Measures for Term and Sentence Relevances: an Evaluation for German

Heike Bieler*, **Stefanie Dipper**†

*Institute of Linguistics, University of Potsdam
Karl-Liebknecht-Str. 24–25, D – 14476 Golm
bieler@ling.uni-potsdam.de

†Institute of Linguistics, University of Bochum
D – 44780 Bochum
dipper@linguistics.rub.de

Abstract

Terms, term relevances, and sentence relevances are concepts that figure in many NLP applications, such as Text Summarization. These concepts are implemented in various ways, though. In this paper, we want to shed light on the impact that different implementations can have on the overall performance of the systems. In particular, we examine the interplay between term definitions and sentence-scoring functions. For this, we define a gold standard that ranks sentences according to their significance and evaluate a range of relevant parameters with respect to the gold standard.

1. Introduction[1]

Many applications in computational linguistics nowadays deal with large, unrestricted text input. These systems have to face large amounts of data, which may be defective in that they include, e.g., typing errors, ungrammatical sentences, chunks in foreign languages, HTML markup, etc.

A prominent way to achieve *robust* applications is to use knowledge-poor methods, such as computing term and sentence relevances based on simple frequency counts. *Term relevance* plays an important role in applications such as Information Retrieval, Text Classification, or Text Summarization — the latter being our main concern. Term relevance measures the importance or representativeness of a term in a document with respect to this document. Terms that are highly scored for relevance should be good keywords of a document, i.e., good indicators of the text's subject. *Sentence relevances* are used in Text Summarization. They measure the importance or representativeness of sentences in a document with respect to this document. Highly-scored sentences are good candidates for summary extracts.

However, Edmundson (1969) and Kupiec et al. (1995) report that term-based summarization yield rather poor results, and that location-based features provide best results: using term-based features only, Edmundson (1969) reports an accuracy of around 36% vs. around 54% with location-based features only. Kupiec et al. (1995), who use a Bayesian classifier to identify important sentences, report accuracies of 20% (term-based features only) vs. 33% (location-based features only). At first sight, terms do not provide such good clues for summarization. However, a reason for the poor results of term-based approaches could be the rather simple ways of calculating relevances taken by Edmundson (1969) and Kupiec et al. (1995): both approaches compute term relevances as a simple function of their frequencies (terms being restricted to high-frequency or content words), and sentences are scored as a function of the individual term weights.

In this paper, we experiment with various ways of defining term relevances and term-based sentence relevances and examine their impact on automatic text summarization. Since the original data used by Edmundson (1969) and Kupiec et al. (1995) were not available to us, and we used German data in our evaluation, no direct comparison of the results is possible. However, our results, which are similar to Edmundson's results, and considerably better than the results reported by Kupiec et al. (1995), might be taken as motivation to question the often-cited superiority of location-based features over term-based features.

The paper is organized as follows. Section 2. presents the idea of using terms as indicators of significance. In Section 3., we discuss and compare various standard definitions of the notion "term" and sentence-scoring functions. In Section 4., we define a gold standard that ranks sentences according to their significance, and evaluate the alternative term definitions and scoring functions, by correlation and F-score measures. The evaluation allows us to single out the individual factors and measure their contributions to the overall performance, e.g., as part of an automatic Text Summarization system.

Similar evaluations have been performed by Orasan et al. (2004) or Cummins and O'Riordan (2005). They focus on the term-scoring function whereas we evaluate alternative term definitions and sentence-scoring functions. Moreover, since our evaluation deals with German data, the results can shed light on the performance of standard techniques—which are most often applied to and evaluated with English data—when applied to an inflectional language.

2. Background

2.1. Terms

In one of the first approaches to automatic Text Summarization, Luhn (1958) introduces the basic idea of measur-

[1] The research reported in this paper was financed by Bundesministerium für Bildung und Forschung, grant 03WKH22.
We would like to thank the anonymous reviewers for their helpful comments.

ing word significance by the frequency of word occurrence in the document. According to this proposal, frequently-occuring words are significant since they indicate that the author focuses and elaborates on a certain subject, which he refers to by these words.
The method of simple word counting faces three main problems:

(i) Inflected and derived words are treated as separate words, even though they may refer to the same subject. For instance, variants such as *house*, *houses*, or *housing* are considered different terms.

(ii) Lexically-related words, such as synonyms, hypernyms (e.g., *house*, *building*), are treated as different terms. Homographs (e.g., *Java* as island or programming language) are treated as one term.

(iii) Certain highly-frequent words, such as determiners, prepositions, conjunctions (function words), in general occur very frequently but are not indicative of a document's content.

These problems can be dealt with in different ways:

(i) Word variants: (A subset of) inflected and derived variants can be detected by preprocessing and simple "morphological" analysis. Standard preprocessing steps are *tokenization* and *normalization*, which remove non-alphanumeric characters such as punctuation marks, parentheses, or quotes, and convert all letters to lower or upper case. Luhn (1958) and Orasan et al. (2004) use pattern matching to identify related words by *common prefixes*. A more linguistically-motivated method is *stemming*, by applying, e.g., the Porter stemmer (Porter, 1980) to the normalized words, as has been done by, e.g., Neto et al. (2000) or Orasan et al. (2004) in Text Summarization.
Finally, words may be represented by a list of *ngrams* (Shannon, 1948). For instance, *house* can be represented by the 4grams *_hou, hous, ouse, use_* (where the underscores represent the leading and trailing white-spaces). In this representation, related words are encoded via the intersection of ngrams. The word *housing* shares the first two 4grams with *house*.
(ii) Lexically-(un)related words: Synonyms etc. can be identified by resources such as WordNet (Fellbaum, 1998); WordNet has been exploited for Text Summarization by, e.g., Aone et al. (1998). Word-sense disambiguation can, e.g., be achieved by training a disambiguator on manually-annotated data. However, such resources and methods go beyond knowledge-poor approaches. For reasons of robustness and efficiency, many applications ignore such lexical relations.
(iii) Highly-frequent words: Determiners, prepositions, etc. can be easily eliminated by a high-frequency cutoff (Luhn, 1958) or a list of stop words (Luhn, 1958; Edmundson, 1969). Alternatively, measures from Information Retrieval, such as TF-IDF ('term frequency $*$ inverse document frequency') can be used (Salton and McGill, 1983): the *document frequency* of a term is the number of documents in a document collection that contain that term. This measure indicates how frequently a term is used in general. Words that are frequently used both in a given document as well as in the overall collection are not discriminative of the document. In contrast, words that are rare in the collection but occur frequently in a given document are significant of that document, which is reflected by a high TF-IDF score. TF-IDF measures have been used in Text Summarization by, e.g., Neto et al. (2000), Orasan et al. (2004).

The first two of the problems addressed in this section concern the definition of the notion "term". In (i), terms are represented by surface strings; in (ii), terms correspond to semantic entities, such as WordNet concepts. (iii) introduces different ways of scoring terms.
In this paper, we focus on the evaluation of different *string-based* definitions of the notion "term" and on the computation of *sentence relevances* (see next section). For scoring term relevances, we use the TF-IDF method.

2.2. Sentence relevances

Having computed term relevances, applications like Text Summarization go one step further and compute scores of sentence relevance. Sentence relevance is usually defined as a function of the scores of the terms in the sentence.
The simplest technique is to compute sentence score as the *average* of the term scores in that sentence (Aone et al., 1998). Luhn (1958) proposes a sophisticated variant of the average approach, which promotes *clusters* of relevant terms, i.e., relevant terms that occur in close proximity to each other. Only up to four non-significant terms, with a relevance below a certain threshold, may intervene in such a cluster. Sentence relevance is then computed as a function of the best cluster in that sentence.
Finally, sentences can be scored according to their *similarity* to the document. The entire document as well as all individual sentences are represented as vectors in a multi-dimensional term space. The score of a sentence is computed as the similarity between the document's vector and the sentence's vector (Salton and McGill, 1983). This technique from Information Retrieval has been applied to Text Summarization by, e.g., Gong and Liu (2001).
Often, however, publications remain unclear as to how exactly the relevances of the individual term find their way into the sentence scores.

3. Comparative Analysis

In this section, we introduce the different definitions of the notion "term" (Sec. 3.1.) and ways of computing sentence scores (Sec. 3.3.), that we used in the evaluation. Term relevances are calculated by the TF-IDF measure (Sec. 3.2.).

3.1. Definitions of "term"

We assume that the input document is tokenized, i.e., punctuation marks have been separated from word tokens, etc. Our evaluation considers the following types of terms:

1. **wf**: wordforms, i.e., the original tokens from the tokenizer

2. **stems**: tokens analyzed by a German Porter stemmer[2]

[2] `http://snowball.tartarus.org/algorithms/german/stemmer.html`

Text sort	Domain	#Documents	#Tokens	#Types
reviews	hotel	254	165,000	19,000
reviews	film	163	138,100	26,000
news articles	general	1,370	530,000	71,000
news articles	opinion	1,488	300,000	35,000
press releases	chemistry/biology	790	250,000	35,000
speeches	politics	1,687	15,111,000	206,000

Table 1: Document collections of different text sorts and domains

3. **4gram_tok** and **5gram_tok**: sequences of 4 or 5 letters, within a token, including leading and trailing spaces
4. **4gram_sent** and **5gram_sent**: sequences of 4 or 5 letters, including spaces, within a sentence; this definition accounts for multi-word expressions.

These terms are *normalized*: all letters are converted to lower-case, the German special character "ß" is replaced by "ss", and Umlaut is replaced by base vowel + "e" (for wordforms and stems) or base vowel only (for ngrams, to keep the number of ngrams low). For instance, "Füße" ('feet') is mapped to "fuesse" for wordforms and stems. The actual 4gram sequence "Füße" is mapped to two 4gram sequences "fuss" and "usse".

3.2. Term relevances

As mentioned above, we use TF-IDF to compute term relevances. As document collections, we have selected six corpora with documents of different text sorts and domains, cf. Table 1.[3]

For calculating the term relevances of a document, the document first has to be classified according to Table 1, i.e., assigned to a specific document collection. The standard way of computing the term relevance $\mathit{TF\text{-}IDF}$ of a term t in a document d is

$$\mathit{TF\text{-}IDF}_{t,d} = \mathit{TF}_{t,d} * log(\frac{D}{\mathit{DF}_t})$$

where $\mathit{TF}_{t,d}$ denotes the frequency of t in d, D denotes the number of documents in the collection, and DF_t is the number of documents in the collection that contain t.

Unknown terms, i.e., terms that do not occur within the document collection, receive a fixed value as their document frequency. We calculate $\mathit{TF\text{-}IDF}$ of unknown terms u as

$$\mathit{TF\text{-}IDF}_{u,d} = \mathit{TF}_{u,d} * \frac{log(10)}{2}$$

which corresponds to half of the value that terms would receive which occur in 10% of the documents.

3.3. Computing sentence scores

For calculating sentence relevances, we again consider different standard methods.

1. **Average_All**: Sentence relevance SR is the average of the TF-IDF values of all terms in the sentence:

$$\mathit{SR} = \frac{\mathit{TF\text{-}IDF}_s}{T_s}$$

where $\mathit{TF\text{-}IDF}_s$ is the sum of all TF-IDF values, T_s the number of terms in the sentence.

2. **Average_All_Smooth** is a variant of Average_All which effectively boosts long sentences, by diminishing the negative impact of sentence length ($\mathit{TF\text{-}IDF}_s$, T_s as above):

$$\mathit{SR} = \frac{\mathit{TF\text{-}IDF}_s * T_s}{log(T_s)}$$

For the measures in 1. and 2., we included variants **Average_Selected** and **Average_Selected_Smooth**, respectively, which use the scores of the 40 most relevant terms only rather than all terms.[4]

3. **Luhn_Orig**: this measure implements the proposal by Luhn (1958). Sentence relevance is a function of the best-weighted (= most relevant) cluster of the sentence, where a cluster is a sequence of terms within a sentence with up to 4 intervening non-significant terms. A term is significant if it is among the 40 most relevant TF-IDF terms. The cluster relevance CR is calculated as

$$\mathit{CR} = \frac{Tsig_c^2}{T_c}$$

where $Tsig_c$ is the number of significant terms in the cluster, T_c the number of all terms in the cluster.

4. **Luhn_Weighted** is a variant of the measure Luhn_Orig. It uses the TF-IDF values of significant terms in a cluster rather than their number. CR is now calculated as

$$\mathit{CR} = \frac{\mathit{TF\text{-}IDF}_c^2}{T_c}$$

where $\mathit{TF\text{-}IDF}_c$ is the sum of the TF-IDF values of the significant terms in the cluster, T_c as above.

[3] We experimented with two ways of computing document frequencies: in one version, we took into account all terms in a document collection; in the other version, terms with low document frequencies were deleted. The threshold varied from 3 (for small corpora) to 12 (for large corpora). It turned out that this difference had no impact on performance, but the second version requires only 1/3 of the storage space used by the first version. In the paper, we ignore this parameter.

[4] Manual inspection revealed that usually the top 40 terms form a useful set of representative keywords.

5. **Similarity** is computed as cosine similarity between the vectors of the sentence and the document (i.e., the vectors encoding the TF-IDF values of the terms in the sentence/document). In contrast to the other measures, terms that occur in the document but not in the current sentence have negative impact on similarity.

To illustrate the impact of the different term definitions and sentence scoring functions on the overall relevance of a sentence, we present a short example text (see Table 2), along with selected term relevances (Table 3), and sentence rankings (Table 4) that result from the different definitions. Table 4 presents all combinations[5] of term definitions with sentence scoring functions, for sentences [1] (upper table) and [2] (lower table). [1] and [2] both contain words related to the text's topic "Ägypten" ('Egypt'), but [2] actually does not convey important information.

Table 3 shows that with all term definitions, the significance of the notion "Ägypten" for the text is recognized. Table 4 shows that, combined with the sentence scoring functions, this can result in quite diverse sentence rankings. In the following, we point out the most prominent differences that show up in the table and try to motivate them.

- Comparing the average scoring functions (1./2.) with their smoothed variants (1a./2a.), we see that the simple versions rank the text's header *Ägypten* (= [1]) first, whereas the smoothed variants rank it last (= rank 26, with terms = wf/stems) or 2nd–4th (with ngram terms).
This can be explained by the fact that the smoothed variants penalize the short, 1-term sentence, even if the term itself is high-ranked.
Ngram-based terms rank [1] on positions 2–4, because [1] consists of a *sequence* of ngram terms, and all of them are high-ranked.
- The similarity function (5.) shows that [1] has much in common with the other sentences in the document. Only wf-terms fail to establish the link between *Ägypten* in [1] and morphological variants like *Ägypter* ('Egyptian people', as in [4] and [9]) or *Ägyptens* (genitive form of 'Egypt', [6]). The ngram-terms even manage to link *Ägypten* with compounds like *altägyptische* ('ancient Egyptian', [3]).
- Comparing the average functions operating on all (1./1a.) or selected (2./2a.) terms, it turns out that sentence [2] is boosted with selected terms. With terms = wf/stems, this is due to the fact that from the total of six words, only two (*gr.*, *Erde*) figure among the 40 most relevant terms, whereas three of the words (*Aigyptos, ägypt., Quemt*) are unknown in the document collections and, hence, receive very low values (see Sec. 3.2.). Restricting the scoring function to relevant terms therefore diminishes the impact of the unknown words.
- Only ngram-based approaches successfully relate the adjective *ägypt.* (and, by chance, the Greek word *Aigyptos*) in [2] to the base noun *Ägypten*. This results in overall higher ranks for [2] with ngrams (ranging from 2–16), vs. lower ranks (7–25) with wf/stems.

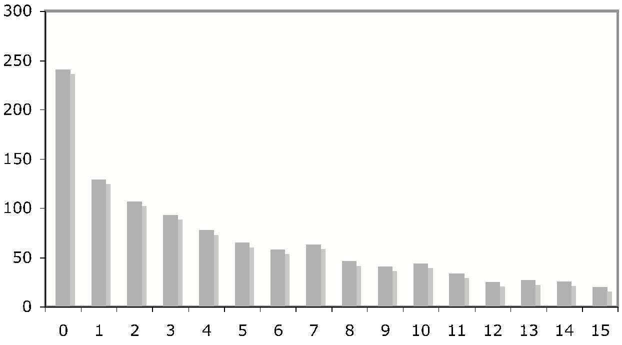

Figure 1: Distribution of the numbers of annotations per sentence (x-axis: number of annotators who marked a sentence as relevant; y-axis: number of sentences)

The example shows that the combinations of term definition and scoring function result in highly diverse rankings. The following section aims at an objective evaluation, by comparing the ranks to a gold standard.

4. Evaluation

4.1. Gold standard

As is well known, it is difficult to define a gold standard for the (intrinsic) evaluation of Information Retrieval or Text Summarization systems. We decided to use a set of 35 documents of all text sorts and domains mentioned in Table 1, with a total of 1097 sentences, each annotated by 15 human annotators. The annotation criteria instructed the annotators to mark all sentences which would be good candidates for an indicative summary, i.e., good indicators of the subject of the text. The annotators were told to include sentences with *redundant content* as well as sentences that contain *pronouns*, if these sentences contained relevant content (such sentences are usually excluded from gold standards for summarization). There were no restrictions on the absolute number of sentences to be marked.

The manual annotations quite naturally result in a ranking of sentence relevance: the more annotators to mark a sentence, the more relevant and important that sentence is.

Figure 1 shows the distribution of the numbers of annotations per sentence: Almost 1/4 of the sentences were unanimously considered "irrelevant" by all of the annotators (241 out of 1097 sentences received 0 annotations). That is, *percent agreement* of the 10% and 20% most unimportant sentences is 100%.[6] With respect to the top end of the field, i.e., the 10% most important sentences (which correspond to the set of sentences that received between 12 and 15 annotations), percent agreement is 89.5%. Overall percent agreement is 81.0%. These results are in line with Jing et al. (1998), who found that human agreement decreases as the length of summary increases.

To evaluate the automatic scoring methods described in the previous sections, there are roughly two ways: (i) to measure how well the methods perform in reproducing the ranking resulting from the manual annotations; (ii) to ignore the

[5] We did not compute Luhn scores for the ngram-based terms because the concept of clusters of consecutive highly-relevant ngrams seems not sensible, at least with synthetic languages.

[6] Percent agreement measures the ratio of observed agreements with the majority opinion to possible agreements with the majority opinion (Jing et al., 1998; Gale et al., 1992).

[1] Ägypten
['Egypt']

[2] (gr. Aigyptos; ägypt. Quemt, "schwarze Erde")
['Greek: Aigyptos, Egyptian: Quemt, "black earth" ']

[3] Eine fremdartige Welt ist die altägyptische Hochkultur mit ihren tierköpfigen Gottheiten, die in zahlreichen Abbildungen überkommen sind.
[4] Auch die fluchbeladenen Mumien üben ihre eigene Faszination aus, wie die eng mit dem Totenkult verwobene gewaltige Bautätigkeit der Ägypter.
[5] Mit den Pyramiden schufen sie die massivsten und dauerhaftesten Bauten der Erde.
[6] Betrachtet man eine Karte Ägyptens , so versetzt es in Erstaunen , wie ein Land , das fast ausschließlich von Wüste bedeckt ist , derartige Leistungen hervorbringen kann .
[7] Alles Leben in Ägypten hängt vom Nil ab , der das Land von Süden nach Norden durchstrüomt und sich in einem fruchtbaren Delta in das Mittelmeer ergießt .
[8] Zwischen der Sahara im Westen und der arabischen Wüste im Osten spielt sich , abgesehen von einigen Oasen , bis heute fast alles Leben in Ägypten in diesem Niltal ab .
[9] Die Abhängigkeit vom Nil und seinem regelmäßigen Hochwasser war prägend für die Ägypter , hing doch die Ernte und damit das Überleben von ihm ab , der mit der Nilschwemme das Niltal überflutete .
[10] Der sich ablagernder Schlamm des Flusses war es , der dem Land die Nährstoffe brachte , sein Wasser ermöglichte das Leben .

Table 2: The first 10 (of 26) sentences from a text about Egypt. Source: `http://www.sungaya.de/schwarz/aegypter/aegypter.htm`, accessed Mar. 19 2008.

Rank	wf	stems	4gram_tok	4gram_sent	5gram_tok	5gram_sent
1	aegypten	aegypt	gypt	gypt	agypt	agypt
2	chr.	chr.	agyp	agyp	_agyp	gypte
3	niltal	niltal	_agy	ypte	gypte	_agyp
4	nil	nil	ypte	_agy	ypten	ypten
5	aegypter	altaegypt	_nil	_nil	pten_	e_agy
6	atlantis	hochkultur	pten	pten	_chr.	_chr.

Table 3: The 6 top-ranked terms according to different term definitions

			wf	stems	4g_tok	4g_sent	5g_tok	5g_sent
[1]	1.	Average_All	1	1	1	1	1	1
	2.	Average_Selected	1	1	1	1	1	1
	1a.	Average_All_Smooth	26	26	4	3	4	3
	2a.	Average_Selected_Smooth	26	26	2	2	2	2
	3.	Luhn_Orig	25	23	-	-	-	-
	4.	Luhn_Weighted	7	13	-	-	-	-
	5.	Similarity	7	1	1	1	1	1
[2]	1.	Average_All	11	17	2	2	2	2
	2.	Average_Selected	7	12	2	2	2	2
	1a.	Average_All_Smooth	24	24	11	15	11	15
	2a.	Average_Selected_Smooth	13	17	7	11	7	11
	3.	Luhn_Orig	24	22	-	-	-	-
	4.	Luhn_Weighted	20	20	-	-	-	-
	5.	Similarity	25	24	15	16	15	16

Table 4: Sentence rankings for sentences [1] (upper part) and [2] (lower part), according to different term definitions (columns) and sentence-scoring functions (rows)

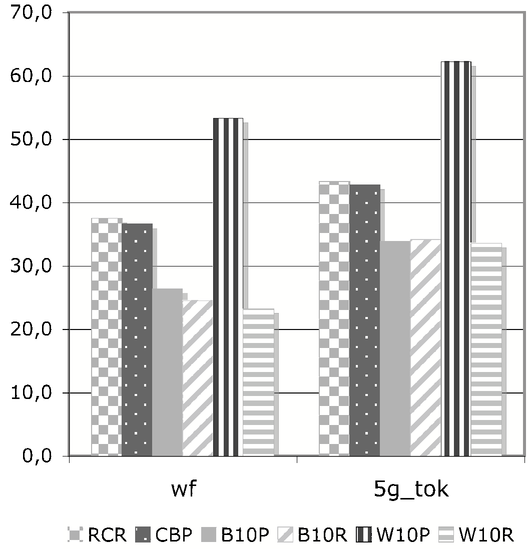

Figure 2: Results for the sentence-scoring function Similarity, with terms = wf/5g_tok, according to different evaluation methods

exact ranking and, instead, to measure how well the methods overlap with the manual annotations in the top-most and bottom-most set of sentences, thus taking into account the findings of Jing et al. (1998).

In our evaluation, we persued both ways, by assuming the following evaluation measures:

1. **Rank_Correlation_Rho (RCR)**: Spearman's rank correlation coefficient. The sentences of a document are ranked (i) according to the sentence relevances computed by the automatic methods, and (ii) according to the number of manual annotations. The order of both ranks is compared.
2. **Correlation_Bravais_Pearson (CBP)**: this measure compares the normalized *absolute* values of sentence relevances and number of manual annotations rather than the relative orders.[7]
3. **Best_10 (B10), Worst_10 (W10)**: recall, precision and F-score of the first and last 10% of the sentences.[8] The F-score of the Best_10 measure is the measure that corresponds most closely to the accuracy measures used by Edmundson (1969) and Kupiec et al. (1995).

4.2. Results

We first focus on the different evaluation methods and then present the overall results. Figure 2 displays the results for the sentence-scoring function Similarity, with wordforms (left side) and 5gram_tok terms (right side). The individual columns encode the results for the different evaluation measures. We see that 5g_tok terms perform clearly better than wordforms, with all measures.

The figure also shows that the two correlation measures (RCR and CBP) assign similar values to the term/scoring combinations. Looking at the Best_10 measure, we see that it assigns very low values: precision (B10P) is 26.4 for wordforms and 33.9 for 5g_tok; recall (B10R) is 24.6 for wordforms and 34.2 for 5g_tok. The values of Worst_10 precision (W10P) are considerably higher (53.3 and 62.3) but diverge a lot from recall (W10R: 23.2 and 33.6).[9]

Table 5 displays the results for all combinations of term definitions and scoring functions. In this table, the values of correlation measures are averaged, and F-score is reported, rather than precision and recall.

For the different term variants, our evaluation clearly shows that 5g_tok perform best and wordforms perform worst, with the vast majority of scoring functions. Only with Average_Selected/F-score of B10, and Average_Selected_Smooth/F-score of W10, 5g_tok does not come out on top. In general, 4g_tok scores similarly well as 5g_tok.

As for the scoring functions, the picture is more complex: the function Average_All performs worst and Average_Selected next to it, at least in most of the cases. For the task of selecting low-ranked sentences (W10), Average_Selected performs best, though. In general, Average_All_Smooth and Similarity turn out best, see the highlighted figures in Table 5.

If one wants to compare our results with the results by Edmundson (1969) and Kupiec et al. (1995), one has to consider our highest Best_10 F-Score (FB), which is around 34% (with 5g_tok, combined with different sentence scoring functions). This result is similar to Edmundson's result (36%) and considerably better than the result reported by Kupiec et al. (1995) (20%).

5. Conclusion

In this paper, we compared and evaluated alternative definitions of "term" and different ways of computing sentence relevances. We illustrated the diversity of sentence rankings that result from the combinations of term definitions and sentence scoring functions. Our evaluation with respect to a gold standard showed that the token-based 5grams yield the best results among the term alternatives. This could be attributed to the fact that ngram-based terms cope better with inflecting languages like German, the object language of our study. For the sentence-scoring functions, Average_All_Smooth and Similarity performed best, with the evaluation methods of correlation and F-score of best 10%, respectively. The values achieved by these combinations are rather low, though. On the one hand, this seems to confirm the much-cited results by Edmundson (1969) and Kupiec et al. (1995); on the other hand, the results obviously depend heavily on the way the relevances are computed and it is therefore possible to outperform the results by Kupiec et al. (1995).

[7] A value of 0 for the correlation ranks means that there is no correlation at all between the gold standard and the automatic methods; a value of 100 encodes complete correlation; -100 indicates reversed order.

[8] The 10%-portions are determined on the basis of the *entire* corpus, i.e., these portions contain the 10% sentences with highest/lowest relevance *across* the corpus, as assigned by the annotators or automatic methods. As a consequence, the 10% portions of some texts may be empty (if they did not receive enough annotations).

[9] The low recall of W10R can be attributed to the fact that a large number of sentences (almost 1/4) were not marked by any annotator, whereas most of the automatic methods assign *some* value to all of the sentences.

	LuhnOrig			LuhnWeighted			Similarity			Average_All			Average_Select			Av_All_Smooth			Av_Sel_Smooth		
	C	FB	FW	C	FB	FW	C	FB	FW	C	FB	FW	C	FB	FW	C	FB	FW	C	FB	FW
wf	28,0	27,7	37,2	30,3	29,0	37,2	37,1	25,5	32,4	20,3	23,1	26,9	19,9	21,1	37,1	37,1	25,6	33,7	31,3	21,1	38,3
stems	30,3	26,6	38,6	31,9	26,0	38,1	39,0	24,7	37,2	22,8	24,1	29,8	22,8	24,0	38,6	38,3	30,0	35,9	34,4	30,0	33,5
4g_tok							43,0	31,2	42,9	28,2	27,9	34,4	29,9	27,8	45,1	42,7	32,1	41,0	36,7	31,9	44,8
4g_sent							41,2	27,7	39,4	24,9	25,8	32,8	28,6	25,7	43,8	39,7	25,2	38,8	35,3	26,3	43,5
5g_tok							***43,1***	***34,1***	43,7	29,4	28,9	37,0	30,0	27,1	**46,7**	**44,3**	**34,2**	43,1	38,5	**34,2**	36,7
5g_sent							39,9	26,8	38,7	24,8	25,2	31,4	28,0	26,7	***45,3***	39,4	27,4	38,8	36,0	29,9	45,1

Table 5: Results for all term definitions and sentence-scoring functions. The values are the average values of the correlation measures (C), and F-scores for B10 (FB) and W10 (FW). For each of C, FB, FW, the best combination of term definition/scoring function is printed bold-face, the second-best bold-face and italics.

6. References

Chinatsu Aone, Mary Ellen Okurowski, and James Gorlinsky. 1998. Trainable, scalable summarization using robust NLP and machine learning. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics (COLING-ACL)*, pages 62–66.

Ronan Cummins and Colm O'Riordan. 2005. Evolving general term-weighting schemes for information retrieval: Tests on larger collections. *Artificial Intelligence Review*, 24(3-4):277–299, November.

H.P. Edmundson. 1969. New methods in automatic extracting. *Journal of the ACM*, 16(2):264–285.

Christiane Fellbaum, editor. 1998. *WordNet: An electronic lexical database*. MIT Press.

William A. Gale, Kenneth W. Church, and David Yarowsky. 1992. Estimating upper and lower bounds on the performance of word-sense disambiguation programs. In *Proceedings of the 30th Annual Meeting of the ACL*, pages 249–256.

Yihong Gong and Xin Liu. 2001. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th annual international ACM SIGIR conference on research and development in Information Retrieval*, pages 19–25.

Hongyan Jing, Regina Barzilay, Kathleen McKeown, and Michael Elhadad. 1998. Summarization evaluation methods experiments and analysis. In *Proceedings of the AAAI Intelligent Text Summarization Workshop*, pages 60–68.

Julian Kupiec, Jan O. Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *roceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 68–73.

Hans Peter Luhn. 1958. The automatic creation of literature abstracts. *IBM Journal of Research & Development*, 2(2):159–165.

Joel Larocca Neto, Alexandre D. Santos, Celso A.A. Kaestner, and Alex A. Freitas. 2000. Document clustering and text summarization. In *Proceedings of the 4th Int. Conference on Practical Applications of Knowledge Discovery and Data Mining (PADD-2000)*, pages 41–55.

Constantin Orasan, Viktor Pekar, and Laura Hasler. 2004. A comparison of summarisation methods based on term specificity estimation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC-04)*, pages 1037–1041.

Martin F. Porter. 1980. An algorithm for suffix stripping. *Program*, 14:130–137. Reprinted in: K. Sparck Jones and P. Willet (Eds.) *Readings in Information Retrieval*, 313–316. 1997.

Gerard Salton and Michael J. McGill. 1983. *Introduction to Modern Information Retrieval*. New York: McGraw-Hill.

Claude E Shannon. 1948. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423 and 623–656.