Using Machine Learning Methods and Linguistic Features in Single-Document Extractive Summarization

Alexander Dlikman and Mark Last

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

Abstract. Extractive summarization of text documents usually consists of ranking the document sentences and extracting the top-ranked sentences subject to the summary length constraints. In this paper, we explore the contribution of various supervised learning algorithms to the sentence ranking task. For this purpose, we introduce a novel sentence ranking methodology based on the similarity score between a candidate sentence and benchmark summaries. Our experiments are performed on three benchmark summarization corpora: DUC-2002, DUC-2007 and MultiLing-2013. The popular linear regression model achieved the best results in all evaluated datasets. Additionally, the linear regression model, which included POS (Part-of-Speech)-based features, outperformed the one with statistical features only.

Keywords: text summarization, part-of-speech tagging, supervised learning, regression, sentence ranking

1 Introduction

In this study, we seek to improve the performance of extractive summarization algorithms by using multiple statistical and linguistic sentence features combined with advanced machine learning techniques. We apply the following four supervised learning algorithms to the extractive summarization task: Classification and Regression Trees (CART) [3], Cubist [9], linear regression, and a genetic algorithm. The algorithms are trained on benchmark corpora of summarized documents and compared to state-of-the-art extractive summarization tools using the same feature sets. The proposed supervised methodology for sentence extraction is based on a continuous similarity score between candidate sentences and human-generated gold standard summaries. For this purpose, a novel, Penalized Precision metric is introduced.

In: P. Cellier, T. Charnois, A. Hotho, S. Matwin, M.-F. Moens, Y. Toussaint (Eds.): Proceedings of DMNLP, Workshop at ECML/PKDD, Riva del Garda, Italy, 2016.

2 Related Work

2.1 Extractive Text Summarization

Extractive summarization techniques identify the most important sentences in the input text(s) and combine them to create a summary of a pre-defined length. Various sentence scoring metrics, or features, have been proposed in literature. Gupta and Lehal [7] in their survey of text summarization techniques list the following groups of features: keyword-based, title-based, location-based, length-based, proper noun and upper-case word-based, font-based, specific phrase-based, and features based on the sentence similarity to other sentences in text. The MUSE summarization algorithm [15, 14] is a representative example of an extractive summarizer, built upon 31 statistical sentence metrics. These metrics are divided into *structure-based*, *vector-based* and *graph-based* groups. The MUSE summarizer uses a supervised approach with Genetic Algorithm to find the best feature weights from a given corpus of summarized documents.

Several extractive summarization approaches make use of linguistic sentence scoring metrics for text representation and calculation of the final sentence score. The most typical approach is the use of proper nouns or upper case words [7, 11, 12]. Fattah and Ren [5] use the count of numerical data and proper noun occurrences in a sentence. Al-Hashemi [2] employs human-generated rules based on POS (Part-of-Speech) sequences in an extractive summarization system. Mihalcea and Tarau [18] present a graph-based model for keyword extraction which makes use of POS tags. In this approach, a graph represents the text and interconnects words or other text entities. The authors propose several options including all words, only nouns, only nouns and verbs or only nouns and adjectives. One of conclusions of Mihalcea and Tarau's study shows that the performance of models without POS information is significantly lower than those that consider POS information.

2.2 Machine Learning Methods for Sentence Extraction

In the regression approach to the sentence ranking task, the score of each candidate sentence s is evaluated as a weighted average of all its features [20]. The feature weights can be found by various machine learning techniques such as a linear regression [5] or a Genetic Algorithm [14]. Ouyang et al. [21] apply a Support Vector Regression (SVR) model to the task of query-based, multi-document extractive summarization. Their SVR framework is based on a set of seven sentence features. Galanis et al. [6] present an Integer Linear Programming (ILP) based approach for extractive, query-based multi-document summarization. The proposed method simultaneously maximizes both the importance of the sentences that are included in a summary as well as their diversity. In order to find a sentence's importance score sentence, the authors use SVR model based on five various predictors (sentence features). The "true" importance (outcome of the regression) is obtained as a ROUGE score between candidate sentences and human-generated summaries.

Compared to other regression-based summarization methods that use seven predictive features in [21] and five in [6], we employ a much larger set of sentence scoring

metrics (30 statistical features from [14] and 17 novel linguistic features) and perform feature selection to preserve the most important features in the model. In addition, both [21] and [6] utilize a sentence-to-summary similarity score, which prefers the longest sentences in the extraction stage. The sentence-to-summary similarity score proposed in our study (Penalized Precision) handles this limitation and penalizes both "too short" and "too long" sentences.

3 Methodology

3.1 Linguistic features

In this section, we introduce 17 POS-based sentence features, which are listed in **Table 1**. Some of them are completely novel while others are derived from our interpretation of certain metrics used by Litvak and Last [14] in the MUSE summarizer. All proposed POS features take into account only nouns, verbs, adjectives and adverbs due to the semantic importance of these parts of speech [13]. These features can be divided into *POS ratio-based* (defined as a ratio between the number of the above parts-of-speech in a sentence and the sentence length); *POS filtering* (employing the original MUSE features after keeping the above POSs and discarding the rest of the words); and *POS patterns* (these features take into account part-of-speech 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 do not need further explanation, the POS pattern metrics are defined as follows. We assume that the presence of a specific POS pattern in a candidate sentence may indicate sentence relevance in the summary [2]. Our method requires a preprocessing stage where the relevance of the candidate POS patterns is calculated. We define POS pattern relevance as a ratio between normalized pattern frequency in human-generated summaries and normalized pattern frequency in the corpus. The measure is greater than one when the POS n-gram is relatively more frequent in summaries than in the 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- POS grams.

3.2 Sentences Ranking

Our methodology for the sentence ranking task includes the following steps: data preparation, calculation of sentence similarity to benchmark summaries, data scaling, training, and evaluation.

Data Preparation: In the data preparation stage, we generate a sentence-feature matrix for the training corpus. Each row of the matrix refers to a sentence i; each column refers to a feature; and entry of the matrix (m_{ij}) indicates the score of feature j for sentence i.

Each sentence is associated with $sentence\_ID$ and $document\_ID$. The feature set includes the original, language-independent MUSE features as well as our novel linguistic features.

Category	Feature	Description
POS Ratio-Based	POS_NN_RATIO	Ratio of *nouns* to all words in the sentence
	POS_VB_RATIO	Ratio of *verbs* to all words in the sentence
	POS_JJ_RATIO	Ratio of *adjectives* to all words in the sentence
	POS_RB_RATIO	Ratio of *adverbs* to all words in the sentence
POS Filtering	POS_V_TITLE_O	Overlap similarity to the document title
	POS_V_TITLE_J	Jaccard similarity to the document title
	POS_V_TITLE_C	Cosine similarity to the document title
	POS_V_TF	Average term frequency for all POS words
	POS_V_COV	Coverage of POS keywords
	POS_V_TFISF	Sum of term frequencies times inverse sentence frequencies
	POS_V_KEY	Sum of POS keyword frequencies
	POS_V_D_COV_O	Overlap similarity to the document complement
	POS_V_D_COV_J	Jaccard similarity to the document complement
	POS_V_D_COV_C	Cosine similarity to the document complement
POS Patterns	POS_N2	POS 2-gram relevance measure
	POS_N3	POS 3-gram relevance measure
	POS_N4	POS 4-gram relevance measure

Table 1. Part-of-Speech features

Sentence to Summary Similarity Score: The most complex stage is determining the similarity between each sentence and a gold standard summary of the corresponding document. Similarity measures such as ROUGE and other recall-based measures, which normalize joint terms between sentence and benchmark summaries by a summary length, prefer longer sentences by assigning them a higher score. On the other hand, precision-based measures, which normalize joint terms by sentence length, prefer shorter sentences.

To address those issues, we have modified the *BLEU (Bilingual Evaluation Understudy)* measure, which originally was used for evaluating the quality of machine translation [22]. Our implementation of the BLEU score (Eq. 1) is precision penalized when a sentence is "too short".

$$PenPr = P * pen$$

$$pen = \begin{cases} 1 & if\ length(s) > min.length \\ e^{1-\frac{min.length}{length(s)}} & if\ length(s) \leq min.length \end{cases} \quad (1)$$

P stands for the sentence precision, which naturally penalizes "too long" sentences as well, and the $min.length$ parameter represents the minimum sentence length in a gold standard summary. When several benchmark summaries exist per each document, we calculate the $PenPr$ value for each benchmark summary separately and then provide the average similarity of a sentence to benchmark summaries, exactly as in the ROUGE method.

Data Scaling: The max-min rescaling method is used to normalize the feature values to the [0, 1] range based on their minimum and maximum values in the training corpus. In contrast, to normalize the values of sentence similarity to the gold standard, we calculate the minimum and maximum similarity values separately for each document. This approach allows to deal with the fact that gold standard summaries in the corpus can be both extractive and abstractive (for extractive summaries, the similarity values tend to be higher than for the abstractive ones).

Training: By using the columns in the sentence-feature matrix as regression predictors and sentence similarity to the gold standard as a continuous target variable, any regression algorithm can be trained. The resulting regression model will include the values of the feature weights.

Evaluation. To evaluate the performance of the induced model on a hold-out set, we first compute the predicted value of each sentence similarity score ($\hat{y}$). After this, n top ranking sentences (based on $\hat{y}$) are extracted to a peer summary, subject to a summary length constraint. The resulting peer summaries can be evaluated using various ROUGE measures and available gold standard summaries.

4 Evaluation Experiments

4.1 Datasets and Software Tools

For training and testing, we used three different English corpora containing summarized documents. *DUC-2002* [4], which was prepared for the summarization competition task at the Document Understanding Conference, is a gold-standard dataset that contains 531 news articles from the Wall Street Journal (1987-1992), and the Financial Times (1991-1994). Each textual document contains at least 10 sentences and appears with two to three human-generated ("gold standard") abstractive summaries of around 100 words.

An additional evaluated corpus is *DUC-2007* [4]. The main task of DUC-2007 was, given a topic and a set of 25 relevant documents, to synthesize a fluent, well-organized

250-word summary of the documents that would answer the question in the topic statement, i.e., perform a multi-document query-based summarization. Each topic is accompanied with up to four human-generated abstractive summaries of around 250 words. In order to allow single-document training, all documents on a particular topic were merged into one text.

We have also used an English corpus from the MultiLing 2013 single-document summarization task [19]. The dataset includes 30 Wikipedia articles with one gold standard (human-generated) summary of around 270 words per article. Due to relatively small amount of documents, MultiLing-2013 is used only as test data in cross-corpus evaluation experiments.

In our study, we used MUSEEC, an open-source text summarization tool [16]. For the purpose of preprocessing (sentence splitting, tokenization, stop words removal and lemmatization) and part-of-speech tagging, we used the popular *Stanford CoreNLP toolkit* [17], an extensible pipeline that provides core natural language analysis. For sentence ranking, we used several R packages: *GA* Package [23] for Genetic Algorithm, *rpart* [24] for CART algorithm, *cubist* [10] for Cubist algorithm. The *Caret* R package [8] was used for parameter optimization of those algorithms and cross-validation when implementing the experiments described below.

4.2 Evaluation Results

We evaluated four regression approaches to the sentence ranking task: CART [3], LM (linear regression model), GA (Genetic Algorithm) and Cubist [9]. We also compared the results to MUSE [14] as a state-of-the-art supervised method for extractive summarization. Each model was evaluated with four different feature sets: *MUSE* (30 original features used by MUSE); *POS only* (17 POS-based features); *POS Extended* (17 POS-based features + Sentence Position + Sentence Length); and *MUSE & POS* (both MUSE and POS-based features).

DUC-2002 (10-fold cross-validation): Cubist and LM using the most complete feature set (MUSE & POS) were the top ranking approaches. Since the difference between them was not found statistically significant (p-value of 0.205) we preferred the simpler LM approach. In further statistical tests, we compared LM models with different feature sets (the first four rows in **Table 2**). As can be seen from the results, the MUSE & POS feature combination is significantly better than the other feature sets. The subsequent experiments (the last three rows in **Table 2**) compared the LM model with three other models (all using MUSE & POS features). The results are statistically significant and show that LM outperforms all other models. Using the Akaike Information Criterion (AIC) statistics [1] for stepwise feature selection, 4 statistical features (D_COV_J, KEY_DEG, KEY_PR, SVD) and 4 POS-based features (POS_B, POS_RB_RATIO, POS_V_TITLE_C, POS_V_TITLE_O) were discarded as statistically insignificant.

DUC-2007 (10-fold cross-validation): In this dataset, the difference between the MUSE and the MUSE & POS feature sets was not found statistically significant and, thus, the MUSE feature set was preferred due to simplicity. The experiments have shown that the LM model with MUSE features outperforms all other models with the same feature set.

MultiLing-2013 (training on DUC-2002): In the MultiLing-2013 corpus, both Cubist and LM with the MUSE & POS feature set are the top-ranking models, without a statistically significant difference between them. Consequently we prefer the simpler LM approach. The results show that LM with the MUSE & POS feature set outperforms all other models.

Model	Features	ROUGE-1 F	p-value
LM	MUSE & POS	0.464	--
LM	POS Extended	0.460	0.031
LM	MUSE	0.457	0.000
LM	POS only	0.454	0.001
MUSE	MUSE & POS	0.457	0.003
GA	MUSE & POS	0.452	0.000
CART	MUSE & POS	0.444	0.000

Table 2. DUC-2002 results with different feature sets

5 Conclusions

In this work, we have explored the contribution of various machine learning algorithms to sentence ranking and introduced a novel, Penalized Precision metric. The results of our experiments show that in all evaluated textual corpora, the linear model outperforms the more sophisticated CART and Cubist regression models, the heuristic optimization with genetic algorithm, as well as the state-of-the-art summarization approach (MUSE). Additionally, the linear models which included POS features, outperform those with statistical features only. To achieve the best results, we suggest using the Linear Model with statistical and POS-based features. Future work may focus on extending the proposed POS-based features and sentence ranking techniques to other languages and domains.

6 References

1. Akaike, H. 1974. A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control* 19 (6): 716-723.
2. Al-Hashemi, R. 2010. Text Summarization Extraction System (TSES) Using Extracted Keywords. *International Arab Journal of e-Technology* 1 (4): 164-168.
3. Breiman, L., Friedman, J., Stone, C., and Olshen, R. 1984. *Classification and regression trees*. CRC press.
4. *Document Understanding Conferences*. http://duc.nist.gov/.

5. Fattah, M. A., and Ren, F. 2009. GA, MR, FFNN, PNN and GMM based models for automatic text summarization. *Computer Speech & Language* 23 (1): 126-144.
6. Galanis, D., Lampouras, G., and Androutsopoulos, I. 2012. Extractive Multi-Document Summarization with Integer Linear Programming and Support Vector Regression. *COLING 2012: Technical Papers.* Mumbai,India. 911-926.
7. Gupta, V., and Lehal, G. 2010. A Survey of Text Summarization Extractive Techniques. *Journal of Emerging Technologies in Web Intelligence* 2 (3): 258-268.
8. Kuhn, M. 2015. *caret: Classification and Regression Training.* http://CRAN.R-project.org/package=caret.
9. Kuhn, M., and Johnson, K. 2013. *Applied predictive modeling.* New York: Springer.
10. Kuhn, M., Weston, S., Keefer, C., and Coulter, N. 2014. *Cubist: Rule- and Instance-Based Regression Modeling.* http://CRAN.R-project.org/package=Cubist.
11. Kupiec, J., Pedersen, J., and Chen, F. 1995. A trainable document summarizer. *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval.* 68-73.
12. Kyoomarsi, F., Khosravi, H., and Eslami, E. 2008. Optimizing Text Summarization Based on Fuzzy Logic. *Seventh IEEE/ACIS International Conference on Computer and Information Science (icis 2008).* 347-352.
13. Lioma, C., and Blanco, R. 2009. Part of Speech Based Term Weighting for Information Retrieval. In *Advances in Information Retrieval*, 412-423. Springer Berlin Heidelberg.
14. Litvak, M., and Last, M. 2013. "Cross-lingual training of summarization systems using annotated corpora in a foreign language." *Information retrieval* 16 (5): 629-656.
15. Litvak, M., Last, M., and Friedman, M. 2010. A new approach to improving multilingual summarization using a genetic algorithm. *48th Annual Meeting of the Association for Computational Linguistics.* 927-936.
16. Litvak, M., Vanetik, N., Last, M., and Churkin, E. 2016. MUSEEC: A Multilingual Text Summarization Tool. to appear in *54th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.
17. Manning, C., Surdeanu, M., and Bauer, J. 2014. The Stanford CoreNLP Natural Language Processing Toolkit. *52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations.* 55-60.
18. Mihalcea, R., and Tarau, P. 2004. TextRank: Bringing order into texts. *Conference on Empirical Methods in Natural Language Processing.* Barcelona, Spain: Association for Computational Linguistics. 404–411 .
19. *MultiLing Community Site.* http://multiling.iit.demokritos.gr/.
20. Nenkova, A., and McKeown, K. 2012. A survey of text summarization techniques. In *Mining Text Data*, 43-76. Springer US.
21. Ouyang, Y., Li, W., Li, S., and Lu, Q. 2011. Applying regression models to query-focused multi-document summarization. *Information Processing & Management* 47 (2): 227-237.
22. Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. 2002. BLEU: a method for automatic evaluation of machine translation. *40th Annual Meeting of the Association for Computational Linguistics.* 311-318.
23. Scrucca, L. 2013. GA: A Package for Genetic Algorithms in R. *Journal of Statistical Software* 53 (4): 1-37.
24. Therneau, T., Atkinson, B., and Ripley, B. 2015. *rpart: Recursive Partitioning and Regression Trees.* http://CRAN.R-project.org/package=rpart.