Towards multi-lingual summarization: A comparative analysis of sentence extraction methods on English and Hebrew corpora

Marina Litvak and **Hagay Lipman** and **Assaf Ben Gur** and **Mark Last**
Ben Gurion University of the Negev
{litvakm, lipmanh, bengura, mlast}@bgu.ac.il

Slava Kisilevich and **Daniel Keim**
University of Konstanz
slaks@dbvis.inf.uni-konstanz.de
Daniel.Keim@uni-konstanz.de

Abstract

The trend toward the growing multilinguality of the Internet requires text summarization techniques that work equally well in multiple languages. Only some of the automated summarization methods proposed in the literature, however, can be defined as "language-independent", as they are not based on any morphological analysis of the summarized text. In this paper, we perform an in-depth comparative analysis of language-independent sentence scoring methods for extractive single-document summarization. We evaluate 15 published summarization methods proposed in the literature and 16 methods introduced in (Litvak et al., 2010). The evaluation is performed on English and Hebrew corpora. The results suggest that the performance ranking of the compared methods is quite similar in both languages. The top ten bilingual scoring methods include six methods introduced in (Litvak et al., 2010).

1 Introduction

Automatically generated summaries can significantly reduce the information overload on professionals in a variety of fields, could prove beneficial for the automated classification and filtering of documents, the search for information over the Internet and applications that utilize large textual databases.

Document summarization methodologies include *statistic*-based, using either the classic vector space model or a graph representation, and *semantic*-based, using ontologies and language-specific knowledge (Mani & Maybury, 1999). Although the use of language-specific knowledge can potentially improve the quality of automated summaries generated in a particular language, its language specificity ultimately restricts the use of such a summarizer to a single language. Only systems that perform equally well on different languages in the absence of any language-specific knowledge can be considered language-independent summarizers.

As the number of languages used on the Internet increases continiously (there are at least 75 different languages according to a estimate performed by A. Gulli and A. Signorini[1] in the end of January 2005), there is a growing need for language-independent statistical summarization techniques that can be readily applied to text in any language without using language-specific morphological tools.

In this work, we perform an in-depth comparative analysis of 16 methods for language-independent extractive summarization introduced in (Litvak et al., 2010) that utilize either vector or graph-based representations of text documents computed from word segmentation and 15 state-of-the art language-independent scoring methods. The main goal of the evaluation experiments, which focused on English and Hebrew corpora, is to find the most efficient language-independent sentence scoring methods

[1] http://www.cs.uiowa.edu/ asignori/web-size/

Proceedings of the 4th International Workshop on Cross Lingual Information Access at COLING 2010, pages 61–69,
Beijing, August 2010

in terms of summarization accuracy and computational complexity across two different languages.

This paper is organized as follows. The next section describes related work in extractive summarization. Section 3 reviews the evaluated language-independent sentence scoring approaches. Section 4 contains our experimental results on English and Hebrew corpora. The last section comprises conclusions and future work.

2 Related Work

Extractive summarization is aimed at the selection of a subset of the most relevant fragments, which can be paragraphs, sentences, keyphrases, or keywords from a given source text. The extractive summarization process usually involves ranking, such that each fragment of a summarized text gets a relevance score, and extraction, during which the top-ranked fragments are extracted and arranged in a summary in the same order they appeared in the original text. Statistical methods for calculating the relevance score of each fragment can rely on such information as: fragment *position* inside the document, its *length*, whether it contains *keywords* or *title* words.

Research by Luhn (1958), in which the significance factor of a sentence is based on the frequency and the relative position of significant words within that sentence, is considered the first on automated text summarization. Luhn's work was followed shortly thereafter by that of Edmundson (1969) and some time later by studies from Radev et al. (2001) and Saggion et al. (2003), all of who applied linear combinations of multiple statistical methods to rank sentences using the vector space model as a text representation. In (Litvak et al., 2010) we improve the summarization quality by identifying the best linear combination of the metrics evaluated in this paper.

Several information retrieval and machine learning techniques have been proposed for determining sentence importance (Kupiec et al., 1995; Wong et al., 2008). Gong and Liu (2001) and Steinberger and Jezek (2004) showed that singular value decomposition (SVD) can be applied to generate extracts.

Among text representation models, graph-based text representations have gained popularity in automated summarization, as they enable the model to be enriched with syntactic and semantic relations. Salton et al. (1997) were among the first to attempt graph-based ranking methods for single document extractive summarization by generating similarity links between document paragraphs. The important paragraphs of a text were extracted using degree scores. Erkan and Radev (2004) and Mihalcea (2005) introduced approaches for unsupervised extractive summarization that rely on the application of iterative graph based ranking algorithms. In their approaches, each document is represented as a graph of sentences interconnected by similarity relations.

3 Language-Independent Scoring Methods for Sentence Extraction

Various language dependent and independent sentence scoring methods have been introduced in the literature. We selected the 15 most prominent language independent methods for evaluation. Most of them can be categorized as *frequency*, *position*, *length*, or *title*-based, and they utilize vector representation. *TextRank (ML_TR)* is the only method that is based on graph representation, but there are also *position* and *length*-based methods that calculate scores using the overall structure of a document. We have also considered 16 methods proposed in (Litvak et al., 2010), including 13 based on the *graph-theoretic* representation (Section 3.1).

Figure 1 (Litvak et al., 2010) shows the taxonomy of the 31 methods considered in our work. All methods introduced in (Litvak et al., 2010) are denoted by an asterisk (*). Methods requiring a threshold value $t \in [0, 1]$ that specifies the portion of the top rated terms considered significant are marked by a cross in Figure 1 and listed in Table 1 along with the optimal average threshold values obtained after evaluating the methods

Table 1: Selected thresholds for threshold-based scoring methods

Method	Threshold
LUHN	0.9
LUHN_DEG	0.9
LUHN_PR	0.0
KEY	[0.8, 1.0]
KEY_DEG	[0.8, 1.0]
KEY_PR	[0.1, 1.0]
COV	0.9
COV_DEG	[0.7, 0.9]
COV_PR	0.1

on English and Hebrew documents (Litvak et al., 2010).

The methods are divided into three main categories: *structure*-, *vector*-, and *graph*-based methods, and each category also contains an internal taxonomy. Sections 3.2, 3.3, and 3.4 present structure-, *vector*-, and *graph*-based methods, respectively. With each description, a reference to the original work where the method was proposed for extractive summarization is included. We denote sentence by S and text document by D.

3.1 Text Representation Models

The vector-based scoring methods listed below use *tf* or *tf-idf* term weights to evaluate sentence importance while that used by the graph-based methods (except for TextRank) is based on the word-based graph representation model presented in Schenker et al. (2004). We represent each document by a directed, labeled, unweighted graph in which nodes represent unique terms (distinct normalized words) and edges represent order-relationships between two terms. Each edge is labeled with the IDs of sentences that contain both words in the specified order.

3.2 Structure-based Scoring Methods

In this section, we describe the existing structure-based methods for multilingual sentence scoring. These methods do not require any text representation and are based on its structure.

– *Position* (Baxendale, 1958):
POS_L Closeness to the end of the document: $score(S_i) = i$, where i is a sequential number of a sentence in a document;
POS_F Closeness to the beginning of the document: $score(S_i) = \frac{1}{i}$;
POS_B Closeness to the borders of the document: $score(S_i) = max(\frac{1}{i}, \frac{1}{n-i+1})$, where n is the total number of sentences in D.
– *Length* (Satoshi et al., 2001):
LEN_W Number of *words* in a sentence;
LEN_CH Number of *characters* in a sentence.

3.3 Vector-based Scoring Methods

In this section, we describe the vector-based methods for multilingual sentence scoring, that are based on the vector space model for text representation.

– *Frequency*-based:
LUHN (Luhn, 1958)
$score(S) = max_{c_i \in \{clusters(S)\}}\{cs_i\}$, where clusters are portions of a sentence bracketed by keywords[2] and $cs_i = \frac{|keywords(c_i)|^2}{|c_i|}$.
KEY (Edmundson, 1969) Sum of the keyword frequencies: $score(S) = \sum_{i \in \{keywords(S)\}} tf_i$, where tf_i is term in-document frequency of keyword i.
COV (Kallel et al., 2004) Ratio of keyword numbers (Coverage): $score(S) = \frac{|keywords(S)|}{|keywords(D)|}$
TF (Vanderwende et al., 2007) Average term frequency for all sentence words:
$score(S) = \frac{\sum_{i \in \{words(S)\}} tf_i}{|S|}$.
TFISF (Neto et al., 2000) Average term frequency inverted sentence frequency for all sentence words: $score(S) = \sum_{i \in \{words(S)\}} tf_i \times isf_i$,
where $isf_i = 1 - \frac{log(n_i)}{log(n)}$, where n is the number of sentences in a document and n_i is the number of sentences containing word i.
SVD (Steinberger & Jezek, 2004) $score(S)$ is equal to the length of a sentence vector in $\Sigma^2 V^T$ after computing the Singular Value Decomposition of a term by sentence matrix $A = U \Sigma V^T$
– *Title* (Edmundson, 1969) similarity[3] to the title, $score(S) = sim(S, T)$:
TITLE_O using overlap similarity: $\frac{|S \cap T|}{min\{|S|,|T|\}}$
TITLE_J using Jaccard similarity: $\frac{|S \cap T|}{|S \cup T|}$

[2] Luhn's experiments suggest an optimal limit of 4 or 5 non-significant words between keywords.

[3] Due to multilingual focus of our work, *exact* word matching was used in all similarity-based methods.

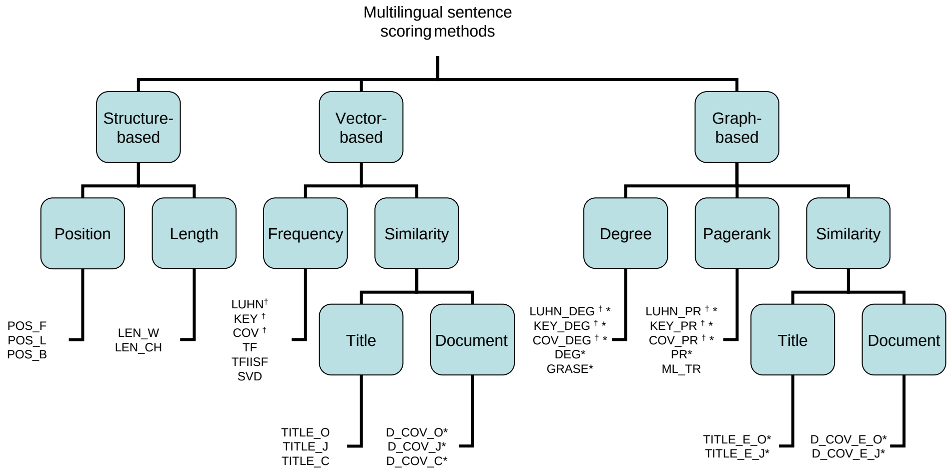

Figure 1: Taxonomy of statistical language-independent sentence scoring methods (Litvak et al., 2010)

TITLE_C using cosine similarity:
$sim(\vec{S},\vec{T}) = cos(\vec{S},\vec{T}) = \frac{\vec{S}\times\vec{T}}{|\vec{S}|\times|\vec{T}|}$

– *Document Coverage* (Litvak et al., 2010). These methods score a sentence according to its similarity to the rest of the sentences in the document ($D - S$) based on the following intuition: the more document content is covered by a sentence, the more important the sentence is to a summary. Redundant sentences containing repetitive information are removed using a similarity filter. $score(S) = sim(S, D - S)$:

D_COV_O using Overlap similarity: $\frac{|S \cap T|}{min\{|S|,|D-S|\}}$

D_COV_J using Jaccard similarity: $\frac{|S \cap T|}{|S \cup D-S|}$

D_COV_C using Cosine similarity:
$cos(\vec{S}, D \vec{-} S) = \frac{\vec{S}\times D\vec{-}S}{|\vec{S}|\times|D\vec{-}S|}$

3.4 Graph-based Scoring Methods

In this section, we describe the methods for multilingual sentence scoring using the graph text representation based on sentence (ML_TR) or word (all except ML_TR) segmentation.

ML_TR Multilingual version of TextRank (Mihalcea, 2005) without morphological analysis. Each document is represented as a directed graph of nodes that stand for sentences interconnected by similarity (*overlap*) relationship. To each edge connecting two vertices the weight is assigned and equal to the similarity value between the corresponding sentences. We used backward links, as it was the most successful according to the reported results in (Mihalcea, 2005). $score(S)$ is equal to PageRank (Brin & Page, 1998) of its node, according to the formula adapted to the weights assigned to edges.

– *Degree*-based (Litvak et al., 2010):[4]

LUHN_DEG A graph-based extension of the LUHN measure, in which a node degree is used instead of a word frequency: words are considered significant if they are represented by nodes of a higher degree than a predefined threshold (see Table 1).

KEY_DEG Graph-based extension of KEY measure.

COV_DEG Graph-based extension of COV measure.

DEG Average degree for all sentence nodes: $score(S) = \frac{\sum_{i \in \{words(S)\}} Deg_i}{|S|}$.

GRASE(GRaph-based Automated Sentence Extractor) Modification of Salton's algorithm (Salton et al., 1997) using the graph

[4] All proposed here degree-based methods, except for GRASE, use undirected graphs and degree of nodes as a predictive feature. The methods based on the directed word graphs and distinguishing between in- and out-links were outperformed in our preliminary experiments by the undirected approach.

representation defined in Section 3.1 above. In our graph representation, all sentences are represented by paths, completely or partially. To identify the relevant sentences, we search for the *bushy* paths and extract from them the sentences that appear the most frequently. Each sentence in the *bushy* path gets a domination score that is the number of edges with its label in the path normalized by the sentence length. The relevance score for a sentence is calculated as a sum of its domination scores over all paths.

– *PageRank*-based:[5]

LUHN_PR A graph-based extension of the LUHN measure in which the node PageRank value is used instead of the word frequency: keywords are those words represented by nodes with a PageRank score higher than a predefined threshold (see Table 1).

KEY_PR Graph-based extension of KEY measure.

COV_PR Graph-based extension of COV measure.

PR Average PageRank for all sentence nodes: $score(S) = \frac{\sum_{i \in \{words(S)\}} PR_i}{|S|}$.

– *Similarity*-based. Edge matching techniques similar to those of Nastase and Szpakowicz (2006) are used. Edge matching is an alternative approach to measure the similarity between graphs based on the number of common edges:

TITLE_E_O Graph-based extension of TITLE_O – Overlap-based edge matching between title and sentence graphs.

TITLE_E_J Graph-based extension of TITLE_J – Jaccard-based edge matching between title and sentence graphs.

D_COV_E_O Graph-based extension of D_COV_O – Overlap-based edge matching between sentence and document complement (the rest of a document sentences) graphs.

D_COV_E_J Graph-based extension of D_COV_J – Jaccard-based edge matching between sentence and document complement graphs.

4 Experiments

4.1 Overview

The quality of the above-mentioned sentence ranking methods was evaluated through a comparative experiment on corpora of English and Hebrew texts. These two languages, which belong to different language families (Indo-European and Semitic languages, respectively), were intentionally chosen for this experiment to increase the generality of our evaluation. The main difference between these languages, is that Hebrew morphology allows morphemes to be combined systematically into complex word-forms. In different contexts, the same morpheme can appear as a separate word-form, while in others it appears agglutinated as a suffix or prefix to another word-form (Adler, 2009).

The goals of the experiment were as follows:
- To evaluate the performance of different approaches for extractive single-document summarization using graph and vector representations.
- To compare the quality of the multilingual summarization methods proposed in our previous work (Litvak et al., 2010) to the state-of-the-art approaches.
- To identify sentence ranking methods that work equally well on both languages.

4.2 Text Preprocessing

Extractive summarization relies critically on proper sentence segmentation to insure the quality of the summarization results. We used a sentence splitter provided with the MEAD summarizer (Radev et al., 2001) for English and a simple splitter for Hebrew splitting the text at every period, exclamation point, or question mark.[6]

4.3 Experimental Data

For English texts, we used the corpus of summarized documents provided for the single doc-

[5] Using undirected word graphs with PageRank does not make sense, since for an undirected graph a node pagerank score is known to be proportional to its degree. Reversing links will result in hub scores instead authority. The methods distinguishing between authority and hub scores were outperformed in our preliminary experiments by the degree-based approach.

[6] Although the same set of splitting rules may be used for both languages, separate splitters were used since the MEAD splitter is restricted to European languages.

ument summarization task at the Document Understanding Conference 2002 (DUC, 2002). This benchmark dataset contains 533 news articles, each of which is at least ten sentences long and has two to three human-generated abstracts of approximately 100 words apiece.

However, to the best of our knowledge, no summarization benchmarks exist for the Hebrew language texts. To collect summarized texts in Hebrew, we set up an experiment[7] in which 50 news articles of 250 to 830 words each from the *Haaretz*[8] newspaper internet site were summarized by human assessors by extracting the most salient sentences. In total, 70 undergraduate students from the Department of Information Systems Engineering, Ben Gurion University of the Negev participated in the experiment. Ten documents were randomly assigned to each of the 70 study participants who were instructed (1) To dedicate at least five minutes to each document, (2) To ignore dialogs and citations, (3) To read the whole document before starting sentence extraction, (4) To ignore redundant, repetitive, or overly detailed information, (5) To obey the minimal and maximal summary constraints of 95 and 100 words, respectively. Summaries were assessed for quality by procedure described in (Litvak et al., 2010).

4.4 Experimental Results

We evaluated English and Hebrew summaries using the ROUGE-$1, 2, 3, 4, L, SU$ and W metrics[9], described in Lin (2004). Our results were not statistically distinguishable and matched the conclusion of Lin (2004). However, because ROUGE-1 showed the largest variation across the methods, all results in the following comparisons are presented in terms of ROUGE-1 metric. Similar to the approach described in Dang (2006), we performed multiple comparisons between the sentence scoring methods. The Friedman test was used to reject the null hypothesis (all methods perform the same) at the 0.0001 significance level, after which we ran the Bonferroni-Dunn test (Demsar, 2006) for pairwise comparisons. Tables 2 and 3 show the results of multiple comparisons and are arranged in descending order with the best approaches on top. Methods not sharing any common letter were significantly different at the 95% confidence level.

Table 2: English: Multiple comparisons of sentence ranking approaches using the Bonferroni-Dunn test of ROUGE-1 Recall

Approach	ROUGE-1													
COV_DEG*	0.436	A												
KEY_DEG*	0.433	A	B											
KEY	0.429	A	B	C										
COV_PR*	0.428	A	B	C	D									
COV	0.428	A	B	C	D									
D_COV_C*	0.428	A	B	C	D									
D_COV_J*	0.425		B	C	D	E								
KEY_PR*	0.424		B	C	D	E								
LUHN_DEG*	0.422			C	D	E	F							
POS_F	0.419					E	F	G						
LEN_CH	0.418			C	D	E	F	G						
LUHN	0.418				D	E	F	G						
LUHN_PR*	0.418					E	F	G	H					
LEN_W	0.416				D	E	F	G	H					
ML_TR	0.414					E	F	G	H					
TITLE_E_J*	0.413						F	G	H	I				
TITLE_E_O*	0.413						F	G	H	I				
D_COV_E_J*	0.410						F	G	H	I				
D_COV_O*	0.405							G	H	I	J			
TFISF	0.405							G	H	I	J			
DEG*	0.403							G	H	I	J			
D_COV_E_O*	0.401								H	I	J	K		
PR*	0.400							G	H	I	J	K		
TITLE_J	0.399									I	J	K		
TF	0.397									I	J	K		
TITLE_O	0.396										J	K		
SVD	0.395									I	J	K		
TITLE_C	0.395										J	K		
POS_B	0.392											K	L	
GRASE*	0.372												L	
POS_L	0.339													M

The Pearson correlation between methods ranking in English and Hebrew was 0.775, which was larger than zero at a significance level of 0.0001. In other words, most of the methods were ranked in nearly the same relative positions in both corpora, and the top ranked methods performed equally well in both languages. The differences in ranking were caused by morphological differences between two languages.

To determine which approaches performed best in both languages, we analyzed the clustering results of the methods in both corpora and found the intersection of the top clusters from the two clustering results. For each language, a document-method matrix of ROUGE scores was created with methods represented by vectors of their ROUGE scores for each document in a corpora. Since most scores are not normally

[7] The software enabling easy selection and storage of sentences to be included in the document extract, can be provided upon request.

[8] http://www.haaretz.co.il

[9] ROUGE toolkit was adapted to Hebrew by specifying “token” using Hebrew alphabet

Table 3: Hebrew: Multiple comparisons of sentence ranking approaches using the Bonferroni-Dunn test of ROUGE-1 Recall

Approach	ROUGE-1						
D_COV_J*	0.574	A					
KEY	0.570	A	B				
COV_DEG*	0.568	A	B				
POS_F	0.567	A	B				
COV	0.567	A	B				
TITLE_J	0.567	A	B				
POS_B	0.565	A	B				
LUHN_PR*	0.560	A	B	C			
LUHN_DEG*	0.560	A	B	C			
D_COV_E_J*	0.559	A	B	C			
LUHN	0.559	A	B	C			
TITLE_E_J*	0.556	A	B	C			
TITLE_E_O*	0.556	A	B	C			
KEY_DEG*	0.555	A	B	C			
LEN_W	0.555	A	B	C			
LEN_CH	0.553	A	B	C			
KEY_PR*	0.546	A	B	C			
COV_PR*	0.546	A	B	C			
TITLE_O	0.545	A	B	C			
D_COV_C*	0.543	A	B	C			
TITLE_C	0.541	A	B	C			
ML_TR	0.519	A	B	C	D		
TFISF	0.514	A	B	C	D		
D_COV_E_O*	0.498	A	B	C	D		
SVD	0.498	A	B	C	D		
D_COV_O*	0.466		B	C	D		
TF	0.427			C	D	E	
DEG*	0.399				D	E	F
PR*	0.331					E	F
GRASE*	0.243						F
POS_L	0.237						F

Table 4: English: Correlation between sentence ranking approaches using Pearson

Approach	Correlated With
POS_F	(LUHN_PR, 0.973), (TITLE_E_J, 0.902), (TITLE_E_O, 0.902)
TITLE_O	(TITLE_J, 0.950)
LEN_W	(LEN_CH, 0.909)
KEY_PR	(COV_PR, 0.944)
TITLE_E_O	(TITLE_E_J, 0.997)

Table 5: Hebrew: Correlation between sentence ranking approaches using Pearson

Approach	Correlated With
KEY	(KEY_DEG, 0.930)
COV	(D_COV_J, 0.911)
POS_F	(POS_B, 0.945), (LUHN_DEG, 0.959), (LUHN_PR, 0.958)
POS_B	(LUHN_DEG, 0.927), (LUHN_PR, 0.925)
TITLE_O	(TITLE_E_J, 0.920), (TITLE_E_O, 0.920)
TITLE_J	(TITLE_E_J, 0.942), (TITLE_E_O, 0.942)
LEN_W	(LEN_CH, 0.954), (KEY_PR, 0.912)
LEN_CH	(KEY_PR, 0.936), (KEY_DEG, 0.915), (COV_DEG, 0.901)
LUHN_DEG	(LUHN_PR, 0.998)
KEY_DEG	(COV_DEG, 0.904)

Table 6: Ranking of the best bilingual scores

Scoring method	Rank in English corpus	Rank in Hebrew corpus	Text Representation
KEY	3	2	vector
COV	4	4	vector
KEY_DEG	2	10	graph
COV_DEG	1	3	graph
KEY_PR	6	12	graph
COV_PR	4	12	graph
D_COV_C	4	14	vector
D_COV_J	5	1	vector
LEN_W	10	10	structure
LEN_CH	9	11	structure

distributed, we chose the K-means algorithm, which does not assume normal distribution of data, for clustering. We ran the algorithm with different numbers of clusters ($2 \leq K \leq 10$), and for each K, we measured two parameters: the minimal distance between neighboring clusters in the clustered data for each language and the level of similarity between the clustering results for the two languages. For both parameters, we used the regular Euclidean distance. For $K \geq 6$, the clusters were highly similar for each language, and the distance between English and Hebrew clustering data was maximal. Based on the obtained results, we left results only for $2 \leq K \leq 5$ for each corpus. Then, we ordered the clusters by the average ROUGE score of each cluster's instances (methods) and identified the methods appearing in the top clusters for all K values in both corpora. Table 6 shows the resulting top ten scoring methods with their rank in each corpus. Six methods introduced in this paper, such as *Document Coverage* (*D_COV_C/J*) and graph adaptations of *Coverage* (*COV_DEG/PR*) and *Key* (*KEY_DEG/PR*), are among these top ten bilingual methods.

Neither *vector*- nor *graph*-based text representation models, however, can claim ultimate superiority, as methods based on both models prominently in the top-evaluated cluster. Moreover, highly-correlated methods (see Tables 4 and 5 for highly-correlated pairs of methods in English and Hebrew corpora, respectively) appear in the same cluster in most cases. As a result, some pairs from among the top ten methods are highly-correlated in at least one language, and only one from each pair can be considered. For example, *LEN_W* and *LEN_CH* have high correlation coefficients (0.909 and 0.954 in English and Hebrew, respectively). Since *LEN_CH* is more appropriate for multilingual processing due to variations in the rules of tokenization between languages (e.g., English vs. German), it may be considered a preferable multilingual metric.

In terms of summarization quality and computational complexity, all scoring functions presented in Table 6 can be considered to perform equally well for bilingual extractive summarization. Assuming their efficient implementation, all methods have a linear computational complexity, $O(n)$, relative to the total number of words in a document. *KEY_PR* and *COV_PR* require additional $O(c(|E|+|V|))$ time for running PageRank, where c is the number of iterations it needs to converge, $|E|$ is the number of edges, and $|V|$ is the number of nodes (distinct words) in a document graph. Since neither $|E|$ nor $|V|$ in our graph representation can be as large as n, the total computation time for *KEY_PR* and *COV_PR* metrics is also linear relative to the document size.

In terms of implementation complexity, *LEN_W* and *LEN_CH* are simpliest, since they even do not require any preprocessing and representation building; *KEY* and *COV* require keywords identification; *D_COV_C,* and *D_COV_J* require vector space model building; *KEY_DEG* and *COV_DEG* need graphs building (order of words); whereas *KEY_PR* and *COV_PR*, in addition, require PageRank implementation.

5 Conclusion and Future Research

In this paper, we conducted in-depth, comparative evaluations of 31 existing (16 of which are mostly graph-based modifications of existing state-of-the-art methods, introduced in (Litvak et al., 2010)) scoring methods[10] using English and Hebrew language texts.

The experimental results suggest that the relative ranking of methods performance is quite similar in both languages. We identified methods that performed significantly better in only one of the languages and those that performed equally well in both languages. Moreover, although vector and graph-based approaches were among the top ranked methods for bilingual application, no text representation model presented itself as markedly superior to the other.

Our future research will extend the evaluations of language-independent sentence ranking metrics to a range of other languages such as German, Arabic, Greek, and Russian. We will adapt similarity-based metrics to multilingual application by implementing them via n-gram matching instead of exact word matching. We will further improve the summarization quality by applying machine learning on described features. We will use additional techniques for summary evaluation and study the impact of morphological analysis on the top ranked bilingual scores using part-of-speech (POS) tagging[11], anaphora resolution, named entity recognition, and taking word sense into account.

Acknowledgments

We are grateful to Michael Elhadad and Galina Volk for providing the ROUGE toolkit adapted to Hebrew alphabet.

References

Adler, M. (2009). Hebrew morphological disambiguation: An unsupervised stochastic word-based approach. Dissertation. http://www.cs.bgu.ac.il/ adlerm/dat/thesis.pdf.

Baxendale, P. (1958). Machine-made index for technical literature-an experiment. *IBM Journal of Research and Development*, *2*, 354–361.

Brin, S., & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, *30*, 107–117.

Dang, H. T. (2006). Overview of DUC 2006. *Proceedings of the Document Understanding Conference*.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, *7*, 1–30.

DUC (2002). Document understanding conference. http://duc.nist.gov.

Edmundson, H. P. (1969). New methods in automatic extracting. *J. ACM*, *16*.

Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, *22*, 457–479.

[10] We will provide the code for our summarizer upon request.

[11] Our experiments have shown that syntactic filters, which select only lexical units of a certain part of speech, do not significantly improve the performance of the evaluated bilingual scoring methods.

Gong, Y., & Liu, X. (2001). Generic text summarization using relevance measure and latent semantic analysis. *Proceedings of the 24th ACM SIGIR conference on Research and development in information retrieval* (pp. 19–25).

Kallel, F. J., Jaoua, M., Hadrich, L. B., & Hamadou, A. B. (2004). Summarization at LARIS laboratory. *Proceedings of the Document Understanding Conference*.

Kupiec, J., Pedersen, J., & Chen, F. (1995). A trainable document summarizer. *Proceedings of the 18th annual international ACM SIGIR conference* (pp. 68–73).

Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. *Proceedings of the ACL'04 Workshop: Text Summarization Branches Out* (pp. 74–81).

Litvak, M., Last, M., & Friedman, M. (2010). A new approach to improving multilingual summarization using a genetic algorithm. *Proceedings of the Association for Computational Linguistics (ACL) 2010*. Uppsala, Sweden.

Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development*, *2*, 159–165.

Mani, I., & Maybury, M. (1999). *Advances in automatic text summarization*.

Mihalcea, R. (2005). Language independent extractive summarization. *AAAI'05: Proceedings of the 20th national conference on Artificial intelligence* (pp. 1688–1689).

Nastase, V., & Szpakowicz, S. (2006). A study of two graph algorithms in topic-driven summarization. *Proceedings of the Workshop on Graph-based Algorithms for Natural Language*.

Neto, J., Santos, A., Kaestner, C., & Freitas, A. (2000). Generating text summaries through the relative importance of topics. *Lecture Notes in Computer Science*, 300–309.

Radev, D., Blair-Goldensohn, S., & Zhang, Z. (2001). Experiments in single and multidocument summarization using MEAD. *First Document Understanding Conference*.

Saggion, H., Bontcheva, K., & Cunningham, H. (2003). Robust generic and query-based summarisation. *EACL '03: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*.

Salton, G., Singhal, A., Mitra, M., & Buckley, C. (1997). Automatic text structuring and summarization. *Information Processing and Management*, *33*, 193–207.

Satoshi, C. N., Satoshi, S., Murata, M., Uchimoto, K., Utiyama, M., & Isahara, H. (2001). Sentence extraction system assembling multiple evidence. *Proceedings of 2nd NTCIR Workshop* (pp. 319–324).

Schenker, A., Bunke, H., Last, M., & Kandel, A. (2004). Classification of web documents using graph matching. *International Journal of Pattern Recognition and Artificial Intelligence*, *18*, 475–496.

Steinberger, J., & Jezek, K. (2004). Text summarization and singular value decomposition. *Lecture Notes in Computer Science*, 245–254.

Vanderwende, L., Suzuki, H., Brockett, C., & Nenkova, A. (2007). Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion. *Information processing and management*, *43*, 1606–1618.

Wong, K., Wu, M., & Li, W. (2008). Extractive summarization using supervised and semi-supervised learning. *Proceedings of the 22nd International Conference on Computational Linguistics* (pp. 985–992).