Supervised Keyphrase Extraction Leveraging Candidate Clustering [DRAFT PAPER]

Adrien Bougouin
LINA – UMR CNRS 6241
Université de Nantes, France
adrien.bougouin@univ-nantes.fr

Akiko Aizawa
National Institute of Informatics
Tokyo, Japan
aizawa@nii.ac.jp

Abstract

In this paper, we present a supervised extension of the topic ranking keyphrase extraction method, TopicRank. Experiments show that combining common features and features related to the topical clusters significantly improves TopicRank and outperforms the most popular supervised keyphrase extraction method, KEA.

1 Introduction

Since the last decade, the amount of information available on the web is constantly increasing. While the number of documents continues to grow, the need for efficient information retrieval methods becomes increasingly important. One way to improve retrieval effectiveness is to use keyphrases (Jones and Staveley, 1999). Keyphrases are single or multiword expressions that represent the main content of a document. However, only a small number of documents have keyphrase metadata. Keyphrase extraction has then attracted a lot of attention recently and many different approaches were proposed (Hasan and Ng, 2014).

Generally speaking, keyphrase extraction methods can be categorized into two main categories: supervised and unsupervised approaches. Supervised approaches treat keyphrase extraction as a binary classification task, where each phrase is labeled either as “keyphrase” or “non-keyphrase”. Many supervised methods have been proposed, applying classifiers, such as Naive Bayes classifiers (Witten et al., 1999), SVMs (Zhang et al., 2006) and multilayer perceptrons (Sarkar et al., 2010), with various features, such as the first position (Witten et al., 1999), document sections (Nguyen and Kan, 2007) and known keyphrase distributions (Frank et al., 1999). Conversely, unsupervised approaches usually rank phrases by importance and select the top-ranked ones as keyphrases. Unsupervised approaches proposed so far have involved a number of techniques including clustering (Liu et al., 2009), graph-based ranking (Mihalcea and Tarau, 2004) and even both (Bougouin et al., 2013).

In the current state of the keyphrase extraction task, supervised methods outperform unsupervised methods. Taking advantage of features extracted from training documents paired with reference keyphrases, supervised methods are able to identify among candidate keyphrases the ones most likely to be keyphrases. In opposition, unsupervised methods only rely on the document to analyse (sometimes using extra information) in order to determine the importance of each candidate keyphrase in regard of the document.

In order to extract the most likely keyphrases without neglecting their actual importance within the analysed document, we present a work that combines both supervised and unsupervised visions of the keyphrase extraction task. Using the unsupervised keyphrase extraction method TopicRank (Bougouin et al., 2013), we group candidate keyphrases that represent the same topic, determine the most important topics and apply machine learning to extract the most likely keyphrase of each important topic. Our experiments show that a supervised method can benefit from features related to topical clusters. Our method significantly outperforms TopicRank and the most popular supervised method, KEA.

2 TopicRank

TopicRank aims to extract keyphrases that best represent the main topics of a document using both candidate clustering and graph-based ranking.

At first, TopicRank clusters candidate keyphrases that belongs to the same topic. This topical clustering uses a "naive" stem overlap similarity: at the beginning, each candidate is a single cluster and candidates sharing an average of $1/4$ stemmed words with the candidates of a given cluster are added to this cluster.

Secondly, TopicRank builds a graph of topics and ranks the topic using TextRank (Mihalcea and Tarau, 2004). Every topic is connected to other topics by edges weighted according to the semantics strength between the connected topics. Then, TextRank ranking algorithm gives high importance to topics strongly connected to as most topic as possible and, additionally, important topics contribute more to the importance of the topics they are strongly connected to.

At last, TopicRank extract keyphrases among the candidates of the N most important topics. To avoid topic redundancy, only one keyphrase per topic is extracted. Following previous observations, the strategy to extract the best keyphrase of a topic takes its candidate that appears first in the document. This strategy is currently the weakest link of TopicRank. Bougouin et al. (2013) showed that the best best possible performance of TopicRank (with a "perfect" strategy) is be much higher than its current performance.

3 Supervised Keyphrase Selection from Topical Clusters

Leveraging TopicRank's topic clustering and ranking, we propose a supervised method that extracts keyphrases from the important topics. To stay consistent with Bougouin et al. (2013), we extract only one keyphrase per cluster. Instead of being the first appearing candidate keyphrase, the keyphrase we extract is the one that our trained classifier considers most appropriate.

In this work, we decide to use an SVM classifier. SVMs are very efficient classifiers. They support a large number of features and learn to weight them appropriatelty. Also, the SVM classifier is the best performing classifier in our approach.

3.1 Training samples

In order to better fit our objective, we select positive and negative examples from a set of relevant clusters instead of a set of all clusters. We consider relevant every cluster from which it is actually possible to discriminate one reference keyphrase out of multiple candidate keyphrases. As they do not fit this situation, we exclude clusters containing either a single candidate or only negative examples.

3.2 Features

Previous work using machine learning selected features that helps to determine the keyphrase likelyhood of a candidate. In our work, we also rely on such common features, but we also rely on new features related to the appropriateness of a candidate as a member of the topic. In contrast, we call the first category of features "topically independent features" and the second category "topically dependent features".

3.2.1 Topically Independent Features

Following previous work (Nguyen and Kan, 2007; Lopez and Romary, 2010), we use the length of a candidate (number of words) as a feature, along with other structural and distributional features.

Structural features Previous work showed that position-based features are good indicator of the keyphrase likelyhood of a candidate. Among others, Witten et al. (1999) used the position of the first occurrence of the candidate, while most recently Nguyen and Kan (2007) used a vector of frequencies within sections of scientific papers (abstract, introduction, conclusions, etc.). In this work, we use four structural features: the position of the first occurrence of the candidate and three binary features if the candidate appears, respectively, in the first, second or last third of the document. These last features approximate the feature of Nguyen and Kan (2007). However, they are not restricted to scientific papers.

Distributional features We use three features that relies on candidate distributions to: determine their importance (TF-IDF, cf. equation 1), the lexical cohesion of their words[1] (GDC, cf. equation 2) (Lopez and Romary, 2010) and their relevancy as keyphrases. First introduced by Frank et al. (1999), the relevancy of a candidate as a keyphrase is simply the number of times the candidate is used as a keyphrase in the training corpus.

$$\text{TF-IDF}(c, d) = \frac{\text{count}(c, d)}{|d|} \times \frac{-\log_2 |\{d \in D : c \in d\}|}{|D|} \quad (1)$$

$$\text{GDC}(c, d) = \frac{|c| \times \text{count}(c, d) \times \log_{10} \text{count}(c, d)}{\sum_{w \in c} \text{count}(w, d)} \quad (2)$$

where D is the set of every document of a global corpus and count(c, d) is the number of occurrences of the candidate c in the document d.

3.2.2 Topically Dependent Features

Topically dependent features help to capture the appropriateness of a candidate as the representant of its topic. The clustering strategy of Bougouin et al. (2013) is to add candidates to a cluster if their average stem overlap similarity with the candidates of the cluster is above a given threshold. We decide to use this average similarity as a feature. However, Bougouin et al. (2013) explained that their clustering similarity is naive for the purpose of topic clustering. Therefore, we also use the average number of candidates (in the cluster) that do not share any word with the candidate. As a consequence, our topically dependent features represent both the similarity and the disimilarity of the candidate regarding the other candidates of the cluster.

3.3 Keyphrase Extraction

To extract keyphrsaes, we propose to apply the trained SVM classifier on TopicRank's important topics to choose one keyphrase per each.

3.3.1 Keyphrase Selection

For each cluster built by TopicRank, we apply the SVM classifier to determine its best keyphrase. The best keyphrase of a cluster is the one that have the highest confidence overall candidates of the cluster. SVM's confidence score is given by the distance between the candidate and the separating hyperplane: the higher is the distance beetween the candidate and the hyperplane, the more confidence the candidate gets.

3.3.2 Keyphrase Re-Ranking

To rank the keyphrases, our method benefits from both topic ranking and machine learning. The ranking score is obtained as follow:

$$S(c) = \alpha \times \text{topicrank}(c) + (1 - \alpha) \times p(c) \quad (3)$$

where topicrank(c) is the importance score given by TopicRank for the cluster to which candidate c belongs and $p(c)$ is the probability that candidate c is a keyphrase according to the SVM classifier. As SVMs do not provide probability scores, we use the Platt scaling (Platt, 1999) to obtain $p(c)$. Finally, we empiracally set α to 0.75, meaning that more importance is given to TopicRank's score.

4 Experimental Settings

4.1 Dataset

In this work, we use the SemEval corpus. Built for the task 5 of SemEval-2010 (Kim et al., 2010), SemEval contains 244 English scientific papers collected from the ACM Digital Libraries. We use SemEval's training set (144 documents) and test set (100 documents) with their sets of combined author- and reader-assigned keyphrases.

4.2 Baselines

In order to show that our method benefits from all aspects of its configuration, we design a set of baselines that slightly diverge from our method (deried baselines). First, TopicRank plus the SVM classifier trained on either topically independent and dependent features (TopicRank+SVM), while the SVM classifier is trained on all features for our method (TopicRank+SVM$_{all}$). Second the SVM classifier, trained on either topically independent, dependent or all features, is applied to the unranked clusters (Clustering+SVM). Finally, the SVM classifier, trained on topically independent features, is applied candidate keyphrases (SVM).

For comparison purpose, we also report results of a Naive Bayes classifier trained with the first position and the TF-IDF features (Witten et al., 1999, KEA), TF-IDF and TopicRank

[1] We set a default value to single word candidates (0.5 for nouns and 0.0 for adjectives).

4.3 Preprocessing

For our method, as well as all baselines, we use TopicRank's outputs. Therefore, our results can directly be compared to results in (Bougouin et al., 2013).

4.4 Evaluation Measures

We evaluate the performances of our method and the baselines in terms of precision (P), recall (R) and f-score (f1-measure, F) when at most 10 keyphrases are extracted. In order to reduce mismatches due to flexions such as plural, we also stem candidate and reference keyphrases during the evaluation.

5 Results

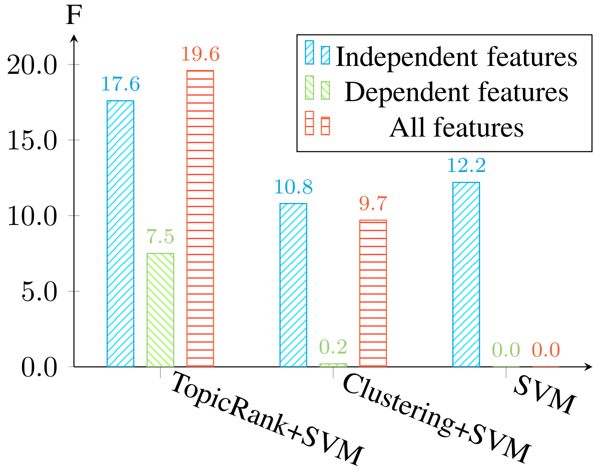

Figure 1: Performance of TopicRank+SVM$_{all}$ compared to derived baselines.

Method	P	R	F
KEA	18.8	13.3	15.4
TF-IDF	13.2	8.9	10.5
TopicRank	14.9	10.3	12.1
TopicRank+SVM$_{all}$	24.2	16.7	19.6
TopicRank$_{max}$	37.6	25.8	30.3

Table 1: Performance of TopicRank+SVM$_{all}$ compared to previous work.

6 Conclusion and Future Work

In this paper, we presented a supervised extension of TopicRank. Using the SemEval corpora, we showed that combining common features and features related to the topical clusters significantly improves TopicRank and outperforms the most popular supervised method, KEA.

This work showed that using clusters of topics ranked by importance can improve the performance of supervised methods. In future work, we plan to improve TopicRank's clustering and to explore topic labelling methods (Mei et al., 2007; Lau et al., 2011).

References

Adrien Bougouin, Florian Boudin, and Béatrice Daille. 2013. Topicrank: Graph-Based Topic Ranking for Keyphrase Extraction. In *Proceedings of the 6th International Joint Conference on Natural Language Processing (IJCNLP)*, pages 543–551, Nagoya, Japan, October. Asian Federation of Natural Language Processing.

Eibe Frank, Gordon W. Paynter, Ian H. Witten, Carl Gutwin, and Craig G. Nevill Manning. 1999. Domain-Specific Keyphrase Extraction. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence - Volume 2*, pages 668–673, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Kazi Saidul Hasan and Vincent Ng. 2014. Automatic Keyphrase Extraction: A Survey of the State of the Art. In *Proceedings of the Association for Computational Linguistics (ACL)*, Baltimore, Maryland, June. Association for Computational Linguistics.

Steve Jones and Mark S. Staveley. 1999. Phrasier: a System for Interactive Document Retrieval Using Keyphrases. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 160–167, New York, NY, USA. ACM.

Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. 2010. SemEval-2010 task 5: Automatic Keyphrase Extraction from Scientific Articles. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval)*, pages 21–26, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jey Han Lau, Karl Grieser, David Newman, and Timothy Baldwin. 2011. Automatic Labelling of Topic Models. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1536–1545, Portland, Oregon, USA, June. Association for Computational Linguistics.

Zhiyuan Liu, Peng Li, Yabin Zheng, and Maosong Sun. 2009. Clustering to Find Exemplar Terms for Keyphrase Extraction. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1 (EMNLP)*, pages 257–266, Stroudsburg, PA, USA. Association for Computational Linguistics.

Patrice Lopez and Laurent Romary. 2010. HUMB: Automatic Key Term Extraction from Scientific Articles in GROBID. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 248–251, Uppsala, Sweden, July. Association for Computational Linguistics.

Qiaozhu Mei, Xuehua Shen, and ChengXiang Zhai. 2007. Automatic Labeling of Multinomial Topic Models. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 490–499. ACM.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing Order Into Texts. In Dekang Lin and Dekai Wu, editors, *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 404–411, Barcelona, Spain, July. Association for Computational Linguistics.

Thuy Dung Nguyen and Min-Yen Kan. 2007. Keyphrase Extraction in Scientific Publications. In *Proceedings of the 10th International Conference on Asian Digital Libraries: Looking Back 10 Years and Forging New Frontiers*, pages 317–326, Berlin, Heidelberg. Springer-Verlag.

John C Platt. 1999. Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. In *Advances in Large Margin Classifiers*.

Kamal Sarkar, Mita Nasipuri, and Suranjan Ghose. 2010. A New Approach to Keyphrase Extraction Using Neural Networks. *International Journal of Computer Science Issues Publicity Board 2010*.

Ian H. Witten, Gordon W. Paynter, Eibe Frank, Carl Gutwin, and Craig G. Nevill Manning. 1999. KEA: Practical Automatic Keyphrase Extraction. In *Proceedings of the 4th ACM Conference on Digital Libraries*, pages 254–255, New York, NY, USA. ACM.

Kuo Zhang, Hui Xu, Jie Tang, and Juanzi Li. 2006. Keyword Extraction Using Support Vector Machine. In *Proceedings of the 7th International Conference on Advances in Web-Age Information Management*, pages 85–96, Berlin, Heidelberg. Springer-Verlag.