MUSEEC: A Multilingual Text Summarization Tool

Marina Litvak
Department of Software Engineering
Shamoon College of Engineering
Beer Sheva, Israel
marinal@sce.ac.il

Alexander Dlikman
Department of Information Systems Engineering
Ben Gurion University of the Negev
Beer Sheva, Israel
dlikman@post.bgu.ac.il

Mark Last
Department of Information Systems Engineering
Ben Gurion University of the Negev
Beer Sheva, Israel
mlast@bgu.ac.il

Abstract

The MUSEEC tool builds upon MUSE (MUltilingual Sentence Extractor), a language-independent summarization algorithm that ranks each sentence in a summarized document by a score, calculated as a weighted linear combination of multiple sentence features. The sentences with the highest score are then selected for the summary. In our previous experiments, which included only language-independent (statistical) features, MUSE has consistently outperformed TextRank, the state-of-the-art language-independent approach to automated text summarization, in four different languages: English, Hebrew, Arabic, and Farsi. MUSEEC extends the original set of MUSE features with novel linguistic metrics based on Part-of-Speech (POS) tagging.

In this paper we provide an overview of MUSEEC's participation in the MultiLing 2015 single document (MSS) and multi-document (MMS) summarization tasks on three languages: English, Hebrew and Arabic.

1 Introduction

High quality automated summaries can significantly reduce the information overload of many professionals in a variety of fields. Moreover, the publication of information on the Internet in an ever-increasing variety of languages increases the importance of developing *multilingual* summarization approaches that can extract the salient parts of text in any language. *Extractive summarization* usually consists of ranking fragments of a summarized text by relevance scores and extracting the top-ranked fragments into a summary. There is a multitude of statistical methods for ranking text parts: *cue*-based (Edmundson, 1969), *keyword*- or *frequency*-based (Luhn, 1958; Edmundson, 1969; Neto et al., 2000; Steinberger and Jezek, 2004; Kallel et al., 2004; Vanderwende et al., 2007), *title*-based (Teufel and Moens, 1997), *position*-based (Baxendale, 1958; Lin and Hovy, 1997) and *length*-based (Nobata et al., 2001).

Many works applied linear combinations of statistical scores for sentence ranking (Edmundson, 1969; Radev et al., 2001; Saggion et al., 2003; Goldstein et al., 1999). MUSE (Last and Litvak, 2012) made the first attempt to find the optimal weights of multiple language-independent scores, using a genetic algorithm (GA).

2 MUSEEC: Overview

MUSEEC is a multi-lingual text summarization platform built upon the MUSE algorithm. Sections below describe MUSE and its MUSEEC extension in detail.

2.1 MUltilingual Sentence Extractor (MUSE)

Given a collection of text documents, their gold standard summaries, and a target summary length, MUSE implements a *supervised* learning approach to extractive summarization for maximizing the similarity of automated summaries to gold standard summaries. This not a standard classification task, since the goal is to find the best ranking order of sentences in each document. A sentence ranking score is calculated as a weighted linear combination of multiple sentence features. The best set of feature weights is found by a GA. The obtained weighting vector can be used for sentence scoring in summarization of future documents.

Using the MUSEEC tool, the user can choose a subset of sentence metrics to be included by MUSE in the linear combination. In (Last and

Litvak, 2012), we have presented 31 statistical metrics that do not rely on any language-specific knowledge. These metrics have been divided into three main categories—*structure*-, *vector*-, and *graph*-based—according to their text representation model, where each sub-category contains a group of metrics using the same scoring method. A detailed description of language-independent sentence metrics used by MUSE can be found in (Last and Litvak, 2012).

A typical GA requires (1) a genetic representation of the solution domain, (2) a fitness function to evaluate the solution domain, and (3) some basic parameter settings such as selection and reproduction rules. We represent each solution as a fixed-size vector of feature weights —real-valued numbers in an unlimited range, which are normalized in such a way that they sum up to 1. Defined over the genetic representation, the fitness function measures the quality of the represented solution. We used ROUGE-1 and ROUGE-2, Recall (Lin and Hovy, 2003) [1] as fitness functions for measuring the summarization quality—similarity to gold standard summaries, which should be *maximized* during the training (optimization procedure). The reader is referred to (Litvak et al., 2010) for a detailed description of the optimization procedure implemented by MUSE.

The training time of the GA is proportional to a product of the number of GA iterations, the GA population size, and the fitness (ROUGE) evaluation time. On average, in our experiments the GA performed $5 - 6$ iterations of selection and reproduction before reaching convergence.

2.2 MUSEEC Architecture

As a supervised learner, MUSE consists of two major stages: *training* (model construction) and *summarization* (model usage).

The *training module* receives as input a corpus of documents, each accompanied by one or several gold-standard summaries—abstracts or extracts—compiled by human assessors. The set of documents may be either monolingual or multilingual, but their summaries must be in the same language as the original text. The *training module* applies a GA to a document-feature matrix of precomputed sentence scores with the purpose of finding the best linear combination of features using any ROUGE metric as a fitness function. The output (model) of the training module is a vector of weights for user-specified sentence ranking features.

The *summarization module* performs summarization of input texts in real time. Each sentence of an input text is assigned a relevance score according to the trained model, and the top ranked sentences are extracted to the summary in their original order. The length of resulting summaries is limited by a user-specified value (maximum number of characters, words or sentences, or a compression ratio). Being activated in real-time, the *summarization module* is expected to use the model trained on the same language as input texts. However, if such model is not available (no annotated corpus in the text language), the user can use a model trained on some other language or corpus. In (Last and Litvak, 2012) it is shown that the same model can be efficiently used across different languages.

MUSEEC performs the following pre-processing operations: (1) sentence segmentation (to be able to score individual sentences), (2) word segmentation (for calculating word-related features), (3) stemming (if available), and (4) stop-words removal (if available). The basic version of MUSE uses only statistical features and thus it can work with documents written in arbitrary language by treating the text as a sequence of UTF-8 characters. The extended version of MUSE uses also linguistic features, which require POS tagging as an additional pre-processing step.

The generated summaries can be post-processed by anaphora resolution (AR) and named entity (NE) tagging operations, if the corresponding tools are provided for a given language.

2.3 MUSE Extension with Part-of-Speech based Features

Most extractive summarization approaches make wide usage of statistical, language-independent features that do not require any natural language processing (Gupta and Lehal, 2010). The advantage of those features is their ability to engage in multi-lingual schemes and the simplicity of their calculation. On the other hand, the statistical metrics have been in use since 1958 (Luhn, 1958) and perhaps have already reached their performance limit.

The Part-of-Speech (POS) grammatical data

[1] We utilized the language-independent implementation of ROUGE that handles Unicode characters (Krapivin, 2014)

can indicate, to an extent, the presence or absence of information content in texts (Lioma and Blanco, 2009). Several text analytics methods have already used POS information. Lioma and Blanco (2009) show how POS–based term weighting improves performance in Information Retrieval tasks. Leskovec et al (2005) use POS tags as words features to train a supervised extractive summarizer with SVM. Al-Hashemi (2010) employs human-generated rules based on POS sequences in an extractive summarization system.

In MUSEEC, we have implemented a list of 17 POS-based sentence features (Table 1). Some of them are novel and others are interpretations of certain metrics used in the original MUSE summarizer (Last and Litvak, 2012). The proposed POS-based features take into account only nouns, verbs, adjectives and adverbs, due to the semantic importance of these parts of speech (Lioma and Blanco, 2009). These features can be divided into *POS ratio*-based, defined as a ratio between certain POS counts in a sentence and the sentence length, *POS filtering* that employs the original MUSE features after keeping particular POS and discarding the rest of the words, and *POS patterns*, which take into account POS n-grams, which are frequent in human-generated summaries and, at the same time, relatively rare in the original texts.

While the first two methods are straightforward, the metrics of the POS patterns are defined below. We assume that the presence of a specific POS pattern in a candidate sentence may indicate sentence relevance in the summary (Al-Hashemi, 2010). Our method requires a pre-processing stage where the relevance of the candidate POS patterns is calculated. We define a POS pattern relevance as a ratio between normalized pattern frequency in human–generated summaries and normalized pattern frequency in the corpora. The measure is greater than one when the POS n–gram is relatively more frequent in summaries than in original texts. In the last stage, we sum up all POS n-gram relevance measures, which are greater than one, and normalize this value by the total amount of n–grams in a sentence. In the current work, we calculated the above metrics separately for 2-, 3- and 4-grams of parts of speech.

Features	Description
POS ratio – 4 features	Ratio between the counts of nouns, verbs, adjectives, or adverbs in a sentence and the total sentence length (before POS filtering)
POS filtering – 10 features	MUSE, vector–based features (except for LUHN and SVD features) calculated after keeping only nouns, verbs, adjectives, and adverbs POS
POS patterns –3 features	Sum of POS 2-, 3- or 4-gram relevance measures in a sentence normalized by the total amount of n–grams (with the same n) in the sentence

Table 1: Part-of-Speech based features

3 MultiLing 2015 Results

Tables 3, 4, and 5 contain the summarized results of automated evaluations for the MultiLing 2015 single-document summarization (MSS) task. This task is aimed at generating single document summaries for a set of 30 Wikipedia articles in one or several of about 40 languages provided. The provided training data is the 2013 Single-Document Summarization Pilot Task data from MultiLing 2013. For each document of the training set, the human-generated summary is provided. Each machine summary should be as close to the pre–defined target length as possible. In the competition, the machine summaries are evaluated automatically by their similarity to the Gold Standard summaries that were withheld from the participants. The quality of the summaries is measured by ROUGE-1 (Recall, Precision, and F-measure) (Lin, 2004). We also present the absolute ranks of each submission–P-Rank, R-Rank, and F-Rank–with their scores sorted by Precision, Recall, and F-measure, respectively. Only the best submissions (in terms of F-measure) for each participating system are presented and sorted in descending order of their F-measure scores. Two systems–Oracles and Lead–were used as top-line and baseline summarizers, respectively. The Oracles computed summaries for each article using the combinatorial covering algorithm in (Davis et al., 2012)–sentences were selected from a text to maximally cover the tokens in the human summary, using as few sentences as possible, until its size exceeded the human summary, at which point it was truncated. Since the Oracles can actually "see" the human summaries, it is considered as the optimal algorithm and its scores are the best scores that extractive approaches can achieve. The Lead simply extracts the leading substring of the body text of the articles having the same length as the human summary of the article.

Five submissions, where summaries were generated by MUSE with different settings, were provided to the MultiLing evaluators. Table 2 shows the descriptions for each submission. We

distinguish between basic and extended versions of MUSE (described in section 2.3). The basic MUSE was applied with models trained on the source files provided within the MultiLing 2015 training set and on manually cleaned files (the source files contained some "trash", such as anchor text, from parsing HTML sources). Also, AR was performed after applying the basic MUSE in two submissions. Only two of five submissions were provided for Hebrew (runs of BGU-SCE-M-1 and BGU-SCE-M-5) and one for Arabic (BGU-SCE-M-5).

submission	description
BGU-SCE-M-1	MUSE basic (trained on clean files)
BGU-SCE-M-2	MUSE extended (trained on source files and DUC'02 corpus)
BGU-SCE-M-3	MUSE basic + AR (trained on source files)
BGU-SCE-M-4	MUSE basic + AR (trained on clean files)
BGU-SCE-M-5	MUSE basic (trained on source files)

Table 2: MUSEEC submissions.

system	P score	R score	F score	P-Rank	R-Rank	F-Rank
Oracles	0.601	0.619	0.610	1	1	1
BGU-SCE-M-5	**0.488**	**0.500**	**0.494**	**2**	**3**	**2**
BGU-SCE-M-1	**0.484**	**0.492**	**0.487**	**3**	**4**	**6**
CCS	0.477	0.495	0.485	4	6	3
BGU-SCE-P	0.475	0.494	0.484	5	8	5
EXB	0.467	0.495	0.480	9	13	4
BGU-SCE-M-2	**0.480**	**0.478**	**0.479**	**11**	**5**	**11**
BGU-SCE-M-4	**0.467**	**0.468**	**0.467**	**12**	**14**	**12**
NTNU	0.470	0.456	0.462	13	12	17
BGU-SCE-M-3	**0.461**	**0.460**	**0.460**	**14**	**16**	**15**
LCS-IESI	0.461	0.456	0.458	15	15	18
UA-DLSI	0.457	0.456	0.456	17	18	16
Lead	0.425	0.434	0.429	20	24	20

Table 3: MSS task. English.

system	P score	R score	F score	P-Rank	R-Rank	F-Rank
CCS	0.202	0.213	0.207	1	1	1
BGU-SCE-M-1	**0.196**	**0.210**	**0.203**	**2**	**2**	**2**
BGU-SCE-M-5	**0.193**	**0.208**	**0.200**	**3**	**3**	**3**
SCE-P	0.189	0.203	0.196	4	4	6
EXB	0.186	0.205	0.195	5	5	4
Oracles	0.182	0.204	0.192	6	6	5
Lead	0.168	0.178	0.173	12	13	12
LCS-IESI	0.181	0.170	0.172	13	7	14

Table 4: MSS task. Hebrew.

system	P score	R score	F score	P-Rank	R-Rank	F-Rank
Oracles	0.630	0.658	0.644	1	1	1
BGU-SCE-M-5	**0.562**	**0.569**	**0.565**	**2**	**4**	**2**
CCS	0.554	0.571	0.562	4	3	3
EXB	0.546	0.571	0.558	8	2	7
SCE-P	0.545	0.560	0.552	10	9	9
LCS-IESI	0.540	0.527	0.531	11	13	12
Lead	0.524	0.535	0.529	13	12	13

Table 5: MSS task. Arabic.

As can be seen, MUSE (basic) outperformed all other systems participating in MultiLing 2015 except for CCS in Hebrew. CCS (the CCS-5 submission, to be precise) uses the document tree structure of sections, subsections, paragraphs, and sentences, and compiles a summary from the leading sentences of recursive bottom-up interweaving of the node leading sentences, starting from leaves (usually, paragraphs in a section).

Three submissions–M-2, M-3, and M-5–were also provided to evaluation for English documents, and one–M-5–for Hebrew and Arabic in the multi-document summarization (MMS) task. MUSEEC scored the first place on Hebrew and the second place on English and Arabic languages, out of 9 participants.

We have also compared the performance of POS-based features with 31 MUSE language-independent features. In our experiments, we have used POS features along with five structural sentence features that are a subset of MUSE features and which include sentence length and sentence position metrics. In the evaluation, we used DUC-2002 (DUC, 2002) and MultiLing 2015 training data in English. According to the 10-fold cross validation results, the POS–based features significantly outperform the basic MUSE in terms of ROUGE-1 recall (p-value of 0.009 and 0.044, respectively with the Wilcoxon test).

It is noteworthy that the POS and structural features combination contains fewer features than the basic MUSE. Additionally, POS features require a smaller computational effort compared to the baseline (because they exclude SVD, Page Rank, and other features, which are relatively time-consuming).

In the MultiLing 2015 contest, one of our submissions (BGU-SCE-M-2) was based on POS and structural features (as described above). We trained MUSEEC on the provided training set, in order to evaluate the feature weights. However, we used POS patterns relevance values calculated from the DUC-2002 dataset due to the time constraints. The competition results, based on the Gold Standard summaries, indicate that, in contrast to our evaluation results, based on the training data, the original MUSE features outperform the novel POS features by 1.5% (in terms of the F-measure). This fact can be explained, partially, by not using the original data for obtaining POS patterns relevance measures and by the relatively small size of the dataset used for training (30 articles only).

4 Conclusions and Future Work

In this paper, we present an extractive summarization system, called MUSEEC, based on a supervised optimization of a weighted linear combination of multiple sentence features. We also introduce new POS-based features for extending the existing MUSE algorithm and adapting it to a particular language. The MultiLing 2015 automatic evaluation results show that MUSE outperforms other participating systems on English and Arabic corpora, and all systems, except one, on the Hebrew corpus in the MSS task. In the MMS task, on the other hand, MUSE outperforms all systems on the Hebrew corpus, and all systems, except one, on English and Arabic corpora. Contrary to our previous evaluation results, the competition results have not revealed any superiority of the POS-based features over the language-independent features used by the basic version of MUSE.

Acknowledgments

This work was partially funded by the U.S. Department of the Navy, Office of Naval Research.

References

Rafeeq Al-Hashemi. 2010. Text summarization extraction system (tses) using extracted keywords. *Int. Arab J. e-Technol.*, 1(4):164–168.

P. B. Baxendale. 1958. Machine-made index for technical literature – an experiment. *IBM Journal of Research and Development*, 2(4):354–361.

S.T. Davis, J.M. Conroy, and J.D. Schlesinger. 2012. OCCAMS – An Optimal Combinatorial Covering Algorithm for Multi-document Summarization. In *Proceedings of the IEEE 12th International Conference on Data Mining Workshops*, pages 454–463.

DUC. 2002. Document Understanding Conference. http://duc.nist.gov.

H. P. Edmundson. 1969. New Methods in Automatic Extracting. *ACM*, 16(2):264–285.

J. Goldstein, M. Kantrowitz, V. Mittal, and J. Carbonell. 1999. Summarizing Text Documents: Sentence Selection and Evaluation Metrics. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 121–128.

Vishal Gupta and Gurpreet Singh Lehal. 2010. A survey of text summarization extractive techniques. *Journal of Emerging Technologies in Web Intelligence*, 2(3):258–268.

F. J. Kallel, M. Jaoua, L. B. Hadrich, and A. Ben Hamadou. 2004. Summarization at LARIS Laboratory. In *Proceedings of the Document Understanding Conference*.

Eugene Krapivin. 2014. JRouge—Java ROUGE Implementation. https://bitbucket.org/nocgod/jrouge/wiki/Home.

M. Last and M. Litvak. 2012. Cross-lingual training of summarization systems using annotated corpora in a foreign language. *Information Retrieval*, pages 1–28, September.

Jure Leskovec, Natasa Milic-Frayling, and Marko Grobelnik. 2005. Impact of linguistic analysis on the semantic graph coverage and learning of document extracts. pages 1069–1074.

C.Y. Lin and E. Hovy. 1997. Identifying topics by position. In *Proceedings of the fifth conference on Applied natural language processing*, pages 283–290.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using N-gram co-occurrence statistics. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 71–78.

Chin-Yew Lin. 2004. ROUGE: A Package for Automatic Evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out (WAS 2004)*, pages 25–26.

Christina Lioma and Roi Blanco. 2009. Part of speech based term weighting for information retrieval. In *Advances in information retrieval*, pages 412–423. Springer.

Marina Litvak, Mark Last, and Menahem Friedman. 2010. A new approach to improving multilingual summarization using a Genetic Algorithm. In *ACL '10: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 927–936.

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

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

C. Nobata, S. Sekine, M. Murata, K. Uchimoto, M. Utiyama, and H. Isahara. 2001. Sentence Extraction System Assembling Multiple Evidence. In *Proceedings of 2nd NTCIR Workshop*, pages 319–324.

Dragomir Radev, Sasha Blair-Goldensohn, and Zhu Zhang. 2001. Experiments in single and multidocument summarization using MEAD. In *Proceedings of the First Document Understanding Conference (DUC)*.

Horacio Saggion, Kalina Bontcheva, and Hamish Cunningham. 2003. Robust generic and query-based summarisation. In *EACL '03: Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics*, pages 235–238.

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

S. Teufel and M. Moens. 1997. Sentence Extraction as a Classification Task. In *Proceedings of the Workshop on Intelligent Scalable Summarization, ACL/EACL Conference*, pages 58–65.

L. Vanderwende, H. Suzuki, C. Brockett, and A. Nenkova. 2007. Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing and Management*, 43(6):1606–1618.