ExB Text Summarizer

Stefan Thomas, Christian Beutenmüller, Xose de la Puente
Robert Remus and **Stefan Bordag**
ExB Research & Development GmbH
Seeburgstr. 100
04103 Leipzig, Germany
{thomas, beutenmueller, puente, remus, bordag}@exb.de

Abstract

We present our state of the art multilingual text summarizer capable of single as well as multi-document text summarization. The algorithm is based on repeated application of TextRank on a sentence similarity graph, a bag of words model for sentence similarity and a number of linguistic pre- and post-processing steps using standard NLP tools. We submitted this algorithm for two different tasks of the MultiLing 2015 summarization challenge: *Multilingual Single-document Summarization* and *Multilingual Multi-document Summarization*.

1 Introduction

The amount of textual content that is produced and consumed each day all over the world, through news websites, social media, and other information sources, is constantly growing. This makes the process of selecting the right content to read and quickly recognizing basic facts and topics in texts a core task for making content accessible to the users. Automatic summarization strives to provide a means to this end. This paper describes our automatic summarization system, and its participation in the MultiLing 2015 summarization challenge.

Our focus has been on producing a largely language-independent solution for the MultiLing 2015 challenge that, in contrast to most attempts in this field, requires a strict minimum of language-specific components and uses no language-specific materials for the core innovative elements.

Our motivation comes in part from Hong et al. (2014), who compares a number of single language summarization systems on the same standardized data set and shows that many complex, language-specific, highly optimized and trained methods do not significantly out-perform simplistic algorithms that date back to the first summarization competitions in 2004.

Language-independent text summarization is generally based on sentence extractive methods: A subset of sentences in a text are identified and combined to form a summary, rather than performing more complex operations, and the primary task of summarization algorithms is to identify the set of sentences that form the best summary. In this case, algorithms differ mostly in how sentences are selected.

One textual feature that has proven useful in identifying good summary sentences is the relative prominence of specific words in texts when contrasted to a reference distribution (like frequency in a large general corpus). For example, the "keyness" metric in El-Haj and Rayson (2013), singular value decomposition on a term-vector matrix (Steinberger, 2013) and neural network-derived transformations of term vectors (Kågebäck et al., 2014) have all produced significant results. There are also a number of rule-based approaches like Anechitei and Ignat (2013). Hong et al. (2014) provides an overview of various current approaches, ranging from simple baseline algorithms to complex systems with many machine learning and rule-based components of various kinds.

One promising recent approach is graph theory-based schemes which construct sentence similarity graphs and use various graph techniques to determine the importance of specific sentences as a heuristic to identify good summary sentences (Barth, 2004; Li et al., 2013b; Mihalcea and Tarau, 2004).

In this paper, we describe ExB's graph-based summarization approach and its results in two MultiLing 2015 tasks: *Multilingual Single-document Summarization* and *Multilingual Multi-document Summarization*. ExB's submissions covered all languages in each task. Furthermore,

Proceedings of the SIGDIAL 2015 Conference, pages 260–269,
Prague, Czech Republic, 2-4 September 2015.

we summarize and discuss some unexpected negative experimental results, particularly in light of the problems posed by summarization tasks and their evaluation using ROUGE (Lin, 2004).

2 Process Overview

The procedures used in both tasks start from similar assumptions and use a generalized framework for language-independent sentence selection-based summarization.

We start from the same basic model as LDA approaches to text analysis: Every document contains a mixture of topics that are probabilistically indicative of the tokens present in it. We select sentences in order to generate summaries whose topic mixtures most closely match that of the document as a whole (Blei et al., 2003).

We construct a graph representation of the text in which each node corresponds to a sentence, and edges are weighted by a similarity metric for comparing them. We then extract key sentences for use in summaries by applying the PageRank/TextRank algorithm, a well-studied algorithm for measuring graph centrality. This technique has proven to be good model for similar extraction tasks in the past (Mihalcea and Tarau, 2004).

We deliberately chose not to optimize any parameters of our core algorithm for specific languages. Every parameter and design decision applied to all languages equally and was based on cross-linguistic performance. Typically it is possible to increase evaluation performance by 2%-4% through fine tuning, but this tends to produce overfitting and the gains are lost when applied to any broader set of languages or domains.

Our approach consists of three stages:

1. Preprocessing using common NLP tools. This includes steps like tokenization and sentence identification, and in the multilingual summarization case, an extractor for time references like dates and specific times of day. These tools are not entirely language-independent.

2. Sentence graph construction and sentence ranking as described in Sections 2.2 and 2.3 respectively.

3. Post-processing using simple and language-independent rules for selecting the highest ranking sentences up to the desired length of text.

2.1 Preprocessing

Our processing pipeline starts with tokenization and sentence boundary detection. For most languages we employ ExB's proprietary language-independent rule-based tokenizer. For Chinese, Japanese and Thai tokenization we use language-dependent approaches:

- Chinese is tokenized using a proprietary algorithm that relies on a small dictionary, the probability distribution of token lengths in Chinese, and a few handcrafted rules for special cases.

- For Thai, we use a dictionary containing data from NECTEC (2003) and Satayamas (2014) to calculate the optimal partition of Thai letter sequences based on a shortest path algorithm in a weighted, directed acyclic character graph using dictionary terms found in the text.

- For Japanese, we employ the CRF-based *MeCab* (Kudo et al., 2004; Kudo, 2013) morphological analyzer and tokenizer. *MeCab* is considered state-of-the-art and is currently being used in the construction of annotated reference corpora for Japanese by Maekawa et al. (2014).

Sentence boundary detection is rule-based and uses all sentence separators available in the Unicode range of the document's main language, along with an abbreviation list and a few rules to correctly identify expressions like "p.ex." or "..."

Finally, we use a proprietary SVM-based stemmer trained for a wide variety of languages on custom corpora.

2.2 Graph construction

Given a set of tokenized sentences S, we construct a weighted undirected graph $G = (V, E)$, where each vertex $V_i \in V$ corresponds to a sentence in S. The weighted edges (S_i, S_j, w) of the graph are defined as a subset of $S \times S$ where $i \neq j$ and $(w \leftarrow sim(S_i, S_j)) \geq t$ for a given similarity measure sim and a given threshold t. We always assume a normalized similarity measure with a scale between 0 and 1.

Sentence similarity is computed with the standard vector space model (Salton, 1989), where each sentence is defined by a vector of its tokens.

We compared these vectors using a number of techniques:

- An unweighted *bag-of-words* model with sentence similarity computed using the Jacquard index.
- Conventional cosine similarity of sentence vectors weighted by term frequency in the sentence.
- TF-IDF weighted cosine similarity, where term frequencies in sentences are normalized with respect to the document collection.
- Semantic similarity measured using the *ExB Themis* semantic approach described in Hänig et al. (2015).

We also evaluated different settings for the threshold t. We did not optimize t separately for different languages, instead setting a single value for all languages.

Surprisingly, when averaged over all 38 languages in the MSS training set, the simple bag-of-words model with a threshold $t = 0.3$ produced the best result using the ROUGE-2 measure.

2.3 Sentence ranking

We then apply to the sentence similarity graph an iterative extension of the *PageRank* algorithm (Brin and Page, 1998) that we have called *FairTextRank* (*FRank*) to rank the sentences in the graph. *PageRank* has been used as a ranker for an extractive summarizer before in Mihalcea and Tarau (2004), who named it *TextRank* when used for this purpose. *PageRank* constitutes a measure of graph centrality, so intuitively we would expect it to select the most central, topical, and summarizing sentences in the text.

Following our assumption that every document constitutes a mix of topics, we further assume that every topic corresponds to a cluster in the sentence graph. However, *PageRank* is not a cluster sensitive algorithm and does not, by itself, ensure coverage of the different clusters present in any graph. Therefore, our *FRank* algorithm invokes *PageRank* iteratively on the graph, at each step ranking all the sentences, then removing the top ranking sentence from the graph, and then running *PageRank* again to extract the next highest ranking sentence. Because the most central sentence in the entire graph is also, by definition, the most central sentence in some cluster, removing it weakens the centrality of the other sentences in that cluster and increases the likelihood that the next sentence selected will be the highest ranking sentence in another cluster.

A similar method of removing selected sentences is used in the *UWB Summarizer* by Steinberger (2013), which was one of the top performing systems at MultiLing 2013. However, the *UWB Summarizer* uses an LSA algorithm on a sentence-term matrix to identify representative sentences, where we have employed *PageRank*.

The complete algorithm is detailed in Algorithm 1. The function adj returns the weighted adjacency matrix of the sentence graph G. An inner for-loop transforms the weighted adjacency matrix into a column-stochastic matrix where for each column c, where $A[i, c]$ is the weight of the edge between sentence i and sentence c, the following expression holds: $\sum_{i \in |A|} A[i, c] = 1$. Informally, each column is normalized at each iteration so that its values sum to 1. pr is the *PageRank*-algorithm with the default parameters $\beta = 0.85$, a convergence threshold of 0.001 and allowed to run for at most 100 iterations as implemented in the JUNG API (O'Madadhain et al., 2010).

Algorithm 1 FairTextRank

function FRANK(G)
  $R \leftarrow []$
  while $|G| > 0$ **do**
    $A \leftarrow adj(G)$
    for $(r, c) \leftarrow |A|^2$ **do**
      $A_{norm}[r, c] \leftarrow \frac{A[r,c],}{\sum_{i \in |A|} A[i,c]}$
    $rank \leftarrow pr(A_{norm})$
    $v \leftarrow rank[0]$
    $R \leftarrow R + v$
    $G \leftarrow G \setminus v$
  return R

2.4 Post-processing

The final step in processing is the production of a plain text summary. Given a fixed maximum summary length, we selected the highest ranked sentences produced by the ranking algorithm until total text length was greater than the maximum allowed length, then truncated the last sentence to fit exactly the maximum allowed length. Although this reduces the human readability of the summary - the last sentence is interrupted without any consideration of the reader at all - it can only increase

the score of an n-gram based evaluation metric like ROUGE.

3 Single Document Summarizer

The *Multilingual Single-document Summarization* (MSS) task consisted of producing summaries for Wikipedia articles in 38 languages. All articles were provided as UTF-8 encoded plain-text files and as XML documents that mark sections and other elements of the text structure. We took advantage of the availability of headers and section boundary information in performing this task.

There was no overlap between the training data and the evaluation data for the MSS task. The released training data consisted of the evaluation data set from MultiLing 2013 as described in Kubina et al. (2013). This training data contains 30 articles in each of 40 languages. The MSS task itself at MultiLing 2015 used 30 articles in each of 38 languages, dropping two languages because there were not enough new articles not included in the training data.

In addition to the preprocessing steps described in Section 2.1, for this task we applied a list of sentence filters developed specifically for Wikipedia texts:

- Skip all headers.
- Skip every sentence with with less than 2 tokens (mostly errors in sentence boundary detection).
- Skip every sentence that contains double quotes.

We then performed sentence graph construction and ranking as described in Sections 2.2 and 2.3

In the post-processing stage, we sorted the sentences selected to go into the summary in order of their position in the original article, before producing a plain text summary by concatenating them.

3.1 Results

The organizers of the MultiLing 2015 challenge measured the quality of our system's output using five different versions of the ROUGE score. We provide a summary of the results for all participants in Table 1. It shows the average ranking of each participating system over all the languages on which it was tested, as well as the number of languages on which each system was tested. The systems labelled **Lead** and **Oracles** are special systems. **Lead** just uses the beginning of the article as the summary and represents a very simple baseline. **Oracles**, on the other hand, is a cheating system that marks the upper bound for any extractive approach.

Only three submissions - highlighted in bold - participated in more than 3 languages. We submitted only one run of our system, defined as a fixed set of parameters that are the same over all languages. One of the other two systems that participated in all 38 languages submitted five runs. According to the frequently used ROUGE-1 and ROUGE-2 scores, our system achieved an average ranking of 3.2 and 3.3, respectively. This table shows that the CCS system performed better on average than our system, and the LCS-IESI system performed on average worse.

However, ROUGE-1 only measures matching single words, whereas ROUGE-2 measures matching bigrams. More complex combinations of words are more indicative of topic matches between gold standard data and system output. We believe that ROUGE-SU4, which measures bigrams of words with some gaps as well as unigrams, would be a better measure of output quality. When manually inspecting the summaries, we have the strong impression that system runs in which our system scored well by ROUGE-SU4 measures, but poorly by ROUGE-2, did produce better summaries with greater readability and topic coverage.

Our system achieves a significantly better overall ranking using ROUGE-SU4 instead of ROUGE-2, even though the system was optimized to produce the highest ROUGE-2 scores. Only two runs of the winning system CCS scored better than our system according to ROUGE-SU4. This underlines the robustness of our system's underlying principles, despite the known problems with ROUGE evaluations.

4 Multi Document Summarizer

The *Multilingual Multi-document Summarization* (MMS) task involves summarizing ten news articles on a single topic in a single language. For each language, the dataset consists of ten to fifteen topics, and ten languages were covered in all, including and expanding on the data used in the 2013 MMS task described by Li et al. (2013a).

The intuition guiding our approach to this task is the idea that if news articles on the same topic contain temporal references that are close together

Competitor system	Langs.	Rank R-1	Rank R-2	Rank R-3	Rank R-4	Rank R-4SU
BGU-SCE-M	3	2.0	3.3	3.7	4.3	3.0
BGU-SCE-P	3	5.0	4.7	5.0	4.3	4.3
CCS	38	2.1	2.1	2.2	2.3	2.5
ExB	38	3.2	3.3	3.7	3.8	2.8
LCS-IESI	38	4.1	4.1	4.0	4.0	4.1
NTNU	2	5.5	6.0	6.0	7.0	5.0
UA-DLSI	3	6.0	5.0	4.7	5.0	6.0
Lead	38	5.1	5.0	4.6	4.3	5.0
Oracles	38	1.1	1.2	1.2	1.2	1.2

Table 1: Number of covered languages and average rank for each system in MSS competition for ROUGE-(1,2,3,4,4-SU) measures. In bold, competitors in all available languages. *Lead* and *Oracles* are two reference systems created by the organizers.

or overlapping in time, then they are likely to describe the same event. We therefore cluster the documents in each collection by the points in time referenced in the text rather than attempting to summarize the concatenation of the documents directly. This approach has the natural advantage that we can present summary information in chronological order, thereby often improving readability. Unfortunately, this improvement is not measurable using ROUGE-style metrics as employed in evaluating this task.

An official training data set with model summaries was released, but too late to inform our submission, which was not trained with any new 2015 data. We did, however, use data from the 2011 MultiLing Pilot including gold standard summaries (Giannakopoulos et al., 2011), which forms a part of the 2015 dataset. We used only the 700 documents and summaries from the 2011 task as training data, and did not use any Chinese, Spanish or Romanian materials in preparing our submission.

Our submission follows broadly the same procedure as for the single document summarization task, as described in Section 2 and Section 3, except for the final step, which relies on section information not present in the news articles that form the dataset for this task. Instead, a manual examination of the dataset revealed that the news articles all have a fixed structure: the first line is the headline, the second is the date, and the remaining lines form the main text. We used this underlying structure in preprocessing to identify the dateline of the news article, and we use this date to disambiguate relative time expressions in the text like "yesterday" or "next week". Articles are also ordered in time with respect to each other on the basis of the article date.

Furthermore, we remove in preprocessing any sentence that contains only time reference tokens because they are uninformative for summarization.

We then extract temporal references from the text, using ExB's proprietary *TimeRec* framework described in Thomas (2012), which is available for all the languages used in this task. With the set of disambiguated time references in each document, we can provide a "timeframe" for each document that ranges from the earliest time referenced in the text to the latest. Note that this may not include the date of the document itself, if, for example, it is a retrospective article about an event that may have happened years in the past.

4.1 Time information processing

Ng et al. (2014) and Wan (2007) investigate using textural markers of time for multi-document summarization of news articles using very different algorithms. Our approach is more similar to Ng et al. in constructing a timeline for each document and for the collection as a whole based on references extracted from texts. Once document timeframes are ordered chronologically, we organize them into groups based on their positions on a time line. We explored two strategies to produce these groups:

- **Least Variance Clustering (LVC)**: Grouping the documents iteratively by adding a new document to the group if the overall variance of the group doesn't go over a threshold. We set the standard deviation limit of the group

in 0.1. The algorithm is a divisive clustering algorithm based on the central time of the documents and the standard deviation. At first the minimal central time of a document collection is subtracted from all other central times, then we compute mean, variance and standard deviation based on days as a unit and normalized by the mean. Afterwards we recursively split the groups with the goal to minimize the variance of both splits until either a group consists only of one document or the recomputed standard deviation of a group is less than 0.1.

- **Overlapping Time Clustering (OTC)**: Grouping documents together if their timeframes overlap more than a certain amount, which we empirically set to 0.9 after experimenting with various values. This means that if two texts A and B are grouped together, then either A's timeframe includes at least 90% of B's timeframe, or B's timeframe includes 90% of A's. This approach proceeds iteratively, with each new addition to a group updating the timeframe of the group as a whole, and any text which overlaps more than 90% with this new interval is then grouped with it in the next iteration.

In addition, we provide two baseline clusterings:

- **One document per cluster (1PC)**: Each document is in a cluster by itself.
- **All in one cluster (AIO)**: All documents from one topic are clustered together.

In the LVC and OTC cases, clustering is iterative and starts with the earliest document as determined by a fixed "central" date for each document. We explored different ways of determining that "central" date: One was using the dateline found in preprocessing on the second line of each document, another was the median of the time references in the document. Our best result used the dateline from each article and, as can be seen in Table 2, was produced by the OTC strategy. This is a surprising result, as we expected LVC to perform better since variance is generally a better measure of clustering. However, we found that LVC generally produced more clusters than OTC and we believe that to account for its poor performance. We experimented with a number of other ordering and clustering approaches, although they do not figure into our submission to the MMS task, but in all cases they failed to out-perform the OTC approach according to the ROUGE-2 recall measure.

For all conditions, identical preprocessing was performed using ExB's proprietary language-specific tokenizer and sentence identifier. ROUGE scores, because they are based on token n-grams, are very sensitive to discrepancies between tokenizers and stemmers. In English, because most tokenizers perform very similarly, this causes fewer problems in scoring than for Arabic or other languages where tokenizers vary dramatically. We used the results in Table 2 to decide which conditions to use in the competition, but we cannot be sure to what degree our results have been influenced by these kinds of ROUGE-related problems.

After clustering, we perform graph-based sentence ranking as described in Sections 2.2 and 2.3 separately for each cluster. We then select sentences from each cluster, ensuring that they are all represented in the final summary, so that the entire time span of the articles is covered. We also order the selected sentences in the summary based on the temporal ordering of the clusters, so that summary presentation is in event order.

4.2 Experimental results

When experimenting with the challenge data we made several observations:

1. Since the dataset of MMS is composed of news articles, just selecting the headlines and first sentences will produce a strong baseline with very high ROUGE scores. It is difficult to beat this baseline using sentence extraction techniques.
2. The quality of the summaries varies a great deal between languages. Instead of producing fine-tuned configurations for each lan-

Clustering Algorithm	English	Arabic
1PC	18.08	26.06
AIO	18.94	24.5
LVC	15.54	24.25
OTC	**19.81**	25.34
1PC-Reorder	17.69	**33.63**

Table 2: ROUGE-2 recall results for different grouping algorithms in MMS-2011 dataset.

Language	AutoSummENG	MeMoG	NPowER	Rank/Total
Arabic	0.135	0.164	1.717	7/9
Chinese	0.118	0.141	1.654	1/5
Czech	0.188	0.2	1.874	4/7
English	0.167	0.191	1.817	6/10
French	0.2	0.195	1.892	5/8
Greek	0.147	0.17	1.75	5/8
Hebrew	0.115	0.147	1.655	8/9
Hindi	0.123	0.139	1.662	3/7
Romanian	0.168	0.183	1.809	4/6
Spanish	0.193	0.202	1.886	3/6

Table 3: Average per-language Score ranked against the best run of each system in MMS competition for MeMoG measure.

guage that optimize ROUGE scores, we focused on increasing the performance in English - a language we can read and in which we can qualitatively evaluate the produced summaries.

3. All the results here of the time information processing are at document-level. We also tried to apply the time grouping algorithms per sentence, but we noticed a drop of about 3% ROUGE-2 score on average.

The most important finding is that using temporal expressions and chronological information does improve the performance of the summary system, and that the iterative FairTextRank algorithm shows a solid performance even for multiple documents.

As can be seen in Table 3, our system gets ranked in middle position in the official scores of the challenge using the *NPowER*, *MeMoG* and *AutoSummENG* measures as described in Giannakopoulos and Karkaletsis (2013) and Giannakopoulos and Karkaletsis (2011). We also note that our system out-performs all other participants in Chinese, a language for which we had no training data.

5 Negative results

We feel that it is important not only to publish positive results, but also negative ones, to counter the strong publication bias identified in many areas in the natural and social sciences (Dickersin et al., 1987; Ioannidis, 2005). Since we conducted a large number of experiments in creating this system, we inevitably also came across a number of ideas that seemed good, but turned out to not improve our algorithm, at least as measured using ROUGE-2.

In another challenge participation we developed a very powerful "semantic text similarity" (STS) toolkit. In *SemEval 2015* Task 2 (Agirre et al., 2015), it achieved by far the highest scores for Spanish texts and the second best scores for English. Since our text summarization methodology is based on a sentence similarity graph, our intuitive hypothesis was that when using this module as opposed to simple matching-words strategies, performance should increase significantly. Matching-words strategies are used as the baseline in SemEval tasks, and it is easily out-performed by more sophisticated approaches.

Therefore, we tried out our STS module as a replacement for Jacquard and cosine similarity measures when constructing the sentence graph, while keeping all other parameters fixed. Surprisingly, it did not improve performance, and lowered ROUGE-2 scores by 2%. We also attempted to use *word2vec* embeddings precomputed on very large corpora (Mikolov et al., 2013) to represent words and hence compute a much finer-grained sentence similarity, but those results were 4% worse. It is possible that those systems were, in fact, better, but because ROUGE scoring focuses on word matches, any other improvement cannot be measured directly. We also attempted to include other factors such as sentence length, position, number of named entities, temporal expressions, and physical measurements into the sentence similarity score, all without seeing any increase in ROUGE scores.

Since identifying temporal expressions increases ROUGE scores, as this paper shows, we surmised that name recognition might also improve summarization. We applied our named entity recognition system, which is available in a number of different languages and won the Germeval 2014 (Benikova et al., 2014) NER challenge, and weighted more heavily sentences with detected names before extracting summary sentences. Interestingly, no matter how the weighting scheme was set up, the performance of the system always dropped by a few percent. Often, the system would select useless sentences that contain long lists of participating authors, or enumerations of entities participating in some reported event. Even when these kinds of sentences are explicitly removed, it still selects sentences that simply contain many names with little relevance to the topics of the news article. We conclude that sen-

tences describing central topics in documents are not strongly correlated with named entity usage.

Another very intuitive assumption is that filtering stop words, or down-weighting very frequent words, or using a TF-IDF based scheme with a similar effect, would improve the results. However, we did not observe any improvement by using these techniques. Nonetheless, there are strong indications that this is due to the limitations of ROUGE-2 scoring and we cannot conclude that these kinds of techniques are useless for summarization. It is easy to achieve very competitive ROUGE-2 scores by just filling the summary with very frequent stop word combinations. A human would immediately recognize the uselessness of such a "summary", but ROUGE-2 would count many bigram matches with a gold standard summary.

Finally, we considered the hypothesis that the summary system could be helped by explicitly removing very similar sentences presenting redundant information. Surprisingly, explicitly removing such sentences did not improve the performance of the system. Manually inspecting a number of summaries, we notice that very similar sentences recurring often in texts are rarely selected by the *FRank* algorithm. We believe this is because our approach is sufficiently robust to discount these sentences on its own.

6 Conclusions

In this paper we outline ExB's largely language-independent system for text summarization based on sentence selection, and show that it supports at least the 38 languages used in this completion without any language-specific fine-tuning. Sentences are selected using an iterative extension of *PageRank calculation* on a sentence similarity graph. Our results in the MultiLing 2015 challenge have validated this approach by achieving the best scores for several languages and competitive scores for most of them, generally surpassed by only one other participating system.

We also show that one basic summarization system can apply to different domains, different languages, and different tasks without special configuration, while retaining state-of-the-art performance. Furthermore, for multi-document news summarization, we show that extracting temporal expressions is a useful feature for combining articles on the same topic.

Our most relevant conclusion is that both the current evaluation methodology (based on various forms of ROUGE) as well as the current principal approach to language-independent text summarization (context-free, sentence selection based) are highly inadequate to model the vague requirements users associate with a text summarization product.

Participants in MultiLing 2015 did not receive the scripts and parameters used in producing evaluations. This made it difficult to optimize parameters and algorithms and has a significant impact on results using ROUGE measures and probably the other measures as well. Hong et al. (2014), for example, notes values between 30.8% and 39.1% using ROUGE-1 for one well-known algorithm on one data set by different authors. It is not clear how the vastly different scores obtained for identical summaries using different ROUGE parameters correlate with the objective quality of a given summary. We have no clear indication that ROUGE scores really capture the quality of a given summary at all.

While it is possible to formulate summarization solutions based on sentence selection and even iteratively improve them using ROUGE scores, the actual achievable performance measured using ROUGE is very low. We have noticed that stemming, stopword filtering and various tokenization strategies can have a very large influence on ROUGE scores, especially in morphologically richer languages than English. More modern evaluation measures like *MeMog* or *NPoweR* might solve the problems inherent to ROUGE, however they currently lack widespread adoption in the research community.

Nonetheless, even if these issues in evaluation can be addressed, we do not believe that summaries based on sentence selection will ever reach a quality where they could be accepted as comparable to a human written summary.

References

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Iigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, German Rigau, Larraitz Uria, and Janyce Wiebe. 2015. SemEval-2015 Task 2: Semantic Textual Similarity, English, Spanish and Pilot on Interpretability. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, Denver, CO, June. ACL.

Daniel Alexandru Anechitei and Eugen Ignat. 2013. Multilingual summarization system based on analyzing the discourse structure at MultiLing 2013. *Proceedings of the Multiling 2013 Workshop on Multilingual Multi-document Summarization*, pages 72–76.

Michael Barth. 2004. Extraktion von Textelementen mittels "spreading activation" für indikative Textzusammenfassungen. Master's thesis, Universität Leipzig. Fakultät für Mathematik und Informatik. Institut für Informatik.

Darina Benikova, Chris Biemann, Max Kisselew, and Sebastian Pad. 2014. GermEval 2014 Named Entity Recognition Shared Task: Companion Paper. In *Proceedings of the KONVENS GermEval Shared Task on Named Entity Recognition*, pages 104–112, Hildesheim, Germany.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *J. Mach. Learn. Res.*, 3:993–1022, March.

Sergey Brin and Lawrence Page. 1998. The Anatomy of a Large-scale Hypertextual Web Search Engine. *Comput. Netw. ISDN Syst.*, 30(1-7):107–117, April.

K. Dickersin, S. Chan, T. C. Chalmers, H. S. Sacks, and Smith. 1987. Publication bias and clinical trials. *Controlled Clinical Trials*, 8(4):343–353.

Mahmoud El-Haj and Paul Rayson. 2013. Using a Keyness Metric for Single and Multi Document Summarisation. *Proceedings of the Multiling 2013 Workshop on Multilingual Multi-document Summarization*, pages 64–71.

George Giannakopoulos and Vangelis Karkaletsis. 2011. AutoSummENG and MeMoG in Evaluating Guided Summaries. In *TAC 2011 Workshop NIST Gaithersburg, MD, USA*.

George Giannakopoulos and Vangelis Karkaletsis. 2013. Summary Evaluation: Together We Stand NPowER-ed. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, volume 7817 of *Lecture Notes in Computer Science*, pages 436–450. Springer Berlin Heidelberg.

George Giannakopoulos, Mahmoud El-Haj, Benoit Favre, Marina Litvak, Josef Steinberger, and Vasudeva Varma. 2011. TAC2011 MultiLing Pilot Overview. In *TAC 2011 Workshop*, Gaithersburg, MD, USA. NIST.

Christian Hänig, Robert Remus, and Xose de la Puente. 2015. ExB Themis: Extensive Feature Extraction from Word Alignments for Semantic Textual Similarity. In *Proceedings of the 9th International Workshop on Semantic Evaluations*, Denver, USA. ACL - to appear.

Kai Hong, John Conroy, Benoit Favre, Alex Kulesza, Hui Lin, and Ani Nenkova. 2014. A Repository of State of the Art and Competitive Baseline Summaries for Generic News Summarization. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 1608–1616, Reykjavik, Iceland, May. European Language Resources Association (ELRA).

John P. A. Ioannidis. 2005. Why Most Published Research Findings Are False. *PLoS Med*, 2(8):e124, 08.

Mikael Kågebäck, Olof Mogren, Nina Tahmasebi, and Devdatt Dubhashi. 2014. Extractive Summarization using Continuous Vector Space Models. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*, pages 31–39. ACL.

Jeff Kubina, John Conroy, and Judith Schlesinger. 2013. ACL 2013 MultiLing Pilot Overview. In *Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization*, pages 29–38, Sofia, Bulgaria, August. ACL.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying Conditional Random Fields to Japanese Morphological Analysis. In *Proceedings of EMNLP 2004*, pages 230–237. ACL.

Taku Kudo. 2013. MeCab: Yet Another Part-of-Speech and Morphological Analyzer. `http://mecab.googlecode.com/svn/trunk/mecab/doc/index.html`. Accessed: 2015-04-24.

Lei Li, Corina Forascu, Mahmoud El-Haj, and George Giannakopoulos. 2013a. Multi-document multilingual summarization corpus preparation, Part 1: Arabic, English, Greek, Chinese, Romanian. In *Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization*, pages 1–12, Sofia, Bulgaria, August. ACL.

Lei Li, Lei Heng, Jia Yu, and Yu Li. 2013b. CIST System Report for ACL MultiLing 2013. *Proceedings of the Multiling 2013 Workshop on Multilingual Multi-document Summarization*, pages 39–44.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In Stan Szpakowicz Marie-Francine Moens, editor, *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain, July. ACL.

Kikuo Maekawa, Makoto Yamazaki, Toshinobu Ogiso, Takehiko Maruyama, Hideki Ogura, Wakako Kashino, Hanae Koiso, Masaya Yamaguchi, Makiro Tanaka, and Yasuharu Den. 2014. Balanced corpus of contemporary written Japanese. *Language Resources and Evaluation*, 48(2):345–371.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into texts. *Proceedings of EMNLP 2004*, pages 404–411.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the Workshops at ICLR 2013*, volume abs/1301.3781.

NECTEC. 2003. LEXiTRON. `http://www.nectec.or.th/`. An adapted version of LEXiTRON developed by NECTEC.

Jun-Ping Ng, Yan Chen, Min-Yen Kan, and Zhoujun Li. 2014. Exploiting Timelines to Enhance Multi-document Summarization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL*, pages 923–933.

Joshua O'Madadhain, Danyel Fisher, and Tom Nelson. 2010. JUNG: Java Universal Network/Graph Framework. `http://jung.sourceforge.net/`.

Gerard Salton. 1989. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Vee Satayamas. 2014. thailang4r. `https://github.com/veer66/thailang4r`. Accessed: 2015-04-24.

Josef Steinberger. 2013. The UWB Summariser at Multiling-2013. *Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization*, pages 50–54.

Stefan Thomas. 2012. Verbesserung einer Erkennungs- und Normalisierungsmaschine für natürlichsprachige Zeitausdrücke. Master thesis, Universität Leipzig. Fakultät für Mathematik und Informatik. Institut für Informatik.

Xiaojun Wan. 2007. TimedTextRank: adding the temporal dimension to multi-document summarization. In *SIGIR 2007: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 867–868.