EASY-M: Evaluation System for Multilingual Summarizers

Marina Litvak, Natalia Vanetik and Yael Veksler
Department of Software Engineering
Shamoon Engineering College
Beer Sheva, Israel
{marinal, natalyav, yaelva}@sce.ac.il

Abstract

Automatic *text summarization* aims at producing a shorter version of a document (or a document set). Evaluation of summarization quality is a challenging task. Because human evaluations are expensive and evaluators often disagree between themselves, many researchers prefer to evaluate their systems automatically, with help of software tools. Such a tool usually requires a point of reference in the form of one or more human-written summaries for each text in the corpus. Then, a system-generated summary is compared to one or more human-written summaries, according to selected measures (also called *metrics*). However, a single metric cannot reflect all quality-related aspects of a summary. In this paper we present the **E**v**A**luation **SY**stem for **M**ultilingual Summarization (EASY-M), which enables the evaluation of system-generated summaries in 23 languages with several quality measures, based on comparison with their human-generated counterparts. The system also provides comparative results with two built-in baselines. The EASY-M system is freely available for the NLP community[1].

1 Introduction

Automatic *text summarization* aims at representing a text document or a document set in a short concise form, called a *summary*. The size of a summary is usually limited by a user-defined number of words, sentences, or percentage of the original text. A summary can be either generic or tailored to fit the user's needs. The former is expected to convey the meaning of the whole text while the latter should reflect the interests of a user. Expressions of the user's interests can come in many forms, including those of query, subject, and style. Several extensive surveys of automatic summarization can be found in (Nenkova et al., 2011; Nenkova and McKeown, 2012; Das and Martins, 2007; Lloret and Palomar, 2012).

Automatic text summarization approaches can be divided into two main categories. *Extractive summarization* (Gupta and Lehal, 2010; Gambhir and Gupta, 2017) deals with selecting a subset of sentences from the original document(s) without modifying them. *Abstractive summarization* can compile summaries by extracting parts of original sentences (this approach is known as compressive summarization (Gambhir and Gupta, 2017)), or by generating new, original sentences. (Kasture et al., 2014)

The need for quality assessment of summarization tools is obvious. Using human evaluators is extremely time-consuming and labor-intensive. Additional issues arise when using this approach, such as the qualification of evaluators and their agreement on a content of generated summaries. (Pittaras et al., 2019) Also, hiring qualified evaluators to work with summaries in multiple languages is not an easy and often tedious task. Therefore, there is an existing need to construct automatic summary evaluation tools that provide consistent results for multiple languages. Moreover, these tools must provide a wide range of metrics for covering multiple aspects of summary quality, such as the informativeness, coverage of the main topics of a document, and the coherency and readability of the summary.

In this paper we introduce an evaluation system we have named EASY-M: **E**v**a**luation **SY**stem for

[1] https://drive.google.com/file/d/1GKeJiHCAxA8fKEBpi424nmVDIHGYWKSt/view?usp=sharing

Proceedings of the Multiling 2019 Workshop, co-located with the RANLP 2019 conference, pages 53–62
Varna, Bulgaria, September 6, 2019.
http://doi.org/10.26615/978-954-452-058-8_008

Multilingiual Summarization. We have designed EASY-M for evaluation of summarization results and ranking summarization tools on multiple languages. At its current state, the system enables the user to select a language and to evaluate the quality of generic summaries using several metrics that address both informativeness and readability of summaries. EASY-M also enables users to compare the scores of evaluated summaries to corresponding scores of summaries that were produced by two baseline methods, one of which produces 'ideal' extractive summaries. By doing so, the system gives the user an idea of how far current summaries lie from the best result that can be possibly achieved by extractive summarization. EASY-M also enables the user to view the correlation between scores of different metrics with Spearman correlation.

This paper is organized as follows. Section 2 surveys related work. Section 3 describes the summarization metrics used by and the baseline summarizers implemented in EASY-M. Section 4 shows and explains system's interface. Section 5 addresses the system's availability. Finally, Section 6 concludes our work.

2 Related Work

Multiple MultiLing reports (Giannakopoulos et al., 2011, 2015, 2017) give a detailed description of evaluating multiple summarization systems in different languages for various tasks. These evaluations utilized several measures including ROUGE (Lin, 2004) and MeMoG (Giannakopoulos and Karkaletsis, 2011) for automatic evaluation of summarization systems. Both tools were applied separately and autonomously, after their adaptation to multiple languages. This experience demonstrates the actual need in the multilingual evaluation system that can be applied once on the summaries generated by different systems and rank them based on various scores measuring different summary qualities.

2.1 Automatic evaluation

Automatic evaluation relies on comparison between the summaries generated by an automatic system (*system summaries*) and summaries that have been produced by humans (called *gold standard summaries* or *reference summaries*). Reference summaries may be created from scratch by humans or produced by merging several human-produced summaries by using the majority rule (Nanba and Okumura, 2000). In both cases, reference summaries usually contain new sentences that are not present in original documents. When reference summaries are not available, system summaries may be compared to original texts through the use of a metric that helps to see how information in the whole text is covered by a summary (Jing et al., 1998). Results of automatic evaluation depend closely on the chosen metric.

2.2 Evaluation metrics

Papers (Jones and Galliers, 1995) and (Jing et al., 1998) contain surveys of early evaluation measures for text summarization. Paper (Mani, 2001) gives an overview of different methods for evaluating automatic summarization systems, and describes different evaluation criteria such as coherence, informativeness, different scoring approaches, and means of analyzing summary content.

Following (Jones and Galliers, 1995) and (Steinberger and Ježek, 2012), summarization evaluation methods can be divided into two categories: *extrinsic* evaluation, where the summary quality is judged by its helpfulness for a given task, and *intrinsic* evaluation, where a summary is analyzed directly. Our study focuses on intrinsic evaluation of generic summaries (where no user queries are supplied).

2.3 Metric types

We can roughly assign all intrinsic evaluation methods to the (1) methods comparing between system and human summaries, and (2) the methods comparing between system summaries and their documents. The metrics provided in the first category measure the closeness (similarity) of the generated summary to reference summaries that represent the ideal summaries, while the metrics calculated in the second category measure the summary's coverage of the main topics described in a document. We will call the first category "similarity" and the second one "coverage." While the "similarity" methods can be performed in either the lexical (i.e., words) or semantic (i.e., topics) level, comparison between a summary and its document in the lexical level is meaningless. Therefore, for measuring coverage of topics in a generated summary, semantic text representation must be utilized.

2.4 Lexical similarity metrics

There are multiple metrics that compare between system and reference summaries in the lexical level. These metrics measure the similarity between vocabularies (Salton and McGill, 1986) of summaries. Some of them are applicable to extractive summarization only, such as metrics based on sentence recall or precision (Kupiec et al., 1995; Jing and McKeown, 1999; Merlino and Maybury, 1999), or metrics that rely on sentence rank (in terms of summary-worthiness); they measure the correlation between sentence sequences representing system and reference summaries (Donaway et al., 2000).

The Bleu machine translation evaluation measure (Papineni et al., 2002) has been used as a summarization metric in (Pastra and Saggion, 2003).

Metrics in the Recall-Oriented Understudy for Gisting Evaluation (ROUGE) family, proposed in (Lin, 2004), count the number of overlapping units such as n-grams, word sequences, and word pairs between the system and the reference summaries. This remains the most popular metric for summarization evaluation. In (Giannakopoulos and Karkaletsis, 2011), the authors present the Merge Model Graph (MeMoG) metric for evaluating summaries, which uses n-gram graphs for comparing system and reference summaries. Tests on summaries produced for MultiLing-2015 tasks (Giannakopoulos et al., 2015) have shown a clear indication that the MeMoG is much less sensitive than ROUGE to differences in text preprocessing. Both tools are also applicable to evaluation of abstractive summaries, but, as all lexical-based methods, they do not consider semantic similarity between system and reference summaries.

2.5 Semantic similarity metrics

An alternative solution to the lexical comparison between system and reference summaries is to consider their semantics. Utility-based metrics (Radev, 2000) use more fine-grained approach to measure importance of summary sentences; however, they increase the chances of disagreement between different evaluators. The Pyramid method discussed in (Nenkova and Passonneau, 2004) involves semantic matching of content units, to which differential weights are assigned based on their frequency in a corpus of summaries. Semantic models such as latent semantic analysis (LSA) (Deerwester et al., 1990), topic modeling with latent Dirichlet analysis (LDA) (Blei et al., 2003), word embeddings with Word2Vec (Mikolov et al., 2013), and Doc2Vec (Le and Mikolov, 2014) can also be used for comparing summaries to reference summaries or to original documents. In (Steinberger and Ježek, 2012) the authors propose an LSA-based evaluation measure and show its high correlation to human rankings. In (Ng and Abrecht, 2015) and (Kusner et al., 2015) word embeddings were shown as a good means for evaluating summaries.

2.6 Readibility and coherency metrics

A separate place in the world of summarization assessment metrics belongs to methods which address the linguistic quality of system-generated summaries rather than their contents. These metrics naturally depend on the language of summaries and cannot be called language-independent. We give a short description of the most popular metrics that are easy to implement with existing tools.

Proper noun ratio (PNR) is the ratio of proper nouns to the overall number of words in the summary. It is hypothesized that higher PNR indicates higher readability (Smith et al., 2012), because proper nouns contribute to a text disambiguation. Noun ratio (NR) is used to capture the proportion of nouns present in the text. The text with lower proportion of nouns is considered to be easier to read (Hancke et al., 2012). Pronoun ratio (PR) is a linguistic measure indicating the level of semantic ambiguity that can arise while searching for the concept that a pronoun represents (Štajner et al., 2012); a text with lower PR is considered more readable. The Gunning fog index (Gunning, 1952) is a readability test for English writing that gives a parametrized measurement of complex words in the text. Average word length (AWL) reflects the ratio of long words used in a text. It was proven that the use of long words makes a text more difficult to understand (Rello et al., 2013).

2.7 Evaluation systems

Attempts to create a platform for summary evaluation have been previously made. The SUMMAC system (Mani et al., 2002) provided the first system-independent framework for summary evaluation. It included several extrinsic and intrinsic methods for evaluating summaries. In the extrinsic categorization task, the evaluation was to determine whether a summary could effectively present enough information to categorize a document. In the extrinsic categorization task, an evaluation is made by finding whether there is enough information contained in a summary to provide successful categorization of the document. In an intrinsic question-answering task a topic-related summary for a document was evaluated in terms of its 'informativeness', namely, the degree to which it contained answers to a set of topic-related questions.

Paper (Hovy et al., 2006) described a framework in which various automated summary content evaluation methods can be situated, and implemented a specific variant that uses short text fragments. Multiple similarity metrics were introduced and their correlations with other known metrics, such as ROUGE, were reported. Most introduced metrics are lexical-based, except one that applied synonym resolution using WordNet. In (Abdi and Idris, 2014) the authors present a summarization assessment system that does not rely on reference summaries. There, a coverage metric was proposed as a combination of syntactic (words order) and semantic (using WordNet) information of sentence words.

Our system, EASY-M, provides different types of metric suitable for the multilingual domain and also supplies comparison to baselines, one of them being extractive topline summarizer.

3 System design

In this section we describe the capabilities of the EASY system and the algorithms it implements. The system receives the following input from the user.

1. A **folder containing original documents** in UTF-8 text format, where every document is stored in a separate file. In case of multi-document summarization, every document set should be merged into a single file.

2. A **folder containing reference summaries** should be available, with one or more summaries for every document. A document and its reference summaries are matched by their case-sensitive name parts before the file extension. Different reference summaries are distinguished by their first extension.

3. A **folder containing system summaries being evaluated**, with one summary for each document. A document and its summary are matched by a case-sensitive comparison of their name parts before file extension.

When input documents and summaries are supplied, the user first selects the language and summary size, then selects metrics (see Section 3.1) and their parameters. The pipeline of EASY-M is depicted in Figure 1. A detailed user story is described in Section 4.

3.1 Summarization quality metrics

In this section, we explain how summarization metrics are used in our system.

3.1.1 ROUGE metrics

Paper (Lin, 2004) presented set of metrics called ROUGE that is used for evaluating automatic summarization. ROUGE represents a set of similar metrics such as ROUGE-N, ROUGE-L, ROUGE-W, ROUGE-S, and ROUGE-SU. Its main idea is to count overlapping units (such as n-grams, word sequences and word pairs) between a system summary and reference summaries. Intuitively, higher ROUGE scores show that the system summary is of higher quality. This metric is currently the most popular metric of its type, especially in the field of text summarization (Cohan and Goharian, 2016).

In our system, we implemented several original ROUGE metrics and a new measure ROUGE-WSU, introduced in (Colmenares et al., 2015), as described below.

1. ROUGE-N, which measures overlap of n-grams between the system summary and reference summaries $R = \{r_1, \ldots, r_k\}$ with a user-defined n, that is usually set to a number between 1 and 4. EASY-M supports both recall- and precision-based ROUGE-N measure.

2. Common-subsequence-based metrics include the following

   (a) ROUGE-L, which measures the length of the longest common subsequence $LCS()$ between the system and reference summaries; this measure is an F-measure computed from LCS-based P_{LCS} precision and recall R_{LCS} as follows:

$$F_{LCS} = \frac{(1+\beta^2)R_{LCS}P_{LCS}}{R_{LCS}+\beta^2 P_{LCS}}$$

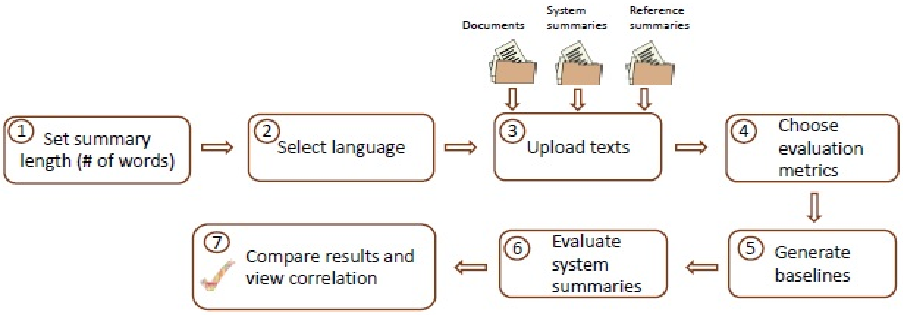

Documents
System summaries
Reference summaries
1 Set summary length (# of words)
2 Select language
3 Upload texts
4 Choose evaluation metrics
5 Generate baselines
6 Evaluate system summaries
7 Compare results and view correlation

Figure 1: EASY-M system flow.

where β is the system parameter with default $\beta = 1$ (to obtain a harmonic mean).

$$P_{LCS} = \frac{\sum_{i=1}^{k} LCS(r_i, S)}{|S|}$$

$$R_{LCS} = \frac{\sum_{i=1}^{k} LCS_{\cup}(r_i, S)}{\sum_{i=1}^{k} |r_i|}$$

Here,

$$LCS_{\cup}(r_i, S) = \cup_{j=1\ldots m} LCS(r_i, s_j)$$

where $s_1, \ldots, s_m$ are the sentences of S.

(b) ROUGE-W (Lin, 2004), which measures the length of the longest weighted common subsequence and differentiates subsequences by their length. It is an F-measure F_{WLCS} of ROUGE-W precision and recall computed as:

$$R_{WLCS} = f^{-1}(\frac{WLCS(S, R}{f(|S|)})$$

$$P_{WLCS} = f^{-1}(\frac{WLCS(S, R}{f(|r_1| + \cdots + |r_k|)})$$

Function $f()$ is smooth with a smooth inverse, and is usually set to $f(k) = k^2$ so that $f^(- 1)(k) = \sqrt{k}$. Parameter β is set to 1 (Sasaki et al., 2007).

3. Skip-based metrics

(a) ROUGE-S measures the overlap of skip-bigrams between a candidate summary and a set of reference summaries. It is similar to ROUGE-2 except that a skip-bigram refers to any pair of words in sentence order that allows for arbitrary gaps. The precision and recall are computed as a ratio of the total number of possible bigrams.
Let $SKIP2(S, r_i)$ denote the number of skip-matches between system summary S and reference summary r_i. Then ROUGE-S is defined as an F-measure R_{SKIP2} based on precision and recall on skip-bigrams where

$$R_{SKIP2} = \frac{SKIP2(S, R)}{C(|S|, 2)}$$

$$P_{SKIP2} = \frac{SKIP2(S, R)}{C(|r_1 + \cdots + r_k|, 2)}$$

and $C(x, 2)$ is the total possible number of bigrams. The maximum skip distance between two words is limited by the maximum distance parameter $d_{MAX-SKIP}$ to be 4, so that skip-bigrams are taken into account within the maximum skipping distance only.

(b) ROUGE-SU measures overlaps of both skip-bigrams and unigrams between a candidate summary and a set of reference summaries. This is because we do not want to assign a 0 score to a candidate summary simply because it does not share a skip bigram with any reference summary when instead it has a common unigram. Therefore, unigrams are added to give credit to the candidate's summary if it does not contain any pair of words with the reference summary.

(c) ROUGE-WSU weights skip-bigrams with respect to their average skip-distance. This overcomes the main problem of ROUGE-SU that gives the same importance to all skip-bigrams extracted from a phrase.

3.1.2 MeMoG metric

The MeMoG metric, presented in (Giannakopoulos and Karkaletsis, 2011), is an evaluation method that based on n-gram graphs. Experimental proof of its high performance for evaluation of summaries in different languages is presented in (Giannakopoulos et al., 2015).

Given a set of reference summaries, the MeMoG metric creates an n-gram graph for each of them and an n-gram graph for the system summary. Formally, let $G = \{V, E, W\}$ be an n-gram graph, where V is the set of character n-grams that can be created from the text, E is the set of edges, and W is the weight function that represents the number of times a pair of n-grams is present in a text within a legal distance from each other. This distance is denoted D_{win}. In order to compute this metric, the user should supply the following parameters:

1. L_{min} - minimum length of n-grams,
2. L_{max} - maximum length of n-grams, and
3. D_{win} - the windows size for two n-grams.

The default parameters are $L_{min} = 3$, $L_{max} = 3$ and $D_{win} = 3$, following (Giannakopoulos and Karkaletsis, 2011). The next step is to represent all reference summaries by a single n-gram graph. We begin by initializing the graph to be an n-gram graph of any of the reference summaries. The initial graph is then updated using every one of the remaining n-gram reference summary graphs as follows. Let G_1 be the current merged n-gram graph, and let G_2 be the n-gram graph of the next reference summary. The *merge function* $U(G_1, G_2, l)$ defined edge weights as

$$w(e) = w^1(e) + (w^2(e) - w^1(e)) * l$$

where $l \in [0, 1]$ is the learning factor, $w^1(e)$ is the weight of e in G_1, and $w^2(e)$ is the weight of e in G_2. In our system we chose $l = \frac{1}{i}$ where $i > 1$ is the number of the reference graph being processed. In the MeMoG metric, the score of a summary is one similarity measurement, denoted by VS, between system summary graph G^j and the merged reference graph G^i. The similarity score between edges is defined as

$$VR(e) = {}^{\min\{w^i(e), w^j(e)\}}/_{\max\{w^i(e), w^j(e)\}}$$

where w^i and w^j are weights of the same edge e (identified by its end-node labels) in graphs G^i and G^j respectively. The final score is computed as

$$VS(G^i, G^j) = {}^{\sum VR(e)}/_{\max\{|G^i|, |G^j|\}}$$

3.2 Topic coverage metrics

Topic model is a type of statistical model for discovering the abstract topics that occur in a collection of documents. Latent Dirichlet allocation (LDA) (Blei et al., 2003; Blei, 2012) allows documents to have a mixture of topics. LDA uses a generative probabilistic approach for discovering the abstract topics, (i.e., clusters of semantically coherent documents). As a result, we assume that every word w in document D is assigned its probability distribution $\{p_{w,T_i}\}$ over topics $T_1, \ldots, T_K$ where K is the number of topics supplied as a user-defined parameter. Then for a system summary S we can naturally define topic similarity to document (TSD) and topic similarity to reference summary (TSR) metrics as follows:

1. For every word w, its topic $T(w)$ is set to be $T(w) = \arg\max_i p_{w,T_i}$.
2. A text is represented by topic vector $TV = (p_{w,T_i})_w$ of word topics; if word w is not present in the text, $TV[w] = 0$.
3. Topic similarity between document D and system summary S is computed as cosine similarity $TSD(D, S) = \cos(TV_S, TV_D)$ between their topic vectors.
4. Topic similarity between system summary S and reference summaries $r_1, \ldots, r_n$ is computed as maximal cosine similarity between their topic vectors: $TSR(r_1, \ldots, r_n, S) = \max_i \cos(TV_S, TV_{r_i})$

3.3 Readability metrics

In our system we implemented proper noun ratio (PNR), noun ratio (NR), pronoun ratio (PR), and average word length (AWL) metrics. Currently, these metrics are supported for the English language only.

3.4 Baselines

3.4.1 TopK baseline

For this baseline, we simply select the first K sentences of the source document so that the number of words of the candidate summary is at least the predefined word limit W, making K minimal.

3.4.2 OCCAMS baseline

The OCCAMS, introduced in (Davis et al., 2012), is an algorithm for selecting sentences from a source document when reference summaries are known. This algorithm finds the best possible sentence subset covering reference summaries because reference summaries are visible to it. While no extractive summary can fully match human-generated abstractive reference summaries, OCCAMS achieves the best possible result (or its good approximation) for the extractive summarization task. Comparing system summaries to the result of OCCAMS shows exactly how far the tested system is from realistic best possible extractive summarization result.

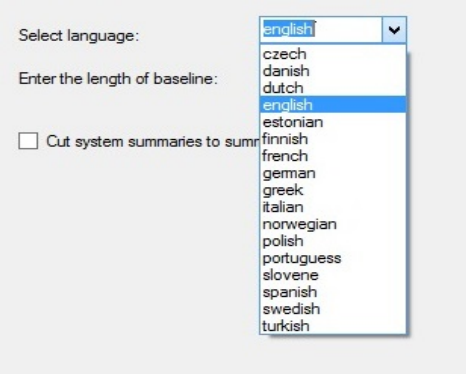
The OCCAMS' parameters are the weights of the terms W, the number of words in sentences C, and the size of the candidate summary L. Let D be the source document consisting of sentences $S_1, \ldots, S_n$ and let $T = \{t_1, \ldots, t_m\}$ be the set of document's terms (tokenized stemmed words). Initially OCCAMS computes document matrix A using term-to-sentence assignment and term entropy weights. Then, OCCAMS computes the singular value decomposition of matrix A as $A = USV^T$, following the approach of (Steinberger and Ježek, 2004). The singular value decomposition produces term weights $w(t_i)$. Then, the final solution is computed by using Budgeted Maximum Coverage (BMC) from (Khuller et al., 1999) and Fully Polynomial Time Approximation Scheme (FPTAS) of (Karger, 2001) greedy algorithms. These algorithms select sentences that provide maximum coverage of the important terms (maximum weight sum), while ensuring that their total length does not exceed the intended summary size.

4 Implementation details

In this section we describe and give examples of the EASY system interface. [2]

4.1 Operational pipeline

The first screen of the system (see Figure 2) asks the user to choose language and to set the summary length (if a summary is too long, it will be cut to the given number of words).



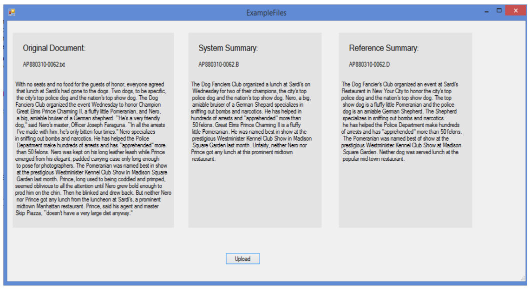
Select language: english

Enter the length of baseline:

□ Cut system summaries to sumr

Figure 2: Choosing language.

In our system, a user can make a choice between analyzing a single file with its system and reference summaries, or analyzing an entire corpus. The user needs to supply file names for the document (or directory of documents), reference summary (or summaries) or reference summaries directory, and the system summary or their directory that is to be evaluated. File names are treated as case-sensitive. Figure 3 shows the input selection interface for the case of a corpus.



Original Document:

AP880310-0062.txt

With no seats and no food for the guests of honor, everyone agreed that lunch at Sardi's had gone to the dogs. Two dogs, to be specific, the city's top police dog and the nation's top show dog. The Dog Fanciers Club organized the event Wednesday to honor Champion Great Elms Prince Charming II, a fluffy little Pomeranian, and Nero, a big, amiable bruiser of a German shepherd. ``He's a very friendly dog,'' said Nero's master, Officer Joseph Faraguna. ``In all the arrests I've made with him, he's only bitten four times.'' Nero specializes in sniffing out bombs and narcotics. He has helped the Police Department make hundreds of arrests and has ``apprehended'' more than 50 felons. Nero was kept on his long leather leash while Prince emerged from his elegant, padded carrying case only long enough to pose for photographers. The Pomeranian was named best in show at the prestigious Westminster Kennel Club Show in Madison Square Garden last month. Prince, long used to being coddled and primped, seemed oblivious to all the attention until Nero grew bold enough to prod him on the chin. Then he blinked and drew back. But neither Nero nor Prince got any lunch from the luncheon at Sardi's, a prominent midtown Manhattan restaurant. Prince, said his agent and master Skip Piazza, ``doesn't have a very large diet anyway.''

System Summary:

AP880310-0062.B

The Dog Fanciers Club organized a lunch at Sardi's on Wednesday for two of their champions, the city's top police dog and the nation's top show dog. Nero, a big, amiable bruiser of a German Shepard specializes in sniffing out bombs and narcotics. He has helped in hundreds of arrests and "apprehended" more than 50 felons. Great Elms Prince Charming II is a fluffy little Pomeranian. He was named best in show at the prestigious Westminister Kennel Club Show in Madison Square Garden last month. Unfairly, neither Nero nor Prince got any lunch at this prominent midtown restaurant.

Reference Summary:

AP880310-0062.D

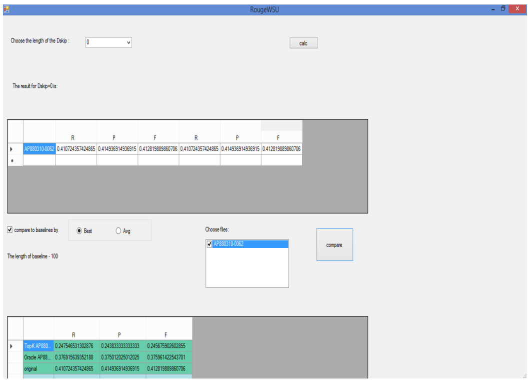
The Dog Fancier's Club organized an event at Sardi's Restaurant in New Your City to honor the city's top police dog and the nation's top show dog. The top show dog is a fluffy little Pomeranian and the police dog is an amiable German Shepherd. The Shepherd specializes in sniffing out bombs and narcotics. he has helped the Police Department make hundreds of arrests and has "apprehended" more than 50 felons. The Pomeranian was named best of show at the prestigious Westminister Kennel Club Show at Madison Square Garden. Neither dog was served lunch at the popular mid-town restaurant.

Figure 3: Choosing texts, reference and system summary.

Figures 4, 5 and 6 show results of computation for ROUGE, MeMoG, and topic summarization metrics and readability metrics for the selected input. Note that readability analysis is currently supported for the English language only. The top part of the interface in both cases enables the user to select parameters for every metric, while the bottom

[2] Screen images are taken from standalone implementation. Web implementation with partial features is also available at `https://summaryevaluation.azurewebsites.net/home`.

part gives the user an opportunity to compute baseline summaries and to compute the chosen metric for baselines with the same parameters as above.



Choose the length of the Dskip : 0

The result for Dskip=0 is:

	R	P	F	R	P	F
AP880310-0062	0.410724357424865	0.414936914936915	0.412819889860706	0.410724357424865	0.414936914936915	0.412819889860706

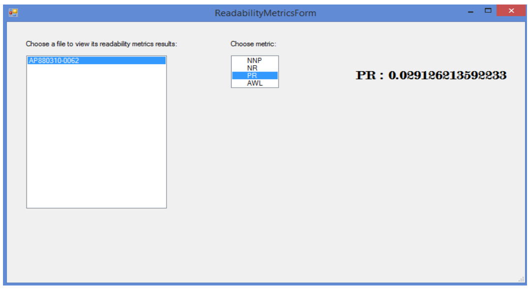
☑ compare to baselines by ● Best ○ Avg

The length of baseline - 100

Choose files:
☑ AP880310-0062

	R	P	F
TopK AP880...	0.247546531302876	0.243833333333333	0.245675902602855
Oracle AP88...	0.376915639352188	0.375012025012025	0.375961422543701
original	0.410724357424865	0.414936914936915	0.412819889860706

Figure 4: Rouge metrics computation with comparison to baselines.

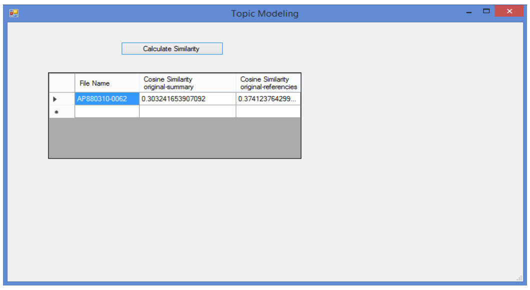


Choose a file to view its readability metrics results:
AP880310-0062

Choose metric:
NNP
NR
PR
AWL

PR : 0.029126213592233

Figure 5: Readability metrics computation.



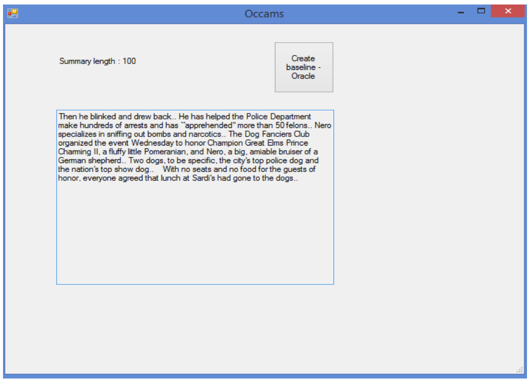
	File Name	Cosine Similarity original-summary	Cosine Similarity original-referencies
	AP880310-0062	0.303241653907092	0.374123764299...

Figure 6: Topic metrics computation.

Figure 7 shows baseline summary computed by the OCCAMS algorithm.

5 Availability and reproducibility

The EASY-M system standalone version is implemented in c#, and its Web version is implemented in Angular7 on the client side, and sp.net WebAPI2 on the server side. Video of the standalone interface operation is available at https://www.youtube.com/watch?v=HQhzhSQ7O1A&t=143s. Currently, the system supports 17 languages: Czech, Danish, Dutch, English, Estonian, Finnish, French, German, Greek, Italian, Norwegian, Polish, Portuguese, Slovene, Spanish, Swedish and Turkish.



Summary length : 100

Then he blinked and drew back.. He has helped the Police Department make hundreds of arrests and has ``apprehended'' more than 50 felons.. Nero specializes in sniffing out bombs and narcotics.. The Dog Fanciers Club organized the event Wednesday to honor Champion Great Elms Prince Charming II, a fluffy little Pomeranian, and Nero, a big, amiable bruiser of a German shepherd.. Two dogs, to be specific, the city's top police dog and the nation's top show dog.. With no seats and no food for the guests of honor, everyone agreed that lunch at Sardi's had gone to the dogs..

Figure 7: OCCAMS summary.

6 Conclusions

In this paper we present a multilingual framework named EASY-M for evaluation of automatic summarization systems. Currently, EASY-M supports 17 different languages. The system enables the users to compute several summarization metrics, including readability measures (English only), for the same set of summaries and to observe how they correlate with each other using Spearsman's correlation.

In our future work we plan to implement additional metrics based on word embeddings, and to add more languages by employing language specific tokenizer tools. We also plan to implement additional baseline methods. We will allow several systems to be compared and ranked simultaneously.

References

Asad Abdi and Norisma Idris. 2014. Automated summarization assessment system: quality assessment without a reference summary. In *The International Conference on Advances in Applied Science and Environmental Engineering-ASEE*.

D M Blei. 2012. Probabilistic topic models. *Communications of the ACM* 55(4):77–84.

D. M. Blei, A. Y. Ng, and M. I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3:993–1022.

Arman Cohan and Nazli Goharian. 2016. Revisiting summarization evaluation for scientific articles. *arXiv preprint arXiv:1604.00400* .

Carlos A Colmenares, Marina Litvak, Amin Mantrach, and Fabrizio Silvestri. 2015. Heads: Headline generation as sequence prediction using an abstract feature-rich space. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. pages 133–142.

Dipanjan Das and André FT Martins. 2007. A survey on automatic text summarization. *Literature Survey for the Language and Statistics II course at CMU* 4:192–195.

Sashka T Davis, John M Conroy, and Judith D Schlesinger. 2012. OCCAMS–an optimal combinatorial covering algorithm for multi-document summarization. In *Data Mining Workshops (ICDMW), 2012 IEEE 12th International Conference on*. IEEE, pages 454–463.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science* 41(6):391–407.

Robert L Donaway, Kevin W Drummey, and Laura A Mather. 2000. A comparison of rankings produced by summarization evaluation measures. In *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic summarization*. Association for Computational Linguistics, pages 69–78.

Mahak Gambhir and Vishal Gupta. 2017. Recent automatic text summarization techniques: a survey. *Artificial Intelligence Review* 47(1):1–66.

George Giannakopoulos, John Conroy, Jeff Kubina, Peter A Rankel, Elena Lloret, Josef Steinberger, Marina Litvak, and Benoit Favre. 2017. Multiling 2017 overview. In *Proceedings of the MultiLing 2017 workshop on summarization and summary evaluation across source types and genres*. pages 1–6.

George Giannakopoulos, Mahmoud El-Haj, Benoit Favre, Marina Litvak, Josef Steinberger, and Vasudeva Varma. 2011. TAC 2011 MultiLing pilot overview. In *Proceedings of Text Analytics Conference*. TAC.

George Giannakopoulos and Vangelis Karkaletsis. 2011. AutoSummENG and MeMoG in evaluating guided summaries. In *Proceedings of Text Analysis Conference*.

George Giannakopoulos, Jeff Kubina, John Conroy, Josef Steinberger, Benoit Favre, Mijail Kabadjov, Udo Kruschwitz, and Massimo Poesio. 2015. Multiling 2015: multilingual summarization of single and multi-documents, on-line fora, and call-center conversations. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. pages 270–274.

Robert Gunning. 1952. *The technique of clear writing*. McGraw-Hill, New York.

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.

Julia Hancke, Sowmya Vajjala, and Detmar Meurers. 2012. Readability classification for german using lexical, syntactic, and morphological features. *Proceedings of COLING 2012* pages 1063–1080.

Eduard Hovy, Chin-Yew Lin, Liang Zhou, and Junichi Fukumoto. 2006. Automated summarization evaluation with basic elements. In *Proceedings of the Fifth Conference on Language Resources and Evaluation (LREC 2006)*. Citeseer, pages 604–611.

Hongyan Jing, Regina Barzilay, Kathleen McKeown, and Michael Elhadad. 1998. Summarization evaluation methods: Experiments and analysis. In *AAAI symposium on intelligent summarization*. Palo Alto, CA, pages 51–59.

Hongyan Jing and Kathleen R McKeown. 1999. The decomposition of human-written summary sentences. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pages 129–136.

Karen Sparck Jones and Julia R Galliers. 1995. *Evaluating natural language processing systems: An analysis and review*, volume 1083. Springer Science & Business Media.

David R Karger. 2001. A randomized fully polynomial time approximation scheme for the all-terminal network reliability problem. *SIAM review* 43(3):499–522.

NR Kasture, Neha Yargal, Neha Nityanand Singh, Neha Kulkarni, and Vijay Mathur. 2014. A survey on methods of abstractive text summarization. *Int. J. Res. Merg. Sci. Technol* 1(6):53–57.

Samir Khuller, Anna Moss, and Joseph Seffi Naor. 1999. The budgeted maximum coverage problem. *Information processing letters* 70(1):39–45.

Julian Kupiec, Jan Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, pages 68–73.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International Conference on Machine Learning*. pages 957–966.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International Conference on Machine Learning*. pages 1188–1196.

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.

Elena Lloret and Manuel Palomar. 2012. Text summarisation in progress: a literature review. *Artificial Intelligence Review* 37(1):1–41.

Inderjeet Mani. 2001. Summarization evaluation: An overview. In *NAACL 2001 Workshop on Automatic Summarization*.

Inderjeet Mani, Gary Klein, David House, Lynette Hirschman, Therese Firmin, and Beth Sundheim. 2002. SUMMAC: a text summarization evaluation. *Natural Language Engineering* 8(1):43–68.

Andrew Merlino and Mark Maybury. 1999. *An empirical study of the optimal presentation of multimedia summaries of broadcast news*. Cambridge, MA: MIT Press.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. pages 3111–3119.

Hidetsugu Nanba and Manabu Okumura. 2000. Producing more readable extracts by revising them. In *Proceedings of the 18th conference on Computational linguistics-Volume 2*. pages 1071–1075.

Ani Nenkova and Kathleen McKeown. 2012. A survey of text summarization techniques. In *Mining text data*, Springer, pages 43–76.

Ani Nenkova, Kathleen McKeown, et al. 2011. Automatic summarization. *Foundations and Trends in Information Retrieval* 5(2–3):103–233.

Ani Nenkova and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the human language technology conference of the north american chapter of the association for computational linguistics: Hlt-naacl 2004*.

Jun-Ping Ng and Viktoria Abrecht. 2015. Better summarization evaluation with word embeddings for rouge. *arXiv preprint arXiv:1508.06034* .

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, pages 311–318.

Katerina Pastra and Horacio Saggion. 2003. Colouring summaries bleu. In *Proceedings of the EACL 2003 Workshop on Evaluation Initiatives in Natural Language Processing: are evaluation methods, metrics and resources reusable?*. Association for Computational Linguistics, pages 35–42.

Nikiforos Pittaras, Stefano Montanelliy, George Giannakopoulos, Alfio Ferraray, and Vangelis Karkaletsis. 2019. Crowdsourcing in single-document summary evaluation: the argo way. In Marina Litvak and Natalia Vanetik, editors, *Multilingual Text Analysis: Challenges, Models, and Approaches*, World Scientific, chapter 8.

Dragomir R Radev. 2000. Summarization of multiple documents: clustering, sentence extraction, and evaluation. In *Proceedings of the Workshop on Automatic Summarization, 2000*. Association for Computational Linguistics.

Luz Rello, Ricardo Baeza-Yates, Laura Dempere-Marco, and Horacio Saggion. 2013. Frequent words improve readability and short words improve understandability for people with dyslexia. In *IFIP Conference on Human-Computer Interaction*. Springer, pages 203–219.

Gerard Salton and Michael J McGill. 1986. *Introduction to modern information retrieval*. McGraw-Hill, Inc.

Yutaka Sasaki et al. 2007. The truth of the F-measure. *Teach Tutor mater* 1(5):1–5.

Christian Smith, Henrik Danielsson, and Arne Jönsson. 2012. A good space: Lexical predictors in vector space evaluation. In *Proceedings of the eighth international conference on Language Resources and Evaluation (LREC), Istanbul, Turkey*. Citeseer.

Sanja Štajner, Richard Evans, Constantin Orasan, and Ruslan Mitkov. 2012. What can readability measures really tell us about text complexity. In *Proceedings of workshop on natural language processing for improving textual accessibility*. Citeseer, pages 14–22.

Josef Steinberger and Karel Ježek. 2004. Text summarization and singular value decomposition. In *International Conference on Advances in Information Systems*. Springer, pages 245–254.

Josef Steinberger and Karel Ježek. 2012. Evaluation measures for text summarization. *Computing and Informatics* 28(2):251–275.