
> 2001 anmäldes nära 7 000 fall av barnmisshandel i *Sverige*. På **Astrid** Lindgrens barnsjukhus i *Solna* upptäcks i dag ungefär ett spädbarn i månaden som är offer för den form av barnmisshandel som kallas Shaken baby-syndrome.
> **Petter Ovander**
> **Example 1.** Summarized with Named Entities

One way of bouting the problem of loss of background information is of course to raise the size of the extraction unit. If we raise the extraction unit to encompass for example paragraphs instead of sentences the system would identify and extract

only the most important paragraph(s) as in Fuentes and Rodríguez (2002). This would lessen the risk of loosing background information at least on paragraph level as well as almost completely eliminate the risk of loss of antecedent for extracted pronouns. On longer texts loss of background information and coherency problem can still of course arise on chapter or text level.

Another way to try to benefit from the use of Named Entity Recognition in Automatic Text Summarization without risking the loss of background information is of course to use a very low weight for NE relative to other weights used (for example keyword frequency and sentence position) and hope that it fine-tunes the summary rather than letting it have a large negative impact on it. This is supported by experiments by Nobata et al. (2002) where they trained an automatic summarization system on English {extract,text} tuples and noted that the weight given by the training system to the Named Entity Recognition module was significantly lower than for the other modules.

5.6.3 Condensed Redundancy

When no weighting of Named Entities is carried out clusters of interrelated sentences tend to get extracted because of the large amount of common words. This gives high cohesion throughout the summary but sometimes leads problems with condensed redundancy. For example:

> 6 veckors baby svårt misshandlad
> Pappan misstänkt för misshandeln
> En sex veckor gammal bebis kom sent i lördags kväll svårt misshandlad in på akuten i Sundsvall. Flickan har mycket svåra skall- och lungskador. - Hennes tillstånd är livshotande, säger jourhavande åklagare Åke Hansson. Barnets pappa har anhållits som misstänkt för misshandeln på den sex veckor gamla flickan.
> Sex veckor gammal
> Flickan - som enligt uppgift till Aftonbladet är sex veckor gammal - kom in till akuten Sundsvalls sjukhus vid 22-tiden i lördags kväll. Hennes skador var livshotande.
> Petter Ovander
> **Example 2.** Summarized without Named Entities

We can clearly see how redundancy in the original text “sex veckor gammal” (“six weeks old”) is not only preserved but rather emphasized in the summary. This is because the term frequency (tf), the frequency of the keywords, heavy influences the selection.

5.6.4 Over-explicitness

When summarizing with weighting of Named Entities the resulting summaries sometimes seem very repetitive (Example 3) but are in fact generally less redundant than the ones created without weighting of Named Entities.

Pojkarna skrek att de ville ha pengar och beordrade **Pierre** att gå till kassan.
Pierre minns inte i detalj vad som sedan hände, mer än att det första yxhugget träffade I ryggen.
Liggande på marken fick **Pierre** ta emot tre yxhugg i huvudet.
Pierre lyckades slita yxan ur händerna på 28-åringen.
Pierre hade svårt att läsa och fick börja om från början igen.
I dag har **Pierre** lämnat händelserna 1990 bakom sig.
Psykiskt har **Pierre** klarat sig bra.
Example 3. Summarized with Named Entities

In this case the male name Pierre is repeated over and over again. With the proper noun repeated in every sentence the text appears overly explicit and staccato like. There is no natural flow and the text feels strained and affected. A solution to this would be to generate pronouns in short sequences and keeping only for example every third occurrence of a name in an unbroken name-dropping sequence.

5.7 Conclusions

Named Entities, as well as high frequent keywords, clearly carry clues to the topic of a text. Named Entities tend to identify informative extraction segments without emphasizing redundancy by preferring similar segments. A major problem we identified in our experiments is that the Named Entity module tends to prioritize elaborative sentences over introductory and thus sometimes is responsible for serious losses of sentences giving background information. Because of this one of the main difficulties using Named Entities in the weighting scheme would be, as with any lexical or discourse parameter, how to weight it relatively the other parameters. When centering the summary on a specific Named Entity there also arises the need for pronoun generation to avoid staccato like summaries due to over-explicitness.

When producing informative summaries for immediate consumption, for example in a news surveillance or business intelligence system, the background may often be more or less well known. In this case the most important parts of the text is what is new and which participants play a role in the scenario. Here Named Entity Recognition can be helpful in highlighting the different participants and their respective role in the text. Other suggested and applied methods of solving the coherence problem are, as we have seen, to raise the extraction unit to the level of paragraphs or to use a very low, almost insignificant, weight on Named Entities.

5.8 Demonstrators

The two different versions of SweSum as well as the small corpus of Swedish news texts and man-made extracts are available on the web if anyone desires to reproduce or do further experiments. The corpus comes with the gold standard extracts generated by majority vote as well as three computer generated baselines. These are available on the following addresses:

SweSum (standard version):
http://swesum.nada.kth.se/index-eng.html
SweSum (NE version):
http://www.nada.kth.se/~xmartin/swesum_lab/index-eng.html
KTH extract corpus:
http://www.nada.kth.se/iplab/hlt/kthxc/showsumstats.php

SweNam is also available online for testing purposes at:
http://www.nada.kth.se/~xmartin/swene/index-eng.html

5.9 Bibliography

Dalianis, H. (2000). SweSum - A Text Summarizer for Swedish. Technical report, KTH NADA, Sweden.

Dalianis, H. and E. Åström (2001). SweNam - A Swedish Named Entity recognizer. Its construction, training and evaluation. Technical report, TRITA-NA-P0113, IPLab-189, KTH NADA.

Edmundson, H. P. (1969). New Methods in Automatic Extracting. *Journal of the Association for Computing Machinery*, 16(2):264–285.

Fuentes, M. and H. Rodríguez (2002). Using cohesive properties of text for Automatic Summarization. In *JOTRI2002 - Workshop on Processing and Information Retrieval*.

Hassel, M. (2001). Internet as Corpus - Automatic Construction of a Swedish News Corpus. In *Proceedings of NODALIDA'01 - 13th Nordic Conference on Computational Linguistics*, Uppsala, Sweden.

Hovy, E. and C.-Y. Lin (1997). Automated Text Summarization in SUMMARIST. In *Proceedings of the ACL97/EACL97 Workshop on Intelligent Scalable Text Summarization*, Madrid, Spain.

Luhn, H. P. (1958). The Automatic Creation of Literature Abstracts. *IBM Journal of Research Development*, 2(2):159–165.

Mani, I. (2001). Summarization Evaluation: An Overview. In *Proceedings of the Second NTCIR Workshop on Research in Chinese & Japanese Text Retrieval and Text Summarization*.

Mani, I. and M. T. Maybury (editors) (1999). *Advances in Automatic Text Summarization*. MIT Press, Cambridge, MA.

Nobata, C., S. Sekine, H. Isahara, and R. Grishman (2002). Summarization System Integrated with Named Entity Tagging and IE pattern Discovery. In *Proceedings of Third International Conference on Language Resources and Evaluation (LREC 2002)*, Las Palmas, Canary Islands, Spain.

Salton, G. (1988). *Automatic Text Processing*. Addison-Wesley Publishing Company.

Index