Chapter 15
Coreference Applications to Summarization

Josef Steinberger, Mijail Kabadjov, and Massimo Poesio

Abstract In this chapter we discuss the connection between anaphora/coreference resolution and summarization. The discussion follows the summarization framework based on Latent Semantic Analysis (LSA), however, the ideas can be applied to any sentence-scoring approach. After describing the ways of combining basic (lexical) features of the summarizer with those received from the coreference resolution system we try to answer the question whether coreference resolution helps to improve the quality of selected content even if coreference resolution systems are still far from perfect. Both single-document and multi-document summarization branches are discussed. Then we focus on post-processing techniques to improve the referential clarity and coherence of extracted summaries.

15.1 Introduction

Information about anaphoric relations could be beneficial for summarization which involves extracting (possibly very simplified) discourse models from text. We investigate exploiting automatically extracted information about the coreferring expressions in a text for two different aspects of the summarization task: firstly to enrich the representation of a text, from which a summary is then extracted; and secondly, to check that the anaphoric expressions contained in the summary thus extracted still have the same interpretation that they had in the original text.

Josef Steinberger
University of West Bohemia, Faculty of Applied Sciences, NTIS Centre, Department of Computer Science and Engineering, Univerzitni 8, Pilsen 306 14, Czech Republic
Mijail Kabadjov
University of Essex, School of Computer Science and Electronic Engineering, Wivenhoe Park, Colchester CO4 3SQ, United Kingdom, e-mail: malexa@essex.ac.uk
Massimo Poesio
University of Essex, School of Computer Science and Electronic Engineering, Wivenhoe Park, Colchester CO4 3SQ, United Kingdom. e-mail: poesio@essex.ac.uk

In the following discussion we follow the latent semantic analysis (LSA [22]) summarization framework proposed in [11] and later improved in [47]. This approach follows what has been called a term-based approach [16]. In term-based summarization, the most important information in a document is found by identifying its main 'terms' (also sometimes called 'topics'), and then extracting from the document the most important information about these terms. Such approaches are usually classified as 'lexical' approaches or 'coreference- (or anaphora-) based' approaches. Lexical approaches to summarization use word similarity and other lexical relations to identify central terms [3]; we would include among these previous approaches based on LSA [11, 47]. Coreference- or anaphora-based approaches[1] [2, 7, 6, 52] identify these terms by running a coreference- or anaphoric resolver over the text. Using both lexical and anaphoric information to identify the main terms was discussed in [49].

Summarization by sentence extraction may produce summaries with 'dangling' anaphoric expressions – expressions whose antecedent has not been included in the summary, and therefore cannot be interpreted or are interpreted incorrectly. A method for using anaphoric information to check the entity coherence of a summary once this has been extracted has been proposed in [49]. The algorithm checks that the interpretation of anaphoric expressions in a summary is consistent with their interpretation in the original text.

Even if the discussion is strongly tied with the LSA-based framework the ideas of using coreference features in combination with basic summarizer's features to improve content selection can be used by any sentence-scoring approach. The post-processing entity-coherence checking can be used irrespective of the method the summary was produced with.

The chapter is organized as follows. Section 15.2 gives overview of the research in text summarization. Then we discuss the potential of using anaphora/coreference resolution in summarization together with the related work (section 15.3). In section 15.4 we discuss ways for incorporating coreference information into vector-space-based source representations. Then, in section 15.5, the improvement in content selection when using coreference is discussed (the case of single-document summarization). We show both upper bound performance when manual coreference annotations are used and performance when automatic tools are involved. In section 15.6 an algorithm for checking the entity-coherence of a summary is shown. Since section 15.5 discusses the case of single-document summarization and intra-document coreference section 15.7 goes further with mutli-document summarization and inter-document coreference. In the last sections (15.8 and 15.9) conclusions and pointers to further reading are given.

[1] We rather use the term 'coreference resolution' as a more general term to anaphora resolution. However, when we discuss single-document summarization the term refers to the task of identifying successive mentions of the same discourse entity (intra-document coreference resolution/anaphora resolution), as opposed to the task of 'inter-document coreference resolution' appropriate in the case of multi-document summarization which involves collecting all information about an entity, including information expressed by appositions and other predicative constructions.

15.2 Text Summarization

A basic processing model for Text Summarization, proposed by Sparck-Jones [46] comprises three main stages: source text interpretation to construct a source representation, source representation transformation to form a summary representation, and summary text generation. More practically-motivated approaches use shallow linguistic analysis and only partially cover the processing model. However, more ambitious ones attempting all three stages using deep semantic analysis have been proposed in the literature.

The first approaches were based on shallow linguistic analysis such as word frequencies [25], cue phrases (e.g., "in conclusion", "in summary") and location (e.g., title, section headings) [10]. Later, machine learning approaches that combine a number of surface features have been proposed [21]. There are also more sophisticated approaches, but still working at the surface level, exploiting cohesive relations like coreference [2, 7, 52] and lexical cohesion [3] to identify salience or purely lexical approaches trying to identify 'implicit topics' by conflating together words using methods inspired by Latent Semantic Analysis LSA [11, 47]. There are also approaches purely based on discourse structure (e.g., RST) [29]. And finally there are knowledge-rich approaches, where the source undergoes a substantial semantic analysis during the process of filling in a predefined template [31] or the source data is available in a more structured way (i.e., events have been identified already) [30].

Summarization evaluation, a closely related issue, is a particularly challenging problem. Manual evaluation gives more precise results, however, it is a subjective task and thus it needs to be done in a large scale and it is very expensive. On the other hand, automatic evaluation methods give just rough image of summary quality, however, they need usually just some reference documents, like human-written abstracts.

Summaries are usually scored from two different perspectives: linguistic quality (readability) and content quality. *Linguistic quality* is often assessed by human annotators. They assign a value from a predefined scale to each summary. The quality aspects contain: grammaticality, non-redundancy, reference clarity, coherence and structure.

Content quality is often measured by comparison with a model summary. For sentence extracts, it is often measured by *co-selection*. It finds out how many sentences overlap in a model and an automatic summary. It can be measured by simple precision/recall figures. The main problem with P&R is that human judges often disagree on what the top p% most important sentences are. With *Relative Utility* [44] the model summary represents all sentences of the input document with confidence values for their inclusion in the summary. For example, the confidence value of a sentence can correspond to the number of annotators who selected that sentence.

If the model summary is a free text written by a human or the system does not only extract the most important sentences the evaluation method has to work with smaller units (words, n-grams or phrases). The *ROUGE* (Recall-Oriented Understudy for Gisting Evaluation) family of measures, which are based on the similarity of n-grams, was firstly introduced in 2003 [23]. Suppose a number of annotators

created reference summaries - reference summary set (*RSS*). The ROUGE-*n* score of a candidate summary is computed as follows:

$$\text{ROUGE-}n = \frac{\sum_{C \in RSS} \sum_{gram_n \in C} Count_{match}(gram_n)}{\sum_{C \in RSS} \sum_{gram_n \in C} Count(gram_n)}, \tag{15.1}$$

where $Count_{match}(gram_n)$ is the maximum number of *n*-grams co-occurring in a candidate summary and a reference summary and $Count(gram_n)$ is the number of *n*-grams in the reference summary. Notice that the average *n*-gram ROUGE score, ROUGE-*n*, is a recall metric. There are other ROUGE measures, such as ROUGE-L – a longest common subsequence measure – and ROUGE-SU4 – a bigram measure that enables up to 4 unigrams inside of bigram components to be skipped [24].

Imagine we have two reference summaries and two candidate summaries:

(15.2)
ref1: Eight killed in Finnish school massacre
ref2: Student kills eight in Finnish school shooting
cand1: Shooting in a Finnish school: eight dead
cand2: Finland: school gunman left suicide note

The first candidate summary is very similar to both reference summaries, the second is only partly relevant (*Finland school*). ROUGE-1 will simply count terms co-occurring in a candidate summary and reference summaries denominated by the number of terms in reference summaries (recall). For *cand1* $(4+5)/(6+7) = 0.615$, for *cand2* $(1+1)/(6+7) = 0.153$. ROUGE-2 works the same, but on the bigram level. In the case of *cand1* there is only one bigram (*Finnish school*) which is contained in both reference summaries: $(1+1)/(5+6) = 0.182$, *cand2* doesn't contain any bigram from reference. ROUGE-L is similar to ROUGE-1 but the terms have to be in the same order. ROUGE-L of *cand1* will be $(3+3)/(6+7) = 0.46154$ (*in Finnish school* matches in both reference summaries). ROUGE-L of *cand2* will be the same as its ROUGE-1. ROUGE-SU4 allows up to 4 unigrams between the bigram items, unigrams count also to give advantage to fragments that contain the reference words but not bigrams. ROUGE-SU4 of *cand1*is $(7+7)/(20+26) = 0.304$ (*[in], [in, finnish], [in, school], [finnish], [finnish, school], [school], [eight]*). The second candidate will match only *school*: $(1+1)/(20+26) = 0.043$. For more more details see [24].

The Pyramid method is a semi-automatic evaluation method [35]. Its basic idea is to identify summarization content units (SCUs), which are not bigger than a clause. They are used for comparison of information in summaries. SCUs that appear in more model summaries will get greater weights, so a pyramid will be formed after SCU annotation of model summaries. At the top of the pyramid there are SCUs that appear in most of the summaries and thus they have the greatest weight. The lower in the pyramid the SCU appears, the lower its weight is because it is contained in fewer summaries. The SCUs in a peer summary are then compared against an existing pyramid to evaluate how much information is agreed between the peer and the model summaries. However, this promising method still requires some annotation work.

Another possibility to evaluate summaries is to use *extrinsic (task-based)* methods. They measure the performance of using the summaries for a certain task (e.g. document categorization [27], information retrieval [45] or question answering [33]).

In recent years DUC/TAC[2] took a leading role in the summarization roadmap. The tracks have been developing since DUC2000 untill DUC2007, starting with single-document summarization and moving on to multi-document summarization [37]. The next step from multi-document summarization was update summarization[3] which piloted in DUC2007 and represented the main track in TAC2008 and TAC2009. Aspect summarization[4] in TAC2010 (and 2011) goes even one step futher to understanding the text content by incorporating information extraction. Language-independence gives the picture another dimension, the new goals are multilingual summarization [13, 53] or cross-language summarization [54].

15.3 Coreference and summarization

In [7] the following news article was used to illustrate why being able to recognize coreference chains may help in identifying the main topics of a document.

(15.3) PRIEST IS CHARGED WITH POPE ATTACK

A Spanish priest was charged here today with attempting to murder the Pope. *Juan Fernandez Krohn*, aged 32, was arrested after a man armed with a bayonet approached the Pope while he was saying prayers at Fatima on Wednesday night. According to the police, *Fernandez* told the investigators today that *he* trained for the past six months for the assault. . . . If found guilty, *the Spaniard* faces a prison sentence of 15-20 years.

As Boguraev and Kennedy point out, the title of the article is an excellent summary of the content: an entity (*the priest*) did something to another entity (*the pope*). Intuitively, this is because understanding that *Fernandez* and *the pope* are the central characters is crucial to providing summaries of texts like these.[5] Among the clues that help us to identify such 'main characters,' the fact that an entity is repeatedly mentioned is clearly important.

---

[2] The National Institute of Standards and Technology (NIST) initiated the Document Understanding Conference (DUC) series [55] to evaluate automatic text summarization. Its goal is to further progress in summarization and enable researchers to participate in large-scale experiments. Since 2008 DUC has moved to TAC (Text Analysis Conference) [56] that follows the summarization evaluation roadmap with new or upgraded tracks.

[3] When producing an update summary of a set of topic-related documents the summarizer assumes prior knowledge of the reader determined by a set of older documents of the same topic. The update summarizer thus must solve a novelty vs. redundancy problem.

[4] In the aspect summarization scenario a given list of core information aspects for different event types should be addressed in the automatic summaries.

[5] In many non-educational texts only an 'entity-centered' structure can be clearly identified, as opposed to a 'relation-centered' structure of the type hypothesized in Rhetorical Structures Theory and which serves as the basis for discourse structure-based summarization methods [20, 41].

Methods that only rely on lexical information to identify the main topics of a text, such as the lexical-based methods discussed in the previous section, can only capture part of the information about which entities are frequently repeated in the text. As example (15.3) shows, stylistic conventions forbid verbatim repetition, hence the six mentions of *Fernandez* in the text above contain only one lexical repetition, *'Fernandez'*. The main problem are pronouns, that tend to share the least lexical similarity with the form used to express the antecedent (and anyway are usually removed by stopword lists, therefore do not get included in the SVD matrix). The form of definite descriptions (*the Spaniard*) doesn't always overlap with that of their antecedent, either, especially when the antecedent was expressed with a proper name. The form of mention which more often overlaps to a degree with previous mentions is proper nouns, and even then at least some way of dealing with acronyms is necessary (cf. *European Union* / *E.U.*). On the other hand, it is well-known from the psychological literature that proper names often are used to indicate the main entities in a text. What coreference resolution can do for us is to identify which discourse entities are repeatedly mentioned, especially when different forms of mention are used. Instead of a summary, the approach in [7] extracted list of those linguistic expressions which refer to the most prominent objects mentioned in the discourse.

In [2] coreference-based summarization was used in an information retrieval scenario. Automatically generated document summaries were used to support relevance judgments of the IR user. Firstly, coreference is employed in retrieving referential relations between the terms of the original IR query and the terms of the documents that are considered to be relevant. Coreference is the main clue also in generation of the document summary when sentences containing entities of the query are identified. The system follows the coreference chains and, according to several heuristics, selects a subsequence of sentences of highest relevance. The approach further provides lexically informative substitute expressions for anaphors that may, out of their original context, become incomprehensible.[6]

The use of coreference resolution for the scenario of generic summarization is proposed in [1]. There is no user query that prescribes relevant entities on which the summary should focus. They try to find a single coreference chain which corresponds to the central entity the text is about. The subsequence of sentences in which this entity is salient is then extracted.

According to the representative approaches, coreference information is employed in different processing stages. The first stage consists of relating some terms of the query to coreferring occurrences in the document pool over which the application runs. If there is no query the system first has to find the focus of the document pool (e.g. central entities/events). This may be considered as a special case of the cross-document coreference resolution problem. At the second processing stage, the system follows a coreference chain(s) in order to select a subsequence of sentences that would form a document summary. The last stage is to identifying coreferring

[6] The approach deals with object coreference and event coreference. They further consider the issue of referential relations beyond the identity relation covering a few domain-specific special cases.

antecedents for anaphoric occurrences in order to provide maximally informative substitute expressions.

15.4 Coreference Knowledge Representation

Purely lexical methods determine the main 'topics' of a document on the basis of the simplest possible notion of term, simple words, or n-grams. In this section we will see, however, that coreference information can be easily integrated in a mixed lexical / coreference representation by generalizing the notion of 'term' used to include as well, and counting a discourse entity d as occurring in sentence s whenever the coreference resolver identifies a noun phrase occurring in s as a mention of d.

15.4.1 Coreference Chain Elements Substitution

The simplest way of using coreference information is to keep using only words as terms, and use anaphora resolution as a pre-processing step [49]. That is, after identifying the coreference chains within each text, replace all referring nominal expressions with the first element of their anaphoric chain. The modified text could then be used as an input for any summarization approach. However, in [49] it was shown that this simple approach does not lead to improved results.

15.4.2 Word Terms and Discourse Entities within the same Space

A better approach, it turns out, is to generalize the notion of term, treating coreference chains as another type of term that may or may not occur in a sentence. The idea is illustrated in Table 15.1, where the summarizer's input matrix contains two types of terms: terms in the lexical sense (i.e., words/n-grams) and terms in the sense of discourse entities, represented by coreference chains. The representation of a sentence then specifies not only if that sentence contains a certain word, but also if it contains a mention of a discourse entity. With this representation, the chain terms may tie together sentences that mention the same entity even if they do not contain the same lexical item.

The resulting matrix is effectively an enriched vector-space representation entailing not only lexical, but also anaphoric information, which can then be used as input to summarizers.

Table 15.1 Aggregative source representation.

	unit$_1$	**unit**$_2$	**unit**$_3$	**...**
ngram$_1$ **ngram**$_2$ **...**	Lexical info			
entity$_1$ **entity**$_2$ **...**	Entity info			

15.5 Using Coreference for Salient Content Selection

Next, we briefly introduce latent semantic analysis (LSA, [22]) and the summarization approach based on it. Then we summarize the single-document summarization experiments described in [49] to determine the upper bound performance and to show real performance when an automatic anaphora resolver is used.

15.5.1 LSA-Based Summarization

LSA is a fully automatic mathematical/statistical technique for extracting and inferring relations of expected contextual usage of words in passages of discourse.

It has been extensively used in NLP applications including information retrieval [58] and text segmentation [57], and also summarization [11], which was later extended by Steinberger and Ježek [47].

The approach first builds a term-by-sentence matrix from the source, then applies Singular Value Decomposition (SVD) which finds the latent (orthogonal) dimensions, which in simple terms correspond to the different topics discussed in the source, and finally uses the resulting matrices to identify and extract the most salient sentences.

More formally, firstly a term-by-sentence matrix **A** is created. Each element a_{ij} of **A** represents the weighted frequency of term i in sentence j and is defined as:

$$a_{ij} = L(i,j) \cdot G(i), \tag{15.4}$$

where $L(i,j)$ is the local weight of term i in sentence j and $G(i)$ is the global weight of term i in the text. The weighting scheme found to work best uses a binary local weight and an entropy-based global weight:

$$\begin{aligned} L(i,j) = 1 \text{ if term } i \text{ appears at least once in sentence } j; \\ \text{otherwise } L(i,j) = 0 \end{aligned} \tag{15.5}$$

$$G(i) = 1 - \sum_j \frac{p_{ij} \log(p_{ij})}{log(n)}, p_{ij} = \frac{t_{ij}}{g_i}, \tag{15.6}$$

where t_{ij} is the frequency of term i in sentence j, g_i is the total number of times that term i occurs in the whole text and n is the total number of sentences.

Let's illustrate the approach by the following simplified example – summarizing these 5 article titles:

(15.7)
s1: Cyclone smashes into Bangladesh coast
s2: Ferocious cyclone hits Bangladesh coast
s3: Thousands evacuate as cyclone approaches Bangladesh
s4: WFP offers emergency food to victims
s5: US offers help for cyclone victims

Basically, there are two topics: (1) *the cyclon hit* and (2) *help offered to victims*. The term-by-sentence matrix **A**:

$$\begin{array}{r} \\ A = \end{array} \begin{array}{ccccc|l} s1 & s2 & s3 & s4 & s5 & \\ 1.86 & 1.86 & 1.86 & 0 & 1.86 & cyclone \\ 1 & 0 & 0 & 0 & 0 & smashes \\ 1.68 & 1.68 & 1.68 & 0 & 0 & bangladesh \\ 1.43 & 1.43 & 0 & 0 & 0 & coast \\ 0 & 1 & 0 & 0 & 0 & ferocious \\ 0 & 1 & 0 & 0 & 0 & hits \\ 0 & 0 & 1 & 0 & 0 & thousands \\ 0 & 0 & 1 & 0 & 0 & evacuate \\ 0 & 0 & 1 & 0 & 0 & approaches \\ 0 & 0 & 0 & 1 & 0 & WFP \\ 0 & 0 & 0 & 1.43 & 1.43 & offers \\ 0 & 0 & 0 & 1 & 0 & emergency \\ 0 & 0 & 0 & 1 & 0 & food \\ 0 & 0 & 0 & 1.43 & 1.43 & victims \\ 0 & 0 & 0 & 0 & 1 & us \\ 0 & 0 & 0 & 0 & 1 & help \end{array} \tag{15.8}$$

The next step is to apply the Singular Value Decomposition (SVD) to matrix **A**. The SVD of an $m \times n$ matrix is defined as:

$$\mathbf{A} = \mathbf{U} \cdot \mathbf{S} \cdot \mathbf{V}^T, \tag{15.9}$$

where **U** ($m \times n$) is a column-orthonormal matrix, whose columns are called left singular vectors. The matrix contains representations of terms expressed in the newly

created (latent) dimensions. $\mathbf{S}$ $(n \times n)$ is a diagonal matrix, whose diagonal elements are non-negative singular values sorted in descending order. $\mathbf{V}^T$ $(n \times n)$ is a row-orthonormal matrix which contains representations of sentences expressed in the latent dimensions. The dimensionality of the matrices is reduced to r most important dimensions and thus, we receive matrices $\mathbf{U'}$ $(m \times r)$, $\mathbf{S'}$ $(r \times r)$ a $\mathbf{V'}^T$ $(r \times n)$.

From a mathematical point of view, SVD derives a mapping between the m-dimensional space specified by the weighted term-frequency vectors and the r-dimensional singular vector space.

From an NLP perspective, what SVD does is to derive the *latent semantic structure* of the document represented by matrix A: i.e. a breakdown of the original document into r linearly-independent base vectors which express the main 'topics' of the document.

The following numbers represent 2-dimensional SVD depomposition of our example term-by-sentence matrix $\mathbf{A}$:

	U'			**S'**			**V'**^{T}					
	*dim*1	*dim*2										
cyclone	0.72	0.07										
smashes	0.11	−0.06										
bangladesh	0.53	−0.27										
coast	0.31	−0.17										
ferocious	0.11	−0.06										
hits	0.11	−0.06										
thousands	0.10	−0.04		*dim*1	*dim*2		*s*1	*s*2	*s*3	*s*4	*s*5	
evacuate	0.10	−0.04		5.1	0.0		0.55	0.57	0.50	0.08	0.36	*dim*1
approaches	0.10	−0.04		0.0	3.32		−0.19	−0.21	−0.13	0.69	0.66	*dim*2
WFP	0.02	0.21										
offers	0.12	0.58										
emergency	0.02	0.21										
food	0.02	0.21										
victims	0.12	0.58										
us	0.07	0.20										
help	0.07	0.20										

(15.10)

As $\mathbf{V'}^T$ contains sentences expressed by relative importance of the top topics they mention and $\mathbf{S'}$ contains topic importance, by multiplying these matrices we receive the latent space of matrix $\mathbf{V'}^T$ in which vector length of each dimension correspond to its importance. Let's call the final matrix $\mathbf{F} = \mathbf{S'} \cdot \mathbf{V'}^T$.

Sentence selection starts with the sentence that has the longest vector in matrix **F** (the vector, the column of **F**, is denoted as f_{best}). After placing it in the summary, the topic/sentence distribution in matrix **F** is changed by subtracting the information contained in that sentence:

$$\mathbf{F}_{(i+1)} = \mathbf{F}_{(i)} - \frac{\mathbf{f}_{best} \cdot \mathbf{f}_{best}^T}{|\mathbf{f}_{best}|^2} \cdot \mathbf{F}_{(i)} \tag{15.11}$$

The vector lengths of similar sentences are decreased, thus preventing inner summary redundancy. After the subtraction the process of selecting the sentence that has the longest vector in updated matrix $\mathbf{F}_{(i+1)}$ and subtracting its information from $\mathbf{F}_{(i)}$ is iteratively repeated until the required summary length is reached.

When we return to our example: $\mathbf{F}_{(0)}$ represents the multiplication of matrices **S'** and $\mathbf{V'}^T$. The longest vector in the matrix has sentence *s2*. This sentence will be selected to the summary and the previous equation will be used to recaluculate the matrix $\mathbf{F} \rightarrow \mathbf{F}_{(1)}$. Notice that the values of sentences *s1* and *s3*, sentences similar to the selected one, are reduced. From the next iteration of the matrix, $\mathbf{F}_{(1)}$, *s5* will be selected.

$$\begin{array}{l} \\ \\ \\ \end{array} \quad \begin{array}{c} \mathbf{F}_{(0)} \\ \begin{array}{ccccc} s1 & \mathbf{s2} & s3 & s4 & s5 \end{array} \\ \begin{array}{l} dim1 \mid \\ dim2 \mid \end{array} \begin{array}{ccccc} 2.784 & \mathbf{2.9} & 2.52 & 0.395 & 1.827 \mid \\ -0.627 & \mathbf{-0.696} & -0.445 & 2.276 & 2.183 \mid \end{array} \end{array} \tag{15.12}$$

$$\begin{array}{c} \mathbf{F}_{(1)} \\ \begin{array}{ccccc} s1 & s2 & s3 & s4 & \mathbf{s5} \end{array} \\ \begin{array}{l} dim1 \mid \\ dim2 \mid \end{array} \begin{array}{ccccc} 0.0090 & 0.0 & 0.036 & 0.538 & \mathbf{0.595} \mid \\ 0.039 & 0.0 & 0.151 & 2.242 & \mathbf{2.478} \mid \end{array} \end{array}$$

15.5.2 Performance Upper Bound

In [49], in order to determine whether anaphoric information might help, and which method of adding anaphoric knowledge to the LSA summarizer is best, they annotated 37 documents from the CAST corpus using the annotation tool MMAX [34].

15.5.2.1 The CAST Corpus

The CAST corpus [36] contains news articles taken from the Reuters Corpus and a few popular science texts from the British National Corpus. Summaries are specified by providing information about the importance of sentences [14]: sentences

are marked as *essential* or *important* for the summary. The corpus also contains annotations for *linked* sentences, which are not significant enough to be marked as important/essential, but which have to be considered as they contain information essential for the understanding of the content of other sentences marked as essential/important.

For acquiring model summaries at specified lengths and getting the sentence scores (for relative utility evaluation) a score of 3 was assigned to the sentences marked as essential, a score of 2 to important sentences and a score of 1 to linked sentences.

15.5.2.2 Evaluation Measure

As a main measure Relative Utility (see section 15.2) was chosen because it could be computed automatically given the already existing annotations in the CAST corpus. RU allows model summaries to consist of sentences with variable ranking.

15.5.2.3 Upper Bound Results

Results for the 15% (resp. 30%) summarization ratio using a variety of summarization evaluation measures are presented in Table 15.2. The tables show that even with perfect knowledge of anaphoric links, the performance when using Substitution method does not change much. The problem that happened in some of the documents was that SVD deteriorated when a frequently used entity was substituted by its full nominal expression. As a result, the score of the sentence was extremely boosted when it contained the mention of the entity. And thus, sentences that contained the mention of this entity were all considered important, no matter what else they contained.

On the other hand, the addition method could potentially lead to substantial improvements.

Table 15.2 Improvement over lexical-based LSA with manually annotated anaphoric information

Summarization Ratio	Lexical LSA	Manual – Substitution	Manual – Addition
15%	0.595	0.573	**0.662**
30%	0.645	0.662	**0.688**

15.5.3 Using an Automatic Tool in Single-Document Summarization

In [49] the question of whether using an automatic anaphora resolution tool can lead to an improved performance was also addressed. For this purpose they used GuiTAR[7], described in the chapter on off-the-shelf tools. We next discuss the performance reported in [49] when the anaphora resolution is incorporated in the summarizer.

15.5.3.1 Does Automatic AR Improve Summarization?

To use GuiTAR, [49] first parsed the texts using Charniak's parser [9]. The output of the parser was then converted into the MAS-XML format expected by GuiTAR by one of the preprocessors that come with the system.[8] Finally, GuiTAR was run to add anaphoric information to the files. The resulting files were then processed by the summarizer.

GuiTAR achieved a precision of 56% and a recall of 51% over the 37 documents; on definite description resolution, a precision of 69% and a recall of 53%; for possessive pronouns resolution, a precision of 53%, recall 53%; finally, for personal pronouns, precision of 44%, recall 46%. The figures are based on simple link-based scoring.

The results obtained by the summarizer using GuiTAR's output are presented in Table 15.3 (RU scores).

Table 15.3 Improvement over lexical-based LSA with GuiTAR.

Summarization Ratio	Lexical LSA	GuiTAR – Substitution	GuiTAR – Addition
15%	0.595	0.530	**0.640**
30%	0.645	0.626	**0.678**

Table 15.3 clearly shows that using GuiTAR and the addition method leads to significant improvements over the baseline LSA summarizer. The improvement in Relative Utility measure was significant (95% confidence by the *t*-test). On the other hand, the substitution method did not lead to significant improvements, as was to be expected given that no improvement was obtained with 'perfect' anaphora resolution (see previous section).

In conclusion, this study showed that (i) we could expect performance improvements over purely lexical LSA summarization using anaphoric information, (ii) significant improvements at least by the Relative Utility score could be achieved even if

[7] Available as open source software at `http://guitar-essex.sourceforge.net/`.

[8] This step includes heuristic methods for guessing agreement features.

this anaphoric information was automatically extracted, and (iii) these results were only achieved, however, using the Addition method.

15.5.3.2 Comparison to State-of-the-art

What Steinberger *et al.*'s work [49] did not show was how well are their results compared with the state-of-the-art, as measured by evaluation over a standard reference corpus such as DUC 2002, and using the standard ROUGE measure.

DUC 2002 included a single-document summarization task, in which 13 systems participated. 2002 is the last version of DUC that included single-document summarization evaluation of informative summaries. Later DUC editions (2003 and 2004) contained a single-document summarization task as well, however only very short summaries (75 Bytes) were analyzed. The DUC-2002 corpus used for the task contains 567 documents from different sources; 10 assessors were used to provide for each document two 100-word human summaries. In addition to the results of the 13 participating systems, the DUC organizers also distributed baseline summaries (the first 100 words of a document). The coverage of all the summaries was assessed by humans.

In DUC 2002, the SEE evaluation tool was used, but in later editions of the initiative the ROUGE measure (see section 15.2) was introduced, which is now widely used.

As shown in Table 15.4, for this particular corpus there is a strong correlation between humans and ROUGE-1 (and ROUGE-L). [49] used all four main ROUGE scores to determine the significance of their results.

Table 15.4 Correlation between ROUGE scores and human assessments.

Score	Correlation
ROUGE-1	0.92574
ROUGE-2	0.80090
ROUGE-SU4	0.78396
ROUGE-L	0.92561

In Table 15.5 there are the ROUGE scores[9] of the purely lexical LSA summarizer; of the summarizer combining both lexical and anaphoric information (LSA+GuiTAR); and of the 13 systems which participated in DUC-2002. We also list a baseline and a random summarizer (the lowest baseline).

[9] All system summaries were truncated to 100 words as traditionally done in DUC. ROUGE version and settings:

```
ROUGEeval-1.4.2.pl -c 95 -m -n 2 -l 100 -s -2 4 -a duc.xml.
```

Table 15.5 ROUGE scores.

System	ROUGE-1	ROUGE-2	ROUGE-SU4	ROUGE-L
28	0.42776	0.21769	0.17315	0.38645
LSA+GuiTAR	**0.42280**	**0.20741**	**0.16612**	**0.39276**
21	0,41488	0.21038	0.16546	0.37543
DUC baseline	0.41132	0.21075	0.16604	0.37535
19	0.40823	0.20878	0.16377	0.37351
LSA	**0.40805**	**0.19722**	**0.15728**	**0.37878**
27	0.40522	0.20220	0.16000	0.36913
29	0.39925	0.20057	0.15761	0.36165
31	0.39457	0.19049	0.15085	0.35935
15	0.38884	0.18578	0.15002	0.35366
23	0.38079	0.19587	0.15445	0.34427
16	0.37147	0.17237	0.13774	0.33224
18	0.36816	0.17872	0.14048	0.33100
25	0.34297	0.15256	0.11797	0.31056
Random	0.29963	0.11095	0.09004	0.27951
17	0.13528	0.05690	0.04253	0.12193
30	0.07452	0.03745	0.02104	0.06985

The performance of the 'lexical only' LSA summarizer is significantly worse only than that of the best system in DUC 2002, system 28, in ROUGE-1, ROUGE-2 and ROUGE-SU4, and significantly better than that of 9 in ROUGE-1, 7 in ROUGE-2, 7 in ROUGE-SU4 and 10 in ROUGE-L of the systems that participated in that competition. However, when anaphoric information is included (LSA+GuiTAR) the summarizer is even better: it is significantly better than 11 systems in ROUGE-1, 9 in ROUGE-2, 9 in ROUGE-SU4 and 13 in ROUGE-L, it is significantly better than the baseline in ROUGE-L at the 90% confidence level, and it is not significantly worse than any of the systems.

15.6 Checking Entity Coherence in Summaries

Anaphoric expressions can only be understood with respect to a context. This means that summarization by sentence extraction can wreak havoc with their interpretation: there is no guarantee that they will have an interpretation in the context obtained by extracting sentences to form a summary, or that this interpretation will be the same as in the original text. Consider the following example.

(15.13) PRIME MINISTER CONDEMNS IRA FOR MUSIC SCHOOL EXPLOSION

(**S1**) [Prime Minister Margaret Thatcher]$_1$ said Monday [[the Irish Republican Army]$_2$ members who blew up [the Royal Marines School of Music]$_3$ and killed [10 bandsmen]$_4$ last week]$_5$ are monsters who will be found and punished.
(**S2**) "[The young men whom we lost]$_4$ were murdered by [common murderers who must be found and brought to justice and put behind bars for a very long time]$_5$," [she]$_1$ said following a tour of [[the school's]$_3$ wrecked barracks]$_6$ in Deal, southeast England.
...
(**S3**) [Gerry Adams, president of [Sinn Fein, the legal political arm of [the IRA]$_2$]$_8$]$_7$ issued a statement disputing [Mrs. Thatcher's]$_1$ remarks, saying "[she]$_1$ knows in [her]$_1$ heart of hearts the real nature of the conflict, its cause and the remedy".
...
(**S4**) "[We]$_8$ want an end to all violent deaths arising out of the present relationship between our two countries," [Adams]$_7$ said.
...
(**S5**) [The IRA]$_2$ claimed responsibility for the explosion, and police said they are looking for [three men with Irish accents who rented a house overlooking [the barracks]$_6$]$_5$.

If sentence S2 were to be extracted to be part of the summary, but S1 was not, the pronoun *she* would not be understandable as it would not have a matching antecedent anymore. The reference to *the school* would also be uninterpretable. The same would happen if S5 were extracted without also extracting S2; in this case, the problem would be that the antecedent for *the barracks* is missing.

Examples such as the one just shown suggest another use for anaphora resolution in summarization – correcting the references in the summary. Our idea was to replace anaphoric expressions with a full noun phrase in the cases where otherwise the anaphoric expression could be misinterpreted. We discuss this method in detail next.

15.6.1 Reference Correction Algorithm

The correction algorithm works as follows.

1. Run anaphora resolution over the source text, and create anaphoric chains.
2. Identify the sentences to be extracted using a summarization algorithm such as the one discussed in the previous sections.
3. For every anaphoric chain, replace the first occurrence of the chain in the summary with its first occurrence in the source text. After this step, all chains occurring in both source and summary start with the same lexical form.
   For example, in the text in (15.13), if sentence S4 is included in the summary, but S3 is not, the first occurrence of chain 7 in the summary, *Adams*, would be substituted by *Gerry Adams, president of Sinn Fein, the legal political arm of the IRA*.
4. Run the anaphoric resolver over the summary.
5. For all nominal expressions in the summary: if the expression is part of a chain in the source text and it is not resolved in the summary (the resolver was not

able to find an antecedent), or if it is part of a different chain in the summary, then replace the anaphoric expression with the head of the first chain expression from the source text.

This method can be used in combination with the summarization system discussed in earlier sections, or with other systems; and becomes even more important when doing sentence compression, because intrasentential antecedents can be lost as well. However, automatic anaphora resolution can introduce new errors. We discuss our evaluation of the algorithm next.

15.6.2 Evaluation

To measure the recall of the reference checker algorithm we would need anaphoric annotations, that were not available for DUC data. The precision was manually measured as follows. To measure the precision of the step where the first occurrences of a chain in the summary were replaced by the first mention of that chain in the source text, a sample of 155 documents was taken and precision was measured by hand, obtaining the results shown in Table 15.6.

Table 15.6 Evaluation of step 3, the first chain occurrence replacement.

Observed state	Overall	Per-doc.
Chains in full text	2906	18.8
Chains in summary	1086 (37.4% of full text chains)	7.0
First chain occurrence in the summary	714 (65.7% of summary chains)	4.6
First chains element with the same lexical form	101 (9.3% of summary chains)	0.7
First chain occurrence replaced	271 (25% of summary chains)	1.7
Correctly replaced	186 (**Precision: 68.6%**)	1.2

We can observe that full texts contained on average 19 anaphoric chains, whereas summaries about 7. In 66% of the summary chains the sentence where the first chain occurrence appeared was selected for the summary, and in 9% there was no need to replace the expression because it already had the same form as the first element of the chain. So overall the first chain occurrence was replaced in 25% of the cases; the precision was 68.6%. This suggests that the success in this task correlates with the quality of the anaphora resolver.

After performing anaphora resolution on the summary and computing its anaphoric chains, the anaphors without an antecedent are replaced. A sample of 86 documents was analyzed and again the precision was measured by hand. Overall, 145 correct replacements were made in this step and 65 incorrect, for a precision of 69%. Table 15.7 analyzes the performance on this task in more detail.

Table 15.7 Evaluation of step 5, checking the comprehensibility of anaphors in the summary. (Replaced + means that the expressions were correctly replaced; replaced – that the replacement was incorrect). S = summary, FT = full text.

Observed state	Correct	Incorrect
In a chain only in S	16 (67%)	8 (33%)
In a chain only in FT	32 (70%) (replaced +)	14 (30%) (replaced –)
In the same chain in FT and S	336 (83%)	69 (17%)
In a different chain in FT and S (correct in FT)	81 (72%) (replaced +)	32 (28%) (replaced +)
In a different chain in FT and S (incorrect in FT)	39 (77%) (replaced –)	12 (23%) (replaced –)
Replacements overall	145 (**69%**)	65 (31%)

The first row of the table lists the cases in which an expression was placed in a chain in the summary, but not in the source text. In these cases, our algorithm does not replace anything.

Our algorithm however does replace an expression when it finds that there is no chain assigned to the expression in the summary, but there is one in the source text; such cases are listed in the second row. We found that this replacement was correct in 32 cases; in 14 cases the algorithm replaced an incorrect expression.

The third row summarizes the most common case, in which the expression was inserted into the same chain in the source text and in the summary. That is, the first element of the chain in the summary is also the first element of the chain in the source text. When this happens, in 83% of cases GuiTAR's interpretation is correct; no replacement is necessary.

Finally, there are two subcases in which different chains are found in the source text and in the summary (in this case the algorithm performs a replacement). The fourth row lists the case in which the original chain is correct; the last, cases in which the chain in the source text is incorrect. In the first column of this row are the cases in which the anaphor was correctly resolved in the summary but it was substituted by an incorrect expression because of a bad full text resolution; the second column shows the cases in which the anaphor was incorrectly resolved in both the full text and the summary, however, replacement was performed because the expression was placed in different chains.

15.6.3 A Summary Before and After Reference Checking

Examples (15.14) and (15.15) illustrate the difference between a summary before and after reference checking. A reader of (15.14) may not know who *the 71-year-old Walton* or *Sively* are, and what *store* is referred to in the text. In addition, the pronoun *he* in the last sentence is ambiguous between *Walton* and *Sively*. On the other hand, *the singer* in the last sentence can be easily resolved. This is because the

chains *Walton*, *Sively* and *the store* do not start in the summary with the expression used for the first mention in the source text. These problems are fixed by step 3 of the reference checker. The ambiguous pronoun *he* in the last sentence of the summary is resolved to *Sively* in the summary and *Walton* in the source text[10]. Because the anaphor occurs in a different chain in the summary and in the full text, it has to be substituted by the head of the first chain occurrence noun phrase, *Walton*. *The singer* in the last sentence is resolved identically in the summary and in the full text: the chains are the same, so there is no need for replacement.

(15.14) WAL-MART FOUNDER PITCHES IN AT CHECK-OUT COUNTER
(summary before reference checking)

> ***The 71-year-old Walton***, considered to be one of the world's richest people, grabbed a note pad Tuesday evening and started hand-writing merchandise prices for customers so their bills could be tallied on calculators quickly. *Walton*, known for his down-home style, made a surprise visit to ***the store*** that later Tuesday staged a concert by *country singer Jana Jea* in *its* parking lot. *Walton* often attends promotional events for *his* Arkansas-based chain, and ***Sively*** said *he* had suspected the boss might make an appearance. ***He*** also joined *the singer* on stage to sing a duet and led customers in the Wal-Mart cheer.

(15.15) WAL-MART FOUNDER PITCHES IN AT CHECK-OUT COUNTER
(summary after reference checking)

> ***Wal-Mart founder Sam Walton***, considered to be one of the world's richest people, grabbed a note pad Tuesday evening and started hand-writing merchandise prices for customers so their bills could be tallied on calculators quickly. *Walton*, known for his down-home style, made a surprise visit to ***his store in this Florida Panhandle city*** that later Tuesday staged a concert by *country singer Jana Jea* in *its* parking lot. *Walton* often attends promotional events for *his* Arkansas-based chain, and ***store manager Paul Sively*** said *he* had suspected the boss might make an appearance. ***Walton*** also joined *the singer* on stage to sing a duet and led customers in the Wal-Mart cheer.

15.7 Cross-Document Coreference and Multi-document Summarization

Multi-document summarization brings a cross-document dimension for coreference resolution. Even if instead of a coreference resolver a more general multilingual named entity disambiguator and geo tagger were used results confirmed the improvement [19].

[10] The previous sentence in the source is: "*Walton* continued talking with customers during the concert."

15.7.1 Multilingual Entity Recognition and Disambiguation

The disambiguation of entities in free text consists of first recognizing a named entity in the text and then grounding it to an entity in the real world, say a location, a person or an organization. The EMM system includes modules for entity recognition and disambiguation in 19 languages [42, 43]. These modules are now being used as part of the summarization task to add normalized and disambiguated structural information as input for LSA. Next, we describe in more details two EMM modules for entity disambiguation: one for geographical locations and another one for persons and organizations.

Historically (e.g., MUC-7 [15]), place name recognition consisted of identifying references to locations in text and disambiguating them from homographic person names or from other homographic words. For instance, there are places called 'Javier' (Spain) and 'Solana' (Philippines), and there are places called 'And' (Iran), 'To' (Ghana) and 'Be' (India). Within the EMM framework, the recognition in [42] go beyond this MUC task by furthermore disambiguating between various homographic place names in order to identify precise latitude-longitude information and to put a dot on a map. For example, there are 15 different locations each with the names of 'Berlin', 'Paris' and 'Roma', and there are 205 places called 'San Antonio'. In their experiments Kabadjov *et al.* [19] make use of that EMM component to augment the term-by-sentence matrix (Table 15.1) with disambiguated and normalized location information.

They additionally make use of the multilingual person and organization recognition tools described in [43]. What distinguishes this tool from others is its high multilinguality and, most of all, the functionality to map name variants referring to the same entity. Name spelling variation is not only a multilingual phenomenon, but it is even very frequent within a single language. In [43] up to 170 ways of spelling the same name were identified. By recognizing and mapping existing name variants for the same entity and by feeding this normalized information to the LSA representation (as described in sec. 15.4.2), additional useful cross-document links can be established.

Augmenting the initial matrix with information about disambiguated entities naturally does not only provide stronger inter-sentential cohesion (i.e., the LSA clusters sentences from different documents that make reference to the same entities), but also multilingual capabilities inherited by the multilingual entity disambiguation. Thus, this approach to summarization is not only multi-document, but also multilingual.[11]

[11] The multilingual named entity disambiguator and geo-tagger developed at the JRC have already been used for cross-lingual linking of multilingual news clusters produced by the EMM system [51].

Table 15.8 Improvement with coreference information in multi-document summarization results.

Approach	ROUGE-1	ROUGE-2	ROUGE-SU4
Lexical only TAC-08	0.355	0.088	0.123
Lexical only	0.359	0.087	0.125
Entities only	0.333	0.076	0.113
Lexical + entities	0.367	0.093	0.13

15.7.2 Multi-document Summarization Results

Using a standard English corpus for Summarization research developed by the US National Institute for Standards and Technology (NIST) for the 2008 Text Analysis Conference (hereafter TAC-08), Kabadjov *et al.* [19] obtained promising, though not statistically significant, improvements over a lexical-only baseline ranked in the top 15%-24% across all evaluation metrics at the 2008 TAC competition. For the following experiments the popular ROUGE metric to evaluate the performance was used. The results are presented in table 15.8.

On the standard multi-document summarization task (see table 15.8), we include the official scores at TAC-08 of a lexical-only summarizer that was as a baseline for the experiments (cf. first row of the table) as well as an improved version of it referred to as 'lexical only' (cf. second row). The third row shows the performance when only entities (only coreference information) are used. The fourth row of table 15.8 corresponds to the results obtained by combining the lexical and coreferring expressions.

From table 15.8 can be seen that the performance of the 'lexical+entities' version of the system is higher than the 'lexical only' version, the baseline, but the improvement is not statistically significant. Using only entities for summarization is not sufficient.

[19] note that the EMM entity disambiguation module used in their experiment was optimized for precision, since in the EMM's context the vast amounts of data (i.e., over 80K articles processed per day) makes up for the compromise on recall. However, in the TAC-08 context there is substantial room for improvement of the entity disambiguation recall by bringing in an intra-document coreference resolution system, such as GuiTAR [18]. In the light of this, the performance of both 'entity only' and 'lexical+entities' approaches can be improved by working on the full coreference resolution systems.

15.8 Conclusion

As pointed out by the literature discussed throughout the chapter, using coreference information does lead to improved selection of salient content both in single-document and multi-document lexical-based extractive summarization. Naturally, however, the way in which coreference information is used matters. For instance, substitution did not result in significant improvements even with perfect coreference knowledge. The addition method, on the other hand, produced significant improvements in both cases when annotated coreference information and an automatic resolver were used.

Coreference resolution can be used as well as a post-processing step to correct entity mentions in the summary. Although, the overall performance is highly dependent on the quality of coreference resolution. Hence, a high-precision resolver is needed for this task. Automatic evaluation of linguistic quality of summaries could not be implemented without the use of coreference resolution either.

15.9 Further Reading

We recommend to start with the line of papers we followed in this chapter. The first approach that exploited cohesive relations like coreference to identify salience was proposed in [7].

Coreference-oriented text representation was also used in other works. In modeling local coherence as defined by the Centering theory [12], Barzilay and Lapata [4] put forward a similar document representation to the one discussed in the chapter, called 'entity grid' which was essentially an entity-by-sentence matrix. Though, as opposed to [48] they did not attempt to combine it with a purely lexical representation. Combining several sources of knowledge in the vector space model, among which key words and entities, was independently proposed by R. Steinberger et al. [51] while working on language-independent news cluster representation for cross-lingual news cluster linking. For that representation, they developed multilingual tools for geo-tagging and entity disambiguation [43].[12]

There has been increasing interest recently among text summarization researchers in post-processing techniques to improve the referential clarity and coherence of extractive summaries, and among natural language generation researchers in generating referential expressions in context. The GREC tasks [5] are aimed at researchers in both of these groups, and the objective is the development of methods for generating chains of referential expressions for discourse entities in the context of a written discourse, as is useful for post-processing extractive summaries and repeatedly edited texts (such as Wikipedia articles).

[12] The use of the multilingual tools in higher-level applications can be seen at `http://emm.newsexplorer.eu/`.

ROUGE has been widely used for content quality evaluation so far. In [50] including entities to the n-gram based measure led to an improved performance measured in the AESOP TAC'09 task. One of the five aspects of linguistic quality evaluation in DUC/TAC has been referential clarity. Evaluation of this bit of linguistic quality was proposed in [38]. The approach yielded a satisfactory correlation with human-assigned linguistic quality.

Acknowledgments

This work was supported by project "NTIS - New Technologies for Information Society", European Centre of Excellence, CZ.1.05/1.1.00/02.0090, and by project MediaGist, EU's FP7 People Programme (Marie Curie Actions), no. 630786.

References

[1] Azzam, S., Humphreys, K. and Gaizauskas, R. Using coreference chains for text summarization. In: Proceedings of the ACL'99 Workshop on Conference and its Applications, Baltimore. ACL (1999)
[2] Baldwin, B. and Morton, T.S.: Dynamic coreference-based summarization. In: Proceedings of EMNLP'98 (Granada, Spain). ACL (1998)
[3] Barzilay, R. and Elhadad, M.: Using lexical chains for text summarization. In: I. Mani and M. Maybury (eds.), Advances in Automated Text Summarization. MIT Press (1997)
[4] Barzilay, R. and Lapata, M.: Modeling local coherence: An entity-based approach. In: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (Ann Arbor, US). ACL (2005)
[5] Belz, A., Kow, E. and Viethen, J.: The GREC Named Entity Generation Challenge 2009: Overview and Evaluation Results. In: Proceedings of ACL-IJCNLP'09 Workshop on Language Generation and Summarisation. ACL (2009)
[6] Bergler, S., Witte, R., Khalife, M., Li, Z. and Rudzicz, F.: Using knowledge-poor coreference resolution for text summarization. In: Proceedings of DUC'03 (Edmonton, Canada) NIST (2003)
[7] Boguraev, B. and Kennedy, C.: Salience-based content characterisation of text documents. In: I. Mani and M. Maybury (eds.), Advances in Automated Text Summarization. MIT Press (1997)
[8] Bontcheva, K., Dimitrov, M., Maynard, D., Tablan, V. and Cunningham, H.: Shallow methods for named entity coreference resolution. In: Chanes de rfrences et rsolveurs d'anaphores, workshop TALN'02 (Nancy, France). (2002)

[9] Charniak, E.: A maximum-entropy-inspired parser. In Proceedings of NAACL'00 (Philadelphia, US). ACL (2000)
[10] Edmundson, H.: New methods in automatic extracting. Journal of the Association for Computing Machinery 16(2), pages 264–285. ACM (1969)
[11] Gong, Y. and Liu, X.: 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, Louisiana, United States), pages 19–25. ACM (2001)
[12] Grosz, B., Aravind, J. and Scott, W.: Centering: A framework for modelling the local coherence of discourse. In: Computational Linguistics 21(2), pages 203–225. ACL (1995)
[13] Giannakopoulos, G., El-Haj, M., Favre, B., Litvak, M., Steinberger, J. and Varma, V.: TAC 2011 multiling pilot overview. In: Proceedings of of the Text Analysis Conference 2011, Gaithersburg Maryland USA. NIST (2011)
[14] Hasler, L., Orasan, C. and Mitkov, R.: Building better corpora for summarization. In: Proceedings of Corpus Linguistics (Lancaster, United Kingdom). UCREL, Lancaster University (2003)
[15] Hirschman, L.: MUC-7 coreference task definition, version 3.0. In: Proceedings of the 7th Message Understanding Conference. NIST (1998)
[16] Hovy, E. and Lin, C.: Automated text summarization in SUMMARIST. In: I. Mani and M. Maybury (eds.), Advances in Automated Text Summarization. MIT Press (1997)
[17] Kabadjov, M., Poesio, M. and Steinberger, J.: Task-Based Evaluation of Anaphora Resolution: The Case of Summarization. In: RANLP Workshop "Crossing Barriers in Text Summarization Research" (Borovets, Bulgaria). ACL (2005)
[18] Kabadjov, M.: A Comprehensive Evaluation of Anaphora Resolution and Discourse-new Recognition. PhD thesis. University of Essex 2007
[19] Kabadjov, M., Steinberger, J., Pouliquen, B., Steinberger, R. and Poesio, M.: Multilingual statistical news summarisation: Preliminary experiments with english. In: Proceedings of IAPWNC at the IEEE/WIC/ACM WI-IAT (Milano, Italy). IEEE Computer Society (2009)
[20] Knott, A., Oberlander, J., O'Donnell, M. and Mellish, C.: Beyond elaboration: The interaction of relations and focus in coherent text. In: T. Sanders, J. Schilperoord and W. Spooren (eds.), Text representation: linguistic and psycholinguistic aspects. John Benjamins (2001)
[21] Kupiec, J., Pedersen, J., Chen, F.: A trainable document summarizer. In: Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 68–73. Seattle, Washington ACM (1995)
[22] Landauer, C.T. and Dumais, S.: A solution to plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge. In: Psychological Review 104, 211–240. American Psychological Association (1997)

[23] Lin, C. and Hovy, E.: Automatic evaluation of summaries using n-gram co-occurrence statistics. In: Proceedings of HLT-NAACL (Edmonton, Canada). ACL (2003)
[24] Lin, C.: ROUGE: A package for automatic evaluation of summaries. In: Proceedings of the Workshop on Text Summarization Branches Out (Barcelona, Spain). ACL (2004)
[25] Luhn, H.: The automatic creation of literature abstracts. In: IBM Journal of Research and Development 2(2), pages 159–165. IBM (1958)
[26] Mani, I. (ed.): Proceedings of the Workshop on Intelligent and Scalable Text Summarization at the Annual Joint Meeting of the ACL/EACL, Madrid. ACL (1997)
[27] Mani, I., Firmin, T., House, D., Klein, G., Sundheim, B., Hirschman, L.: The TIPSTER Summac Text Summarization Evaluation. In: Proceedings of the 9th Meeting of the European Chapter of the Association for Computational Linguistics. ACL (1999)
[28] Mani, I., Maybury, M. (eds.): Advances in Automatic Text Summarization. MIT Press (1999)
[29] Marcu, D.: From discourse structures to text summaries. In: Mani [26]
[30] Maybury, M.: Generating summaries from event data. In: Mani and Maybury [28]
[31] McKeown, K., Radev, D.: Generating summaries of multiple news articles. In: Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pages 74–82. Seattle, Washington. ACM (1995)
[32] Mitkov, R.: Robust pronoun resolution with limited knowledge. In: Proceedings of COLING'98 (Montreal, Canada). ACL (1998)
[33] Morris, A., Kasper, G., Adams, D.: The Effects and Limitations of Automatic Text Condensing on Reading Comprehension Performance. In: Information Systems Research 3(1), pages 17–35. INFORMS (1992)
[34] Mueller, C. and Strube, M.: MMAX: A tool for the annotation of multi-modal corpora. In: Proceedings of the 2nd IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems (Seattle, US). Morgan Kaufmann (2001)
[35] Nenkova, A., Passonneau, R., McKeown, K.: The pyramid method: incorporating human content selection variation in summarization evaluation. ACM Transactions on Speech and Language Processing 4(2). ACM (2007)
[36] Orasan, C., Mitkov, R. and Hasler, L.: CAST: a computer-aided summarization tool. In: Proceedings of EACL'03 (Budapest, Hungary). ACL (2003)
[37] Over, P., Dang, H., Harman, D.: DUC in context. Information Processing and Management 43(6), Special Issue on Text Summarisation (Donna Harman, ed.), pages 1506–1520. Elsevier (2007).
[38] Pitler, E., Louis, A. and Nenkova, A.: Automatic Evaluation of Linguistic Quality in Multi-Document Summarization. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics (Uppsala, Sweden), pp 544–554. ACL (2010)

[39] Vieira, R. and Poesio, M.: An empirically-based system for processing definite descriptions. In: Computational Linguistics, 26(4). ACL (2000)
[40] Poesio, M. and Kabadjov, M.: A general-purpose, off-the-shelf anaphora resolution module: implementation and preliminary evaluation. In: Proceedings of LREC (Lisbon, Portugal). ELRA (2004)
[41] Poesio, M., Stevenson, R., Di Eugenio, B. and Hitzeman, J.: Centering: A parametric theory and its instantiations. In: Computational Linguistics 30(3). ACL (2004)
[42] Pouliquen, B., Kimler, M., Steinberger, R., Ignat, C., Oellinger, T., Blackler, K., Fuart, F., Zaghouani, W., Widiger, A., Forslund, A. and Best, C.: Geocoding multilingual texts: Recognition, disambiguation and visualisation. In: Proceedings of the 5th LREC (Genoa, Italy), pages 53–58. ELRA (2006)
[43] Pouliquen, B. and Steinberger, R.: Automatic construction of multilingual name dictionaries. In: Cyril Goutte, Nicola Cancedda, Marc Dymetman, and George Foster (eds.), Learning Machine Translation. MIT Press (2009)
[44] Radev, D., Jing, H. and Budzikowska, M.: Centroid-based summarization of multiple documents. In: ANLP/NAACL Workshop on Automatic Summarization (Seattle, US). ACL (2000)
[45] Radev, D., Teufel, S., Saggion, H., Lam, W., Blitzer, J., Qi, H., Celebi, A., Liu, D., Drabek, E. Evaluation Challenges in Large-scale Document Summarization. In: Proceeding of the 41st meeting of the Association for Computational Linguistics, Sapporo, Japan. ACL (2003)
[46] Sparck-Jones, K.: Automatic summarising: Factors and directions. In: Mani and Maybury [28]
[47] Steinberger, J., Ježek, K.: Text Summarization and Singular Value Decomposition. In: Lecture Notes for Computer Science 2457, pages 245–254. Springer (2004)
[48] Steinberger, J., Kabadjov, M. and Poesio, M.: Improving LSA-based Summarization with Anaphora Resolution. In: Proceedings of HLT/EMNLP'05 (Vancouver, Canada), pages. 1–8. ACL (2005)
[49] Steinberger, J., Poesio, M., Kabadjov, M. and Ježek, K.: Two uses of anaphora resolution in summarization. In: Information Processing and Management 43(6), pages 1663–1680. Elsevier (2007)
[50] Steinberger, J., Kabadjov, M., Pouliquen, B., Steinberger, R. and Poesio, M.: WB-JRC-UT's Participation in TAC 2009: Update Summarization and AESOP Tasks. In: Proceedings of TAC'09. NIST (2009)
[51] Steinberger, R., Pouliquen, B. and Ignat, C.: Using language-independent rules to achieve high multilinguality in text mining. In: François Fogelman-Soulié, Domenico Perrotta, Jakub Piskorski, and Ralf Steinberger (eds.), Mining Massive Data Sets for Security. IOS-Press (2009)
[52] Stuckardt, R.: Coreference-based summarization and question answering: a case for high precision anaphor resolution. In: International Symposium on Reference Resolution and Its Applications to Question Answering and Summarization (Venice, Italy). (2003)

[53] Turchi, M., Steinberger, J., Kabadjov, M., Steinberger, R.: Using parallel corpora for multilingual (multi-document) summarisation evaluation. In: Proceedings of CLEF-10, pages 52–63. Springer (2010)
[54] Wan, X., Li, H., Xiao, J.: Cross-language document summarization based on machine translation quality prediction. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 917–926. ACL (2010)
[55] Document understanding conference: http://duc.nist.gov/.
[56] Text analysis conference: http://www.nist.gov/tac.
[57] Choi, F. Y. Y., Wiemer-Hastings, P., & Moore, J. D. Latent semantic analysis for text segmentation. In Proceedings of EMNLP, Pittsburgh, US. ACL (2001)
[58] Berry, M. W., Dumais, S. T. & O'Brien, G. W. Using linear algebra for intelligent IR. SIAM Review, 37(4). (1995)