Portable Text Summarization

Martin Hassel and Hercules Dalianis
Department of Computer and Systems Sciences (DSV), Stockholm University, Sweden

ABSTRACT

Today, with digitally stored information available in abundance, even for many less commonly spoken languages, this information must by some means be filtered and extracted in order to avoid drowning in it. Automatic summarization is one such technique, where a computer summarizes a longer text into a shorter non-redundant form. The development of advanced summarization systems also for smaller languages may unfortunately prove too costly. Nevertheless, there will still be a need for summarization tools for these languages in order to curb the immense flow of digital information. This chapter sets the focus on automatic summarization of text using as few direct human resources as possible, resulting in what can be perceived as an intermediary system. Furthermore, it presents the notion of taking a holistic view of the generation of summaries.

INTRODUCTION

Text summarization is the process of creating a summary of one or more texts. This summary may serve several purposes. One might, for example, want to get an overview of a document set in order to choose what documents one needs to read in full. Another plausible scenario would be getting the gist of a constant news flow, without having to wade through inherently redundant articles run by several news agencies, in order to find what might differ in reports from different parties. With digitally stored information available in abundance and in a myriad of forms, even for many less commonly spoken languages, it has now become near impossible to manually search, sift and choose which information one should incorporate. Instead this information must by some means be filtered and extracted in order to avoid drowning in it. Automatic summarization is one such technique.

There are a number of techniques for text summarization in use, with the most obvious being the snippets or extracts one can see in the hit lists of search engines in conjunction to the search results. These snippets are carefully created based on the context of the query words provided by the user and give a very condensed overview of the retrieved documents.

The application areas for automatic text summarization are extensive. As the amount of information on the Internet grows abundantly it is difficult to sift through and select relevant information. Information is published simultaneously through many media channels in different versions. The same news story could, for instance, be published in a paper newspaper, web newspaper, as a SMS news flash, mobile radio newscast and a spoken newspaper for the visually impaired. Also, these may today be accessed by a myriad of display devices, sporting a wide range of presentation capacity. Customization 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, and hence save human translators work since they need not translate every document manually. 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.

Some everyday tools for text summarization can be found in Microsoft Word *AutoSummarize* and in the Apple Safari web browser. With the Safari text summarizer you can mark a piece of text and then select *Services* and *Summarize* and get the text summarized at the size you wish either at sentence or paragraph level. Regarding Microsoft Word you can find AutoSummarize in Word under *Tools* and *AutoSummarize*. You can select to see extracts in the context of the original text or obtain a complete new summarized text.

The title of this chapter sets the focus on summarization of text, automatically carried out by a computer program using methods more or less directly transferable from one language to another.

This accomplished by using as few human resources as possible. The resources that are used should to as high extent as possible be already existing, not specifically aimed at summarization and, preferably, created as part of natural literary processes. Moreover, the summarization system should be able to be easily assembled using only a small set of basic language processing tools, again, not specifically aimed at summarization. The summarization system should thus be near language independent as to be quickly ported between different natural languages. The motivation for this is as simple as intuitive. Apart from the major languages of the world, there simply are a lot of languages for which large bodies of data aimed at language technology research, let alone research in automatic text summarization, are lacking. There might also not be resources available to develop such bodies of data, since it is usually time-consuming and hence expensive. Nevertheless, there will still be a need for sufficiently efficacious automatic text summarization for these languages, acting as intermediate states, in order to subdue this constantly increasing amount of electronically produced text.

This chapter is organized in the following way. First comes the section *Background* that gives the reader an overview of the main approaches to automatic text summarization. The second section *Language Dependent Processing* gives an overview of how language specific traits affect summarization tools. The third section, *Language Independent Processing,* is the contrast to language dependant processing and we can here see in what manner a summarization system can be made truly portable. The fourth section *Holistic Summarization* gives the gist and the state-of-the-art of text summarization and finally the fifth section is called *Application of Automatic Text Summarization* were one see the use of all these wonderful tools.

BACKGROUND

Summarization approaches are often divided into two main groups, *text abstraction* and *text extraction*. Text abstraction is in many aspects similar to what humans abstractors do when writing an abstract, even though professional abstractors often utilize surface-level information such as headings, key phrases and position in the text as well as the overall organization of the text into more or less genre specific sections (Liddy 1991, Endres-Niggemeyer et al. 1995, Cremmins 1996). Text abstraction naturally falls under the field of paraphrasing, which is the restating of a passage such that both passages would generally be recognized as lexically and syntactically different while remaining semantically invariable (McCarthy et. al 2009). The parsing and interpretation of text is a venerable 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. Also the generation of text is a vigorous research field with techniques ranging from canned text and template filling to more advanced systems with discourse planners and surface realizers.

Text extraction, on the other hand, simply reuses a subset of the original text, thus preserving the original wording and structure of the source text. Sometimes the extracted fragments are post-edited, for example by deleting subordinate clauses or joining incomplete clauses to form complete clauses (Jing & McKeown 2000, Jing 2000). Most of the research in the field of automatic text summarization has to the nature been that of text extraction. If the aim is a portable system that can – with as little friction as possible – travel between languages, then semantic parsing into a formal description as well as the use of handcrafted semantic resources certainly is out of the question.

A distinction most pertinent to this chapter is that of *language dependent* and *language independent* natural language processing (NLP). A language dependent system would be a system geared at a specific language, or a set of languages. It might perhaps utilize manually built lexical resources such as ontologies, thesauri or other language or domain specific knowledge bases. Other dependencies constraining a system to a specific language may be the employment of advanced tools as, for example, full parsers, semantic role assigners or named entity tagging, or the use of techniques such as template filling. The term “language independent”, on the other hand, usually denotes a NLP system that is easily transferred between different languages or domains. The system is thereby independent of the target language. In this chapter we will mainly investigate two research topics in the context of automatic text summarization. These two desired properties are *resource lean* and *portable*.

LANGUAGE DEPENDENT PROCESSING

A *true* language independent NLP system should be directly transferable to new domains or a completely different natural language. As will be discussed in this chapter, there are many steps to cover on the way to a portable summarization system. No matter to what extent an extraction based summarization system is, or claims to be, language independent it still has to do some more or less language dependent preprocessing before it can be applied to the task in question. At the very least some basic knowledge about natural languages in general must be made available to the system in order to facilitate segmentation of the text into desirable units of extraction.

Preprocessing

Prior to any deeper linguistic treatment of a text the units of that text must be demarcated and possibly classified. The text can initially be viewed as a mere sequence of characters within which we must define these units (Grefenstette & Tapanainen 1994). First after having defined and isolated the units we are interested in we can begin to operate on them. This stage of isolation occurs on many levels; e.g. tokenization divides the character sequence into words, sentence splitting further divides sequences of words into sentences, and so on. These preprocessing steps are often more or less language dependent, and thus deserve a brief discussion.

Tokenization

In order to perform extraction based summarization we must first decide on what granularity our extraction segments will have, i.e. the "size" of the textual units that we copy from the original text and paste into our summary. This could be a paragraph, a sentence, a phrase or even a clause, although the most common probably is extraction performed on sentence level. Often it is necessary to first split the text into words (tokens) in order to correctly identify these boundaries between clauses, phrases or sentences. Sentence splitting as such is often considered as a non-trivial task, considering the irregularities of natural languages. However, at least for many Germanic and Romance languages a small set of regular expression rules, perhaps accompanied by a list of abbreviations commonly including a punctuation mark, usually produces an acceptable result. Using a list of abbreviations of course makes the tokenization inherently language dependent as an abbreviation list usually is handcrafted. There are, though, languages lacking word-boundary markers, such as Chinese and Japanese, which certainly provide a more challenging task (Crystal 1987), but much statistical work has been carried out also for these languages, e.g. Chinese word segmentation (Luk 1994).

Managing Word Forms

In running text the same word unit in many languages can occur in several different morphological variants. These inflected forms are governed by the context, i.e. if the text is presented in singular or plural form, present or past tense etc. In most cases these different lexical forms have similar semantic interpretations and can consequently often be considered as equivalent for the purpose of many Information Management (IM) applications. In order for an IM system to be able to treat these inflected forms as one concept, often referred to as a lexeme, it is common to use a so-called stemming algorithm. Efforts towards statistical language independent stemming have been taken, so this step can possibly be automated in a truly language independent system. A promising such approach, where stem classes are built using co-occurrence statistics, has been proposed by Xu & Croft (1998). The authors have demonstrated an improvement in information retrieval after clustering stem classes for English and Spanish, but for more morphologically complex languages the challenge of language independence still awaits. Another such approach by Bacchin et al. (2002) treats prefixes and suffixes as a community of sub-strings. They attempt to discover these communities by means of searching for the best word splits, which in turn give the best word stems.

Most statistical language models are more or less susceptible to sparse-data issues. In reality this means that the presence of very rare words, or patterns, adds noise to the model since the statistical grounds for modeling a representation of these are too weak. One such phenomenon is compounding. Agglutinative languages, which are languages in which most words are formed by joining morphemes

together, tend to be very productive in creating compound words (Crystal 1987). Theoretically the length of a compound word is unlimited, however longer compounds tend to become unwieldy and are infrequent in actual discourse. For example, the fully valid Swedish compound noun *strålbehandlingsplaneringsdatortomografi* would translate into English, being a mostly analytic language, as "radiation treatment planning computer tomography". This 40-letter word of course makes for a very rare occurrence even in a gigaword corpus. It should thus be perfectly clear that the representation of a document's content would benefit highly from having each constituting lexeme represented separately, thereby consolidating frequencies. Consequently, related to the task of stemming is that of compound splitting. Several statistical approaches to identifying lexeme boundaries in compounds exist, and may be used in different combinations, where one for instance can take advantage of co-occurrence statistics by letting words that occur more often in the same text be the preferred split (Sjöbergh & Kann 2004).

Also, it is common to remove so-called stop words prior to the construction of these document descriptions, leaving only the content bearing words in the text during processing. Often a predefined, handcrafted stop list containing common words is used for removal of words known to be function words, which although far more high frequent than content words are far fewer in number. This does, however, make the creation of document content representations language dependent. A language independent alternative would be to apply term frequency thresholds where terms (words) that have a very high, and possibly also those that have a very low, frequency are removed and thereby reducing noise in the model.

LANGUAGE INDEPENDENT PROCESSING

The approach to language independent summarization presented in this chapter heavily relies on the notion that documents, or rather a source document and a set of proposed summaries, can be compared for similarity. This notion is well established in, for example, Information Retrieval, where user queries act as fuzzy "descriptions" that are matched to a set of documents in order to find the document most similar to that description. When comparing documents for content similarity it is common practice to produce some form of document signatures. These signatures represent the content in some way, often as a vector of features, which are used as basis for such comparison. This comparison can be attempted on several levels, e.g. on lexical, syntactic or semantic grounds.

The Vector Space Model

The Vector Space model is a document similarity model commonly used in Information Retrieval (Salton 1971). In this model the document signatures are represented as feature vectors consisting of the words that occur within the documents, with weights attached to each word denoting its importance for the document. We can, for example, for each term (word) record the number of times it occurs in each document. This gives us what is commonly called a document-term matrix, where the rows represent the documents in the document collection and the columns each represent a specific term existing in any of the documents (a weight can thus be zero).

Using this matrix we can view the feature vectors as projections in a multi-dimensional space where the dimensionality is given by the number of documents and the number of index terms (the vocabulary, if you will). We can then measure the lexical similarity between two documents simply by calculating the in information retrieval commonly used *cosine* angle between these vectors. For our purposes, the two documents being compared for similarity might as well be a document being summarized coupled with a summary of said document, as we shall see further on.

Term Weighting

When constructing document signatures it also common to modify word frequency counts in the hope of promoting semantically salient words. There are many theories on how to model salience, where the most common probably is the *tf·idf* model. In this model *tf* represents the term frequency and corresponds to the number of times a certain content word, represented by its stem or lemma, occurs within a specific document. This count is usually normalized to prevent a bias towards longer documents. However, in the same manner as the much more frequent function words adding noise

when attempting to identify content words describing the content of a document, very common content words can easily "drown" content words describing a specific document, e.g. within a specific domain. One example of this would be that if we have set of documents discussing the medical treatment of cancer, then certain domain specific content words would to a high extent be common in each of the documents in the set. In order to counter this phenomenon of domain specificity it is common to weigh the term frequency by the number of documents in which the term occurs (Spärck-Jones 1972). This notion of document specificity is often referred to as the inverse document frequency, or simply *idf*, of which you then often take the logarithm. The final weight for a specific word t_i within the particular document d_j from the document set D would then be calculated as $(tf{\cdot}idf)_{i,j} = (n_{i,j}/\sum_k n_{k,j}) * log(|D|/|\{d{:}t_i \in d\}|)$ where $n_{i,j}$ is the number of time t_i occurs in d_j, $n_{k,j}$ is the total number of words in d_j, $|D|$ is the total number of documents in the document set D and $|\{d{:}t_i \in d\}|$ is the number of documents in which t_i occurs in the document set. While (high) term frequency can be seen as a measure of local importance, (low) document frequency of the term in the whole collection of documents can be seen as a measure of general importance; the weights hence tend to filter out common words. The *tf·idf* value for a word will always be greater than or equal to zero.

As we above in part are defining salience as frequency fluctuations between documents, the *tf·idf* model requires a set of documents as well as the necessity of examining all of them while recording in which documents a specific word occurs in order to calculate the final weight of each word. To overcome this requirement of a predefined set several other approaches to capturing salience have been suggested. One such approach, proposed by Ortuño et al. (2002), models salience by tracking the distributional pattern of terms within a document. The authors show that the spatial information of a word is reliable in predicting the relevance of that word to the text being processed, independently of its relative frequency. The base of this observation is that words relevant to a text will normally appear in a very specific context, concentrated in a region of the text, presenting large frequency fluctuations; i.e. keywords come in bursts. The *burstiness* of a word is calculated using the standard deviation of the distance between different occurrences of the same word in the text. However, words that occur only with large distances between occurrences usually have a high standard deviation by chance, so the standard deviation is divided by the mean distance between occurrences. This way the salience model only relies on the document currently under consideration.

The problem with counting terms on a lexical level is that the relation between the terms is not always what it seems to be, at least not by only looking at the constituting characters. Rather, the relation between words and concepts is many-to-many. For example, we have synonymy, where a number of words with same "meaning" have very different lexical appearances. In this case lexical term matching misses relevant frequency conflations, thus impacting recall negatively. A hypothetical example would be that we have a document D such as $D = \{kitten, dog, pussy, cat, mouser, doggie, feline\}$. It is quite obvious, given that you know the meanings of the words occurring in the document, that the document is mainly about cats. However, to a system relying on lexical string matching the document would seem to be about seven different concepts to the same degree. With a little creative stemming, conflating *dog* and *doggie*, we might even come to the conclusion that it is mainly about dogs, as the other terms would not easily be matched.

The Meaning of Words

Modeling of the meaning of words has always been an elusive task in natural language processing. Words are nothing more than sounds or a sequence of graphemes until they become associated with an object, an action or some characteristic. Words therefore not only come to denote objects, phenomena or ideas in the physical world, but also gain a connotative substance based on how and when they are used (Mill 1843). In the Saussurean tradition this connotation, or meaning, is seen as to arise from the relative difference between words in a linguistic system. According to Saussure this constantly restructured system of differences is negotiated through social activity in a community of users (Saussure 1916). Two types of relations constitute the base of this difference, where syntagmatic relations concern positioning and paradigmatic relations act as functional contrasts (substitution). The meaning of a word is thus defined by the company it keeps.

Word Space Models

Until the early nineties most of the work in statistical language learning was concentrated on syntax (Charniak 1993). However, with the induction of Latent Semantic Analysis (Dumais et al. 1988) a whole new field of lexical statistical semantics sprang into existence and today enjoys considerable attention in current research on computational semantics.

The general idea behind word space models is to use statistics on word distributions in order to generate a high-dimensional vector space – a concept space. In this vector space context vectors whose relative directions are assumed to indicate semantic similarity represent the words. The basis of this assumption is the distributional hypothesis (Harris 1954), according to which words that occur in similar contexts also tend to have similar properties (meanings/functions). From this follows that if we repeatedly observe two words in the same (or very similar) contexts, then it is not too far fetched to assume that the words also mean similar things, or at the very least to some extent share properties. Furthermore, depending on how we model these contexts we should be able to capture different traits. We should for instance be able to populate the word space model with syntagmatic relations if we collect information about which words that tend to co-occur, and with paradigmatic relations if we collect information about which words that tend to share neighbors (Sahlgren 2006).

This approach is often seen as a solution to semantic difficulties in NLP like synonymy and hypernymy, which are not captured by the traditional use of the vector space model. It should however be noted that word space methods still are oblivious to the nature of the lexical strings they are tracking. Therefore, given a definition of what constitutes a meaning bearing token in a particular language, it usually is advantageous to perform stemming and/or stop word removal, depending on e.g. the level of morphology of the language in question. Such considerations taken into account a word space model can be applied to basically any language. For instance, word space models have been applied to, among many other languages; English, German, Spanish, Swedish as well as Japanese (Hassel 2005, Sahlgren & Karlgren 2005, Sahlgren 2006).

HOLISTIC SUMMARIZATION

The traditional way to perform summarization based on text extraction methods is to rank the sentences in a source document for their respective appropriateness for inclusion in a summary of this document. These sentences are then concatenated into a summary, which is delivered to the user. This conjugate is seen as containing the sentences most central to the topic of the input text, thus being a representative summary. As contrast, the idea behind a holistic view on summarization is that summaries should be weighed for fitness as a whole, already in the production step. This means that no prejudice is exercised on individual sentences – all sentences are treated as equal. Instead it is their ability to co-operate in forming an overview of the source text that is judged upon.

In order to evaluate this fitness we need to have some way of comparing a source document with one or more summary candidates for content similarity. This is accomplished by letting the concepts we have accumulated by tracking co-occurrences of words in contexts, i.e. by use of Hyperspace Analogue to Language (HAL; Lund & Burgess 1996), Latent Semantic Analysis (LSA; Landauer et al. 1998) or Random Indexing (RI; Kanerva et al. 2000), form document signatures. An example of how context information can be gathered into vectors with RI can be seen in Figure 1.

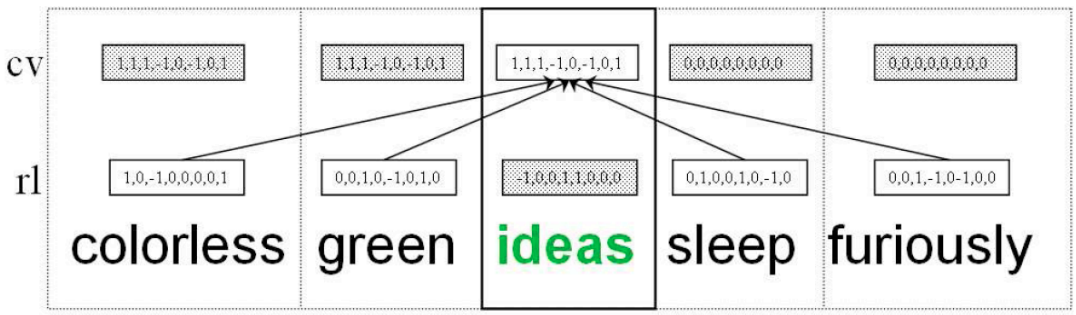

Figure 1. A Random Indexing context window focused on the token ideas, taking note of the co-occurring tokens. The row marked as cv *represents the continuously updated context vectors, and the row marked as* rl *the static randomly generated index labels (random labels), which in practice act as addable meta words. Grayed-out fields are not involved in the current token update.*

Analogously to how we projected the words' semantic representations into a concept space we can now, by letting the document/summary descriptions be weighted the aggregate of the concept vectors of the words contained in that document/summary, project the documents into a multi-dimensional document space.

As with the concepts we can now measure the semantic similarity between the document being summarized and a proposed summary by taking the *cosine* angle between the content vectors of the two. When using this approach to measure document similarity it is common to normalize the vectors by the length of the documents in order to avoid the case where the longer a summary is, the more likely it is to be similar to the original text. Here it might also well be noted that this optimization of semantic similarity between the source document and the considered summary is not in any way constrained to computationally generated summaries. The summaries being evaluated and selected from could in practice be produced by any means, even being man-made.

The HolSum Summarizer

We will now exemplify what we have discussed so far with a summarization system that embodies both the notion of near language independent, portable summarization and that of taking a holistic view of the summary under consideration. HolSum (Hassel 2007) is a text summarizer that aims at providing overview summaries by generating a set of summary candidates, from which it presents the summary most similar to the source to the user. In practice this means that HolSum tries to represent the various (sub)topics of the source text in the summary to the same extent as they occur in the source text.

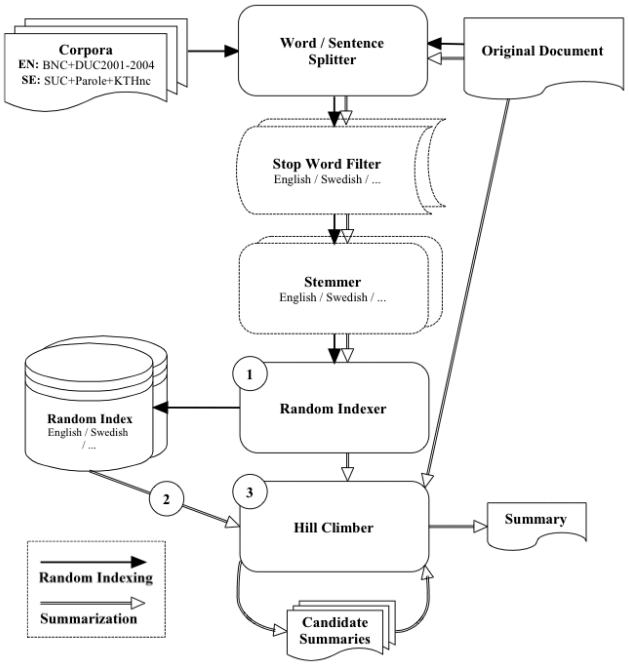

Figure 2. A detailed schematic overview of the HolSum system, with its core language independent properties numbered. (1) being the acquisition of semantic knowledge, (2) the application of the acquired semantic knowledge, and (3) the semantic navigation of the space of possible summaries.

HolSum is trainable for most languages provided a tokenizer that can segment the text into the desired extraction units (e.g. sentences) and tokens suitable for co-occurrence statistics (e.g. words), as defined by that language. Apart from the obligatory sentence and word splitter, and the optional stop word filter and stemmer/compound splitter, there are three main areas of interest in the HolSum system. These three areas, marked with the numbers one through three in the HolSum system layout

(see Figure 2), are the acquisition semantic knowledge, the application of the acquired semantic knowledge and, lastly, the semantic navigation of the space of possible summaries.

(1) The acquisition of semantic knowledge is carried out by using Random Indexing to build concept representations – context vectors – for a very large vocabulary. Even though we have chosen to use RI as means for acquiring the co-occurrence statistics on words, you could basically use any sufficient model for acquiring such semantic knowledge. It should not, at least in theory, make any difference whether you use for instance RI (Kanerva et al. 2000), LSA (Landauer et al. 1998) or HAL (Lund & Burgess 1996). These models are equally language independent given a definition of what constitutes a meaning bearing token (i.e. a "content word") in a specific language, and do all use vectors as their representation for concepts.

(2) In this model, document signatures are formed by aggregating context vectors (i.e. the constituting words' co-occurrence vectors). This aggregation is not an approach specific to our model; rather it is common practice when using word space models as means for document representation. Nevertheless, it is not, from a linguistic point of view, a particularly appetizing approach to the encapsulation of a documents semantic content, albeit one that clearly improves on the traditional vector space model (Hassel & Sjöbergh 2006). The Achilles' heal of this approach, theorywise, is that while the formation of concept representations, in Random Indexing, shies away from the bag-of-words approach in that it has the ability to capture syntactically governed semantic relations between words, the document representations regress into a bag-of-concepts model. Even so, relenting to the model we have there still is room for different models of salience weights when producing these document signatures. In the context of HolSum two such models have been evaluated; the traditional *tf·idf* model and the standard deviation of word occurrences (Hassel 2007).

(3) As mentioned, the document signatures crafted in (2), being vectors as they are, can be positioned in a high-dimensional space where they can be compared for content similarity. Lacking a set of man-made summaries to compare and choose from, a suitable set of summaries must be computationally generated. In the current architecture this is performed by using a greedy search algorithm starting in some initial summary, e.g. the leading sentences of the text being summarized, and iteratively reshaping the summary by swapping sentences from the source text in and out of the summary, until no better summary is found (according to some criteria). Using this approach it is obvious that we risk getting stranded in a local optimum, however, it is not feasible to exhaustively explore the entire document space in the search of the globally best summary. Furthermore, we do not even know how many "best" summaries there are for the current text, which would be interesting in itself perhaps being a measure of the text's "summaribility", which leaves us with little information on whether we should restart the search or not. Despite these objections the approach has performed reasonably well in standard evaluations.

Evaluation of the Portable Holistic Approach

The HolSum approach has mainly been evaluated along the lines of the evaluation setup for task 2 in the Document Understanding Conferences 2004 (DUC; Over & Yen 2004), providing both a well known evaluation data set and metrics, both a baseline and a human ceiling as well as a score of other summarization systems as reference. In task 2 for DUC 2004 the assignment was to produce short multi-document summaries focused by news event clusters (50 clusters, ~10 documents per cluster). The target size for each cluster summary was <= 665 bytes (alphanumerics, whitespace, and punctuation included). The baseline for this task was simply the first 665 bytes of text taken from the most recent document in each cluster and the human ceiling was calculated by taking the mean performance of each of the eight man-made summaries authored for each cluster comparing it to the remaining seven using ROUGE (Lin 2004). In this case we do not necessarily need to normalize the vectors by the document length since the first document in the pair-wise comparisons always is the full documents and the second always is a summary very close to the length of 665 bytes. However, if we also would like to investigate what length of the summary would be more suitable we would also have normalize by length. The system has also been evaluated for Swedish (Hassel & Sjöbergh 2005). The data used for building the conceptual representations for English was comprised of over 100 million words, while the Swedish representations were built on merely 34 million words.

The impact of the dimensionality chosen for the Random Indexing step has been evaluated by running the experiments several times, building semantic representations using a variation of dimensionalities (250, 500 and 1000 dimensions). The results show little variation over different dimensionalities (see Table 1). This means that as long as one does not choose too few dimensions, the dimensionality is not a parameter that needs considerable attention. For each dimensionality the mean performance using four different random seeds was also calculated, since there is a slight variation in how well the method works with different random projections. The dimensionality showing the most variation in the experiments spanned 33.8-34.4% ROUGE-1 (see Table 1). Variations for the other dimensions were slightly less (Hassel & Sjöbergh 2006). The best systems in DUC 2004 scored roughly 39% (Over & Yen 2004), with many of the participating systems scoring below 34%.

	DUC 2004	**DUC 2001-2004**
Human-ceiling	42.6	39.7
Holistic-250	33.9	32.0
Holistic-500	34.2	32.3
Holistic-1000	34.1	32.4
Holistic-RAW	32.7	30.9
Holistic-noRI	30.3	28.5
Baseline-lead	31.0	28.3

Table 1. ROUGE-1 scores, in %, for different dimensionality choices of the context vectors. RAW *indicates no use of stemming and stop word filtering, and* noRI *uses a traditional* tf·idf *weighted vector space model instead of Random Indexing.*

This shows that by applying a set of basic text processing tools with latent semantic analysis one can device a highly portable extraction-based summarization method that, in contrast to most extraction-based systems, does not rank the individual extract segments contributing to the summary. Instead it compares complete summaries to the original text, and chooses the best summary candidate it can find by a simple search strategy.

APPLICATIONS OF AUTOMATIC TEXT SUMMARIZATION

This section describes the application areas of automatic text summarization in order of appearance. The very first application area for automatic text summarization was to create abstracts/extracts from articles without abstracts to be stored in library systems together with the title and author name, (Luhn 1959). At that time one could not store the whole article digitally in the library system due to storage constraints.

Today there is a wide range of application areas for automatic text summarization, the most common and obvious one is in information retrieval. We can already observe it in the result list of search engines where a summarized part of each retrieved document is presented interweaved together with the search terms of the user, the so-called snippets. We can consider these snippets to be a crude form of user adapted text summaries. Another possible application is in the mass media area. Today a news article is written by a journalist, but when typesetting the newspaper the article is shorten manually to the appropriate size so that it can fit in the layout, in between the advertisement. In parallel the same article is also typeset for the web, WAP or SMS text messages. An experiment is described in Dalianis et al. (2004) where both manual editors and the SweSum text summarizer (Dalianis 2000) where given the task to summarize 334 news texts written in Swedish to the appropriate format for the newspaper Sydsvenska Dagbladet. The manually cut down texts were compared with the automatically summarized texts and it was found that the texts where almost identical. Both the editors and the SweSum text summarizer cut down and summarized the texts mainly from the end. The same experiment was carried out for SMS format (maximum 160 characters) and the results from SweSum were considered suitable to be used directly in news paper production.

Business Intelligence systems or news monitoring systems are today very common where one surveys a large flow of news media, this news flow can be summarized so the user can obtain an

overview of the stream before deciding if she should click on the news summary and read the complete news article. One nice live application is the Columbia News Blaster, which takes several news articles describing the same topic and summarizes it to one single news flash (McKeown & Radev 1999, McKeown et al. 2003). Further one might need a multilingual multi-document automatic text summarizer, in which case one could use MEAD (Radev et al., 2004).

If we go to the area of medicine and biomedicine we find several attempts to use automatic text summarization and also the closely related area natural language generation to adapt both text and data to different user groups such as patients, physicians, nurses and scientists. In Hirst et al. (1997) a system is presented that from medical digital libraries produces user adapted information towards individual patients' specific needs, summarized from information on surgery of breast cancer to living with diabetes but also general health education. If we look at generation of text from source data Portet et al. (2009 describe a system that takes survey data from a baby at a neonatal clinic and generates a textual description for several different user groups such as the clinicians, the parents or even the relatives and friends of the patient. The textual description contains information that is adapted to the interest and needs of each user group.

Another system is PERSIVAL, which is described in McKeown et al. (2001). PERSIVAL generates user-adapted information both for patients and physicians, and uses as input the patient record of the patient to find what topics the generated text should contain. PERSIVAL then searches for the relevant information in external resources and summarizes it to the relevant level of the user. The text that is constructed for patients' origins from several consumer health texts, while the text constructed for physicians is collected from medical journal articles.

FUTURE RESEARCH DIRECTIONS

We can envision a system that learns more and more about a given domain or language the more the system processes texts and therefore can create progressively more fine tuned text summaries. The HolSum application shows promising results in this direction as the Random Indexing approach to word space models allows for the document being summarized to be folded into the knowledge base of the summarizer before the actual summarization begins, thus potentially utilizing relations between words even in the current document.

In the future specialist books will be ordered on demand. A specialist book will be user adapted and customized according to who orders the book, delivering content at an appropriate complexity level. The book will contain different parts from other books, pictures, sound samples, videos and maybe even material from other languages that are translated automatically to the target language. An extension to this application could be to in a coherent way incorporate diagrams and tables into the summary, such that the quality of the summary further increases. We already see some advancement in multi-media summarization. In the wake of this development we also believe that more research should be put into the presentation of summarized material, perhaps incorporating the temporal factor. This could, for instance, be done by presenting a steady flow of news on different topics as constantly reshaping documents, much in the manner of Google Wave, where contents could be adapted, evolved and honed as the news story progresses. This would also suit the medical domain, by providing snap-shots of a patient's health status and progression.

Another possible near-future application is that newspapers will be produced completely automatically. The articles will of course be written by journalists, but all typesetting including layout of advertisements will be performed by systems that have embedded text summarization both for the news article and for the advertisement texts.

CONCLUSION

As we have seen in this chapter there are several linguistically motivated text processing tasks that need to be addressed from a language independence point of view. Among these are such tasks as *tokenization* and *sentence segmentation*. Furthermore, we have discussed the impact of these tasks on the representation of the contents of documents, i.e. *document signatures*. The impact of *stemming* and *compound splitting* on document signatures is also discussed. These document signatures can then be compared for similarity using the *vector space model*. Related to this discussion is the notion of *salience* and how one can promote topically relevant words and *concept representations* in the

document signatures. These concept representations are crafted by gathering *word co-occurrence statistics* used for grouping semantically related words in a *word space model*. This model is based on the *distributional hypothesis* according to which words occurring in similar contexts tend to have similar *meaning* or *function*.

Lastly, we have presented the notion of *holistic summarization* were a set of summaries are internally ranked and the "best" summary presented to the user, rather than the traditional conjugate of individually ranked sentences. This notion has been exemplified with the *HolSum summarizer*, which employs the *Random Indexing* word space model for crafting concept representations. These concept representations are used to form document signatures for both the *input text* as well as *generated summaries*, which are compared for *semantic similarity* in a *document space*. The discussion up to this point supports the portability of the approach, as it requires no sophisticated tools, although stop word filtering and simple stemming clearly improves the results. Even though access to large amounts of raw (unannotated) text is needed for good performance, this is for many languages readily available, for instance on the World Wide Web.

REFERENCES

Bacchin, M., Ferro, N. & Melucci, M. (2002). The effectiveness of a graph-based algorithm for stemming. In *Proceedings of the 5th International Conference on Asian Digital Libraries* (pp. 117–128). London, UK: Springer-Verlag.

Charniak, E. (1993). *Statistical Language Learning*. Cambridge, Massachusetts: MIT Press.

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

Crystal, D. (1987). *The Cambridge encyclopedia of language*. Cambridge University Press.

Dalianis, H. (2000). *SweSum - A Text Summarizer for Swedish*. Technical report TRITA-NA-P0015, IPLab-174, NADA, KTH, October 2000.

Dalianis, H., Hassel, M., de Smedt, K., Liseth, A., Lech, T.C. & Wedekind, J. (2004). Porting and evaluation of automatic summarization. In Holmboe, H. (ed.) *Nordisk Sprogteknologi 2003. Årbog for Nordisk Språkteknologisk Forskningsprogram 2000-2004* (pp. 107-121). Museum Tusculanums Forlag

Dumais, S. T., Furnas, G. W., Landauer, T. K. & Deerwester, S. C. (1988). Using latent semantic analysis to improve information retrieval. In *Proceedings of CHI'88: Conference on Human Factors in Computing* (pp. 281–285), Washington DC, USA. May 15-19 1988.

Endres-Niggemeyer, B., Maier, E. & Sigel, A. (1995). How to Implement a Naturalistic Model of Abstracting: Four Core Working Steps of an Expert Abstractor. *Information Processing & Management*, 31(5):631–674.

Grefenstette, G. & Tapanainen, P. (1994). What is a word, what is a sentence? Problems of tokenization. In *Proceedings of the 3rd International Conference on Computational Lexicography* (pp. 79–87), Budapest, Hungary.

Harris, Z. S. (1954). Distributional structure. *Word*, 10(23):146–162.

Hassel, M. (2005). Word Sense Disambiguation Using Co-Occurrence Statistics on Random Labels. In *Proceedings of Recent Advances in Natural Language Processing 2005*. Borovets, Bulgaria.

Hassel, M. (2007). *Resource Lean and Portable Automatic Text Summarization*. Doctoral thesis, School of Computer Science and Communication, Royal Institute of Technology. Stockholm, Sweden.

Hassel, M. & Sjöbergh, J, (2005). A Reflection of the Whole Picture Is Not Always What You Want, But That Is What We Give You. In *Proceedings of "Crossing Barriers in Text Summarization Research" workshop at Recent Advances in Natural Language Processing 2005*, Borovets, Bulgaria.

Hassel, M. & Sjöbergh, J. (2006). Towards holistic summarization: Selecting summaries, not sentences. In *Proceedings of Language Resources and Evaluation 2006*. Genoa, Italy.

Hirst, G., DiMarco, C., Hovy, E. & Parsons, K. (1997). Authoring and generating health-education documents that are tailored to the needs of the individual patient. In *Proceedings of the Sixth International Conference on User Modeling* (pp. 107-118). Vienna, New York: Springer.

Jing, H. (2000). Sentence Reduction for Automatic Text Summarization. In *Proceedings of the 6th Applied Natural Language Processing Conference* (pp. 310–315), Seattle, Washington. April 29–May 4, 2000.

Jing, H. & McKeown, K. R. (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* (pp178–185), Seattle, Washington. April 29–May 4, 2000.

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

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

Liddy, E. D. (1991). Discourse-level Structure of Empirical Abstracts: An Exploratory Study. *Information Processing and Management*, 27(1):550–81.

Lin, C-Y. (2004). ROUGE: A Package for Automatic Evaluation of Summaries. In *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*. Barcelona, Spain. July 25 - 26, 2004.

Luk, R. W. P. (1994). An IBM-PC environment for Chinese corpus analysis. In *Proceedings of the 15th conference on Computational Linguistics* (pp. 584–587). Morristown, New Jersey, USA.

Luhn, H.P. (1959). The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development* (pp. 159-165).

Lund, K. & Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments & Computers*, 28(2), 203-208.

McCarthy, P. M., Guess, R. H. & McNamara, D. S. (2009). The components of paraphrase evaluations. *Behavior Research Methods*, 41(3):682-690.

McKeown, K.R., Barzilay, R., Chen, J., Elson, D., Evans, D., Klavans, J., Nenkova, A., Schiffman, B. & Sigelman, S. (2003). Columbia's Newsblaster: New Features and Future Directions. In *Proceedings of the Human Language Technology Conference*, Vol. II. Edmonton, Canada.

McKeown, K. R., Chang, S-F., Cimino, J., Feiner, S. K., Friedman, C., Gravano, L., Hatzivassiloglou, V., Johnson, S., Jordan, D. A., Klavans, J. L., Kushniruk, A., Patel, V. & Teufel, S. (2001). PERSIVAL, a System for Personalized Search and Summarization over Multimedia Healthcare Information. In *Proceedings of the Joint ACM/IEEE Conference on Digital Libraries (JCDL-01)*, pages (pp. 331–340). Roanoke, Virginia, June 2001.

McKeown, K. R. & Radev, D. R. (1999). Generating summaries of multiple news articles. In *Advances in Automatic Text Summarization* (pp. 381-389). Edited by I. Mani and M. T. Maybury. Cambridge, Massachusetts: The MIT Press.

Mill, J. S. (1843). *A System Of Logic, Raciocinative and Inductive*. London.

Ortuño, M., Carpena, P., Bernaola-Galvan, P., Munoz, E. & Somoza, A. (2002). Keyword detection in natural languages and DNA. *Europhysics Letters*, 57:759–764.

Over, P. & Yen, J. (2004). *An introduction to DUC 2004 intrinsic evaluation of generic new text summarization systems*. http://www-nlpir.nist.gov/projects/duc/pubs/2004slides/duc2004.intro.pdf

Portet, F., Reiter, E., Gatt, A., Hunter, J., Sripada, S., Freer, Y. & Sykes, C. (2009). Automatic generation of textual summaries from neonatal intensive care data. *Artificial Intelligence*, 173(7-8):789-816, Essex, United Kingdom: Elsevier Science Publishers Ltd.

Radev, D., Allison T., Blair-Goldensohn, S., Blitzer, J., Çelebi, A., Dimitrov, S., Drabek, E., Hakim, A., Lam, W., Liu, D., Otterbacher, J., Qi H., Saggion, H., Teufel., Topper, M., Winkel, A. & Zhang, Z. 2003. MEAD-a platform for multi-document multilingual text summarization, in Proceedings of The fourth international conference on Language Resources and Evaluation, LREC 2004, Lisbon, Portugal.

Sahlgren, M. (2006). *The Word-Space Model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces*. Doctoral thesis, Department of Linguistics, Stockholm University. Stockholm, Sweden.

Sahlgren, M. & Karlgren, J. (2005). Automatic Bilingual Lexicon Acquisition Using Random Indexing of Parallel Corpora. *Journal of Natural Language Engineering, Special Issue on Parallel Texts*, 11(3).

Salton, G. (1971). *The SMART Retrieval System – Experiments in Automatic Document Processing*. Upper Saddle River, New Jersey, USA: Prentice-Hall, Inc.
Saussure, F. de. (1916). *Course in General Linguistics* (trans. Roy Harris, 1983). Duckworth, London.
Sjöbergh, J. & Kann, V. (2004). Finding the correct interpretation of Swedish compounds a statistical approach. In *Proceedings of Language Resources and Evaluation 2004* (pp. 899–902). Lisbon, Portugal.
Spärck-Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21.
Xu, J. & Croft, B. (1998). Corpus-based Stemming using Co-occurrence of Word Variants. *ACM Transactions on Information Systems*, 16(1):61–81.

ADDITIONAL READING SECTION

Dalianis, H., Hassel, M., Wedekind, J., Haltrup, D., de Smedt, K. & Christopher, T. L. (2003). From SweSum to ScandSum: Automatic text summarization for the Scandinavian languages. In Holmboe, H. (ed.) *Nordisk Sprogteknologi 2002: Årbog for Nordisk Språkteknologisk Forskningsprogram 2000-2004* (pp. 153-163). Copenhagen: Museum Tusculanums Forlag.
Hassel, M. & Mazdak, N. (2004). FarsiSum - A Persian text summarizer. In *Proceedings of Computational Approaches to Arabic Script-based Languages workshop at COLING'04*. Geneva, Switzerland. August 28, 2004.
Hassel, M. (2003). Exploitation of Named Entities in Automatic Text Summarization for Swedish. In the *Proceedings of NODALIDA'03 - 14th Nordic Conference on Computational Linguistics.* Reykjavik, Iceland. May 30-31, 2003.
Hassel, M. & Dalianis,H. (2005). Generation of Reference Summaries. In *Proceedings of 2nd Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics*. Poznan, Poland. April 21-23, 2005.
Hovy, E. H. and C-Y. Lin. 1998. Automating Text Summarization in SUMMARIST. In I. Mani and M. Maybury (eds), *Advances in Automated Text Summarization*. Cambridge, Massachusetts: MIT Press.
Jing, H. & McKeown, K. R. (1999). The Decomposition of Human-Written Summary Sentences. In M. Hearst, G. F., & Tong, R. (editors), In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 129–136). University of California, Berkeley.
Lin, C-Y. (2005). ROUGE: Recall-oriented understudy for gisting evaluation. http://berouge.com/
Mani, I. (1999). *Advances in Automatic Text Summarization*, Cambridge, Massachusetts: MIT Press
Marcu, D. (1999). The automatic construction of large-scale corpora for summarization research. In *Proceedings of 22nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99)* (pp. 137-144). Berkeley, California.
Nenkova, A., Passonneau, R. & McKeown, K.R. (2007). The Pyramid Method: Incorporating Human Content Selection Variation in Summarization Evaluation. *ACM Transactions on Speech and Language Processing (TSLP)*, Volume 4, Issue 2. May 2007.
Radev, D. R., Blair-Goldensohn, S., Zhang, Z. & Raghavan, R. S. (2001). Newsinessence: A System for Domain-Independent, Real-Time News Clustering and Multi-Document Summarization. In *Proceedings of Human Language Technology Conference (HLT 2001)* San Diego, California.
Radev, D. R., Jing, H., & Budzikowska, M. (2000). Centroid-Based Summarization of Multiple Documents: Sentence Extraction, Utility-Based Evaluation, and User Studies. In Udo Hahn. Chin-Yew Lin, Inderjeet Mani, and Dragomir R. Radev (editors), *Proceedings of the Workshop on Automatic Summarization at the 6th Applied Natural Language Processing Conference and NAACL 2000*, Seattle, Washington.
Saggion, S. 2009. SUMMA - A Robust and Adaptable Summarization Tool. Traitement Automatique des Langues. Volume 49 – n◦ 2/2008, pages 103 à 125.
Sahlgren, M. (2005). An Introduction to Random Indexing. In *Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering*. Copenhagen, Denmark. August 16, 2005.

Sjöbergh, J. (2007). Older versions of the ROUGEeval summarization evaluation system were easier to fool. *Journal of Information Processing and Management, Special Issue on Summarization* (pp. 1500-1505), Volume 43, Issue 6, November, ISSN:0306-4573.
Spärk-Jones, K. & Galliers, K. R.(1995). Evaluating Natural Language Processing Systems: An Analysis and Review. *Lecture Notes in Artificial Intelligence*, number 1083. Springer.
Winkel, A & Radev., D. R. (2002). MEADeval: An evaluation framework for extractive summarization. http://tangra.si.umich.edu/clair/meadeval/meadeval.pdf

KEY TERMS & DEFINITIONS

Corpus: A large collection of writings of a specific kind or on a specific subject. A corpus should be balanced so that it is representative for the language or domain that covers. Corpora are often used in statistical language technology for both training of machine learning approaches and evaluation of the end result.

Holistic: The philosophical view that all the properties of a given system (may it be physical, biological, chemical, social, economic, mental or linguistic) cannot be determined by its component parts alone. Instead, the system as a whole governs in an important way how the parts behave within the system. In the context of summarization this view stands in contrast of the customary ranking of individual sentences within a text being summarized.

Intermediary: Components that are intended to be consumed or converted into another component, and therefore not destined to appear in the final production system. These system or component states need to be sufficiently efficacious when initially moving a language technology system to new domain or language. Given the availability of necessary resources these intermediates are often honed in later stages of research.

Language independent: The ability to seamlessly move from one language or domain to another. This transfer should be facilitated by the systems' ability to adapt to the irregularities natural languages exhibit, e.g. by learning new instances of language use, without the use of hard-coded language specific knowledge or need of specifically annotated or structured data.

Portable: The portability of a system or method is most often an issue of cost. When lacking the necessary resources to build advanced systems the ability to transfer an intermediary system to other languages or domains, with as little effort as possible, becomes crucial. This applies both to the system itself as to the evaluation of the system in its new context.

Resource lean: Resources can be many things, e.g. human resources, economic resources, spatial or temporal resources, data resources etc. The research on summarization in focus in this chapter is in some sense lean on human resources since while it demands large bodies of data for training, this data does, however, not need to be annotated nor structured in any way, and can be collected "in the wild" (e.g. it can be text already in existence, produced for entirely different purposes). Thus it most certainly is resource lean regarding the other identified resource types, since

1. structuring and annotation of data takes time and requires quite a bit of human effort;
2. human labor (usually) is more time consuming than the computerized counterpart;
3. humans desire more space than (most) computers require;
4. time, space and human labor most definitely cost money.

ROUGE scores: Automatic summarization evaluation using unigram co-occurrences between summary pairs. Has in studies been shown to correlate surprising well with human evaluations, but has also met critique for not performing well on longer word chains as well as in earlier versions being quite easy to fool. Inspired by BLEU scores, a similar metric used in machine translation.

Text abstraction: The production of text that is a shorter non-redundant rendition of the source text. This is accomplished by parsing the original text in a deep linguistic way, interpreting the text

semantically into a formal representation, finding new more concise concepts to describe the information carried by the source text and then re-generated in a new shorter form. An abstract may thus contain words and expressions not used in the original text, while still conveying the same basic information as the source text.

Text extraction: Generation of a shorter content representation by reusing a subset of the original text. This relies on the identification of 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 genre or language specific 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.