Term Validation for Vocabulary Construction and Key Term Extraction

Alexander Ulanov
Hewlett-Packard Labs
alexander.ulanov@hp.com

Andrey Simanovsky
Hewlett-Packard Labs
andrey.simanovsky@hp.com

Abstract

We extract new terminology from a text by term validation in a dictionary. Our approach is based on estimating probabilities for previously unseen terms, i.e. not present in a dictionary. To do this we apply several probabilistic models previously not used for term recognition and propose a new one. We apply restriction of domain similarity on terms used for probability estimation and vary the parameters of the models. Performance of our approach is demonstrated using Wikipedia titles vocabulary.

1 Introduction

Keyphrase extraction or automatic term recognition is an important task in the area of information retrieval. It is used for annotating text articles, tagging documents, etc. Keyphrases facilitate easier searching, browsing documents, detecting topics, classification, adding contextual advertisement, and so on.

Current methods of term extraction rely either on statistics of terms inside documents or on external dictionaries. These approaches work relatively well with large texts and with specialized vocabularies. The problem arrives when a text contains a lot of cross-domain terms which are essential and vocabulary does not cover them. One option is to use several vocabularies: a very broad one, like Wikipedia or WordNet, and another one very specific, like Burton's legal thesaurus. Even in this case two types of terms will not be identified: new terms and term collocations. New terms appear in emerging areas, and established thesauri will not catch them. Term collocation means a specific term used in conjunction with a broad-sense term. Usually it is hard to automatically identify if collocation is a new term or not.

This paper addresses the problem of detecting new terms in a text that are missing in the dictionary in order to enrich it, or to create a new, domain-specific one.

2 State of the art

A comprehensive overview and comparison of automatic term recognition (ATR) methods is presented in (Zhang et al., 2008).

The generic approach includes chunking or POS-tagging, stop-word removal, and restricting candidate terms to phrases, usually noun-based (Frantzi and Ananiadou, 1999), (Wermter and Hahn, 2005). These candidates are ranked using word statistics or mappings to external dictionaries. Word statistics is used to calculate termhood and unithood. Termhood is a measure of term relevancy to the subject domain. Unithood is a measure of words cohesion in a term. Termhood is usually frequency-based, computed using plain TF or TF-IDF (Medelyan and Witten, 2006). Other approaches to termhood computation use a notion of weirdness (Ahmad et al., 2000), which is based on the term frequency in a different domain compared to the subject domain. It is extended to the notions of domain pertinence in (Sclano et al., 2007). In the work of (Wartena et al., 2010) term distributions are compared to background corpus as a measure of descriptiveness.

Dictionaries are used to verify that candidate terms cannot be split and POS tags are correct (Aubin and Hamon , 2006). Statistics across corpus can be combined with the values from the dictionary. Several measures of association strength (word cohesion) in bi-grams are inspected in this way (Fahmi et al., 2007). Mukherjea et al. (2004) use external dictionaries such as UMLS to learn typical term suffixes and affixes. Then they are used in patterns for terms extraction. The number of relations between found terms derived from thesauri is proposed to be used to-

gether with the term frequency as a ranking function in (Gazendam et al., 2010). Common terms dictionary is used in (OpenCalais, 2011) for term extraction.

The advantage of our approach is that it does not rely on terms frequency in a text. Instead it uses probabilistic model of a dictionary. The approach is beneficial when texts are rather small and where is the need to enrich a given dictionary. Our approach is more accurate comparing with the present works in which either patterns for finding terms are collected (Mukherjea et al., 2004) or any collocation with a dictionary term is considered as a new term (OpenCalais, 2011).

3 Proposed approach

We propose to detect new terminology with the use of models build on top of vocabularies. The question is how to do this since new terms are not present in vocabularies. We use language modeling approach and treat phrases as n-grams or sequences of tokens. We use bi-grams as approximation for phrases of other length for the sake of simplicity. All possible decompositions of phrases into two parts are considered.

There are several ways how to estimate the probability of unseen n-grams to be in a vocabulary. A straightforward way is redistribution of the probability mass via lower level conditional distributions:

$$P_{BO}(w_m/w_1^{m-1}) = \begin{cases} d_{w_1^m}\frac{c(w_1^m)}{c(w_1^{m-1})} & \text{if } c \geq k; \\ \alpha P_{BO}(w_m/w_1^{m-2}) & \text{otherwise} \end{cases},$$

where w_1^m is m-gram, c is the number of occurences (0 in our case), α is a normalizing constant, d is a probability discounting. In the back-off part this model doesn't address association strength between phrase tokens. This happens since it uses lower level conditional probabilities. This estimation is quite rough, at least for bi-grams. It happens because two words encountered separately may have extremely different meanings and frequencies as compared to when whey stand next to each other in a phrase. To cope with this problem, back-off model is updated with the notions of association strength and similarity restriction. The following smoothing model for bi-grams was proposed by Essen and Steinbiss (1992):

$$P_{SE}(w_2/w_1) = \sum_{w_1',w_2'} P(w_2/w_1')P(w_1'/w_2')P(w_2'/w_1),$$

where w_1 and w_1' are the first tokens, and w_2 and w_2' are the second tokens of bi-grams w_1w_2 and $w_1'w_2'$.

We also use the similarity model for bi-grams (Dagan et al., 1994):

$$P_{SD}(w_2/w_1) = \sum_{w_1' \in S(w_1)} P(w_2/w_1')\frac{W(w_1',w_1)}{\sum_{w_1' \in S(w_1)} W(w_1',w_1)},$$

where $W(w_1', w_1)$ is the weight that determines similarity between tokens w_1' and w_1.

In order to use both similarity and collocation strength we propose the following estimation for unobserved bi-grams in addition to the mentioned models (we will refer to it as "*C-Similarity*"):

$$P_{BS}(w_2/w_1) = \sum_{w_1',w_2'} P(w_2/w_1')P(w_2'/w_1),$$
$$S(w_1w_2', w_1'w_2) \geq S_{max}.$$

where S is the similarity function between bi-grams. The trivia behind this model is to find pairs of bi-grams that share common parts in the same places with unobserved ones. According to the similarity constraint, these bi-grams must be from the same domain.

4 Experiments

As we mentioned in the Introduction we believe that our model is preferrable among others in the case of short texts. The experimental setup was designed to test that hypothesis. We considered the extreme artificial scenario of texts composed of single phrases that should be either recognized as a term or not. We considered Wikipedia titles and their reversals as such collection of texts. Since Wikipedia editors aim at comprehensive coverage of all notable topics and are partial about including alternave lexical representations for them we can assume that if some reversal of a Wikipedia title is a term it should be present among Wikipedia titles. Thus, the titles and reversals collection could be correctly classified into terms and not terms by lookup into Wikipedia titles dictionary. We used that classification as a gold standard. The testing methodology included splitting the collection into training and test sets and measuring precision and recall of the models compared to the gold standard.

The mentioned term validation models were benchmarked using the discussed texts collection. We extracted all articles titles from the Wikipedia dump dated May 2010. Their total number is 8521847. Among them, there are 1567357 single word titles, 2928330 2-gram titles, and 1836494 3-gram titles. We filter out only 2-grams and 3-grams for the sake of simplicity [1].

The four above-mentioned models were used: back-off, smoothing, similarity, and co-similarity. For the similarity model we employed 2 different distance functions to compute W. The first is Kullback-Leibler distance D:

$$D(w_1 \| w_1') = \sum_{w_2} P(w_2/w_1) \log \frac{P(w_2/w_1)}{P(w_2/w_1')}.$$

This model is referred as "*Similarity-KL*". We also used:

$$W(w_1/w_1') = \sum_{w_2} P(w_2/w_1), w_2 : \\ \exists w_2' S(w_1 w_2', w_1' w_2) \geq S_{max}.$$

This model is referred as "*Similarity-S*".

Wikipedia category structure is employed to measure similarities S between terms. For each term we extracted a subset of 27 Wikipedia main topic categories (categories from "Category:Main Topic Classifications"). A certain category was assigned to a term if it was reachable from this category by browsing the category tree down looking in at most 8 intermediate categories. Similarity between two terms was measured as Jaccard coefficient between corresponding category sets:

$$S(term_1, term_2) = \frac{|Categories_1 \bigcap Categories_2|}{|Categories_1 \bigcup Categories_2|}.$$

This function is too rough for determining semantic similarity on the given set of categories. However it is a good and fast approximation for the domain similarity.

We conduct experiments to measure precision and recall of each term validation model. Wikipedia was split into two parts of equal size using modulo 2 for articles *id*'s. Such splitting can be considered pseudo-random because article *id*'s roughly correspond to the order in which articles were added to Wikipedia. One part was treated as a set of observed n-grams and was used to train the models. The other part was used as a gold standard.

[1] We treat n-grams as bi-grams/tri-grams. All possible decompositions of n-grams into two parts are considered.

We required a set on which the gold standard would be a good approximation of the desired behavior of the system. Namely, we needed a set that would be considerably larger than the set of Wikipedia titles, and at the same time contain phrases that are unlikely to become Wikipedia titles. We created such a set by uniting the gold standard 2-grams and 3-grams with their reversals. We rely on an assumption that the editors deliberately decide to include either both or just one of the terms "X Y" and "Y X" into Wikipedia. Thus, we were able to estimate how good the golden standard can be predicted by the model and how precise it is. Precision (P) was computed in the following way:

$$P = \frac{N_{G \cap V}}{N_V},$$

where $N_{G \cap V}$ is the number of validated n-grams from the golden standard and N_V is the number of n-grams validated by the model.

Recall (R) was computed as:

$$R = \frac{N_{G \cap V}}{N_G},$$

where N_G is the number of n-grams in the golden standard.

In our tests, n-grams were validated by our model if their probability estimation exceeded a particular threshold. It was chosen as a minimum non-null probability estimation for an unobserved n-gram.

The results of the experiments are represented in Table 1. Back-off stands for back-off model (P_BO). Smoothing stands for Essen and Steinbiss model (P_SE). Similarity-KL and Similarity-S are the variations of similarity model which we described earlier. C-Similarity stands for the proposed original model. In brief, incorporating semantic similarity into the model allows the extraction to perform significantly better. As one can see from the table, the back-off model is very volatile with respect to Wikipedia titles. For 2-grams its unigram setting provides too relaxed assumptions, while for 3-trams it starts to lack statistics. Smoothing removes volatility, but appears to be too restrictive. The reason is that it relies on observation of connecting $w_1 \prime w_2 \prime$ 2-gram (we refer here to the 2-gram case). If the observation probability is replaced with an arbitrary weight $0 \leq W(w_1 \prime w_2 \prime) \leq 1$, we will obtain generalization of Smoothing and C-Similarity (for

C-Similarity W gets the values of 0 and 1 depending on the similarity between the q-grams). The similarity that was used is less restrictive as a smoothing factor than the observation probability. It is reflected by C-Similarity having smaller precision and greater recall than Smoothing. To compare C-Similarity with the previous similarity model we considered two weighting schemes. Similarity-KL uses a common approach with Kullback-Leibler divergence. Lack of semantics similarity resulted in Similarity-KL performing worse than C-Similarity. In Similarity-S we incorporated semantic similarity knowledge into the previous similarity model. As one can see from the results, our C-Similarity and Similarity-S demonstrate comparable quality, Similarity-S working better with 2-grams and C-Similarity outperforming on 3-grams.

Table 1: Term validation experiments results.

	2-grams		**3-grams**	
Model	**P**	**R**	**P**	**R**
Back-off	0.51	0.69	0.93	0.44
Smoothing	**0.78**	0.28	**0.95**	0.28
Similarity-KL	0.58	0.68	0.81	0.54
Similarity-S	0.58	**0.79**	0.82	0.65
C-Similarity	0.62	0.67	0.83	**0.66**

5 Conclusion

We applied a range of probabilistic models for estimating probability of previously unseen terms to be a part of a dictionary. They use dictionary statistics as compared to current approaches that use corpus. We proposed an additional model. All these models have not been applied before in the field of term recognition. Our experiments showed their applicability in the task of finding new terminology.

Our plans are to conduct more experiments and to use n-grams of any size for validation of a particular n-gram (not only with the same number of words). Further work is connected with exploring various model restrictions that may allow raising recall. For example, we will use various similarity functions. We plan to incorporate term validation with keyphrase extraction techniques as well. Another interesting direction is to iteratively find new terms and update dictionaries.

Our ultimate goal is to build domain-specific dictionaries and determine the meaning of newly discovered terms.

Compiling comparable corpora might be another area of application of the proposed model.

Acknowledgments

The authors would like to thank Dr. Pankaj Mehra for valuable and inspiring discussions.

References

Khurshid Ahmad, Lee Gillam, and Lena Tostevin. 2000. Weirdness Indexing for Logical Document Extrapolation and Retrieval (WILDER) In (Eds.) E.M. Voorhees and D.K. Harman. *The 8th Text Retrieval Conference (TREC-8)*: 717-724.

Sophie Aubin and Thierry Hamon. 2006. Improving Term Extraction with Terminological Resources. *In Advances in Natural Language Processing (5th International Conference on NLP, FinTAL 2006)*: 380-387.

Ido Dagan, Fernando Pereira, and Lillian Lee. 1994. Similarity-based estimation of word cooccurrence probabilities. *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*: 272-278.

Ute Essen and Volker Steinbiss. 1992. Cooccurrence smoothing for stochastic language modeling. *IEEE International Conference on Acoustics, Speech, and Signal Processing,* 1:161-164.

Ismail Fahmi, Gosse Bouma, and Lonneke vd. Plas. 2007. Using Known Terms for Automatic Term Extraction. *Computational Linguistics in Nederland (CLIN).*

Katerina T. Frantzi and Sophia Ananiadou. 1999. The c/nc value domain independent method for multiword term extraction. *Journal of Natural Language Processing.*

Luit Gazendam, Christian Wartena, and Rogier Brussee. 2010. Thesaurus Based Term Ranking for Keyword Extraction. *In 7th International Workshop on Text-based Information Retrieval.*

Slava M. Katz. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35(3):400-401.

Olena Medelyan and Ian H. Witten. 2006. Thesaurus based automatic keyphrase indexing. *Proceedings of the 6th ACMIEEECS joint conference on Digital libraries JCDL 06.*

Sougata Mukherjea, L. Venkata Subramaniam, Gaurav Chanda, Sriram Sankararaman, Ravi Kothari, Vishal S. Batra, Deo N. Bhardwaj, and Biplav Srivastava.

2004. Enhancing a biomedical information extraction system with dictionary mining and context disambiguation. *IBM Journal of Research and Development,* 48(5-6): 693-702.

Reuters Thomson OpenCalais. 2011. `www.opencalais.com`.

Francesco Sclano and Paola Velardi. 2007. Termextractor: a web application to learn the shared terminology of emergent web communities. *In Proceedings of the 3rd International Conference on Interoperability for Enterprise Software andApplications (I-ESA 2007).*

Christian Wartena, Rogier Brussee, and Wout Slakhorst. 2010. Keyword Extraction using Word Co-occurrence. *In 7th International Workshop on Text-based Information Retrieval.*

Joachim Wermter and Udo Hahn. 2005. Finding new terminology in very large corpora. *Proceedings of the 3rd International Conference on Knowledge Capture (K-CAP 2005),* 137-144.

Ziqi Zhang, Jose Iria, Christopher Brewster, and Fabio Ciravegna. A Comparative Evaluation of Term Recognition Algorithms. *In The sixth international conference on Language Resources and Evaluation, (LREC 2008).*