A Keyphrase Extraction approach for Social Tagging Systems

Felice Ferrara and Carlo Tasso
Artificial Intelligence Laboratory
Department of Mathematics and Computer Science, University of Udine, Italy
{*felice.ferrara, carlo.tasso*}*@uniud.it*

Keywords: Keyphrase extraction, social tagging, keyphrase recommendation

Abstract: Social tagging systems allow people to classify resources by using a set of freely chosen terms named tags. However, by shifting the classification task from a set of experts to a larger and not trained set of people, the results of the classification are not accurate. The lack of control and guidelines generates noisy tags (i.e. tags without a clear semantics) which deteriorate the precision of the user generated classifications. In order to face this limitation several tools have been proposed in the literature for suggesting to the users tags which properly describe a given resource. In this paper we propose to suggest n-grams (named keyphrases) by following the idea that sequences of two/three terms can better face potential ambiguities. More specifically, in this work, we identify a set of features which characterize n-grams able to describe meaningful aspects reported in Web pages. By means of these features we developed a mechanism which can support people to manually classify Web pages by automatically suggesting meaningful keyphrases expressed in English.

1 INTRODUCTION

In this paper we propose an innovative content-based approach for automatic tag recommendation in social tagging systems. The keyphrase extraction mechanism proposed in this work opens many interesting perspectives for empowering the access to the knowledge stored in social tagging systems. The main one is related to the task of associating a tag to a specific semantic meaning: keyphrases extracted from a Web page can be used to identify concepts or entries defined in a semantic knowledge source such as Wordnet or Wikipedia.

By enriching the semantic value of tags the effectiveness of other applications can be improved as well. During the last ten years many recommender systems have been proposed to integrate tags in the process of modeling both the user interests and the resources available in social tagging systems. The main limitation of these approaches depends still on the fact that the meaning of a tag is usually inferred by taking into account only statistical information about the co-occurrences of tags. By disambiguating tags and enriching them with other semantic or ontological knowledge we can improve the accuracy of both collaborative filtering mechanisms and content based approaches.

The paper is organized as follows: the proposed approach to extract keyphrases from Web pages is illustrated in Section 2; Section 3 describes the evaluation settings and the results; final considerations conclude the paper in Section 4.

2 Extracting keyphrases from Web pages

By following the traditional schema adopted by several keyphrase extraction mechanisms we split the description of the approach into two parts: the candidate phrase extraction (Section 2.1) and the phrase selection phase (Section 2.2).

2.1 Candidate Phrase Identification

Given an HTML page, a **format conversion** step is exploited for extracting the meaningful textual corpus from the document, i.e the textual parts which contain the relevant facts reported in the resource. More specifically, the format conversion includes:

- the removal of unrelevant parts from the document by exploiting an open source Web service called Boilerpipe[1].

[1] http://code.google.com/p/boilerpipe/

- extracting metadata included in the source of the page by means of HTML tags such as *KEYWORDS*, *DESCRIPTION*, and *TITLE*.
- translating the text into the English language by using freely available API (we are currently using the Google Translate API).

The output of the format conversion phase is a text in English constituted by the title of the Web page, followed by the metadata extracted from the HTML tags, and concluded by the text extracted by the Boilerpipe service.

This text is analyzed in the **cleaning and sentence delimiting** step in order to delimit sentences, following the assumption that each keyphrase cannot be located simultaneously in two distinct sentences.

In the **POS-tagging and n-gram extraction** step we assign a POS tag (noun, adjective, verb, etc.) to each token in the cleaned text by using the Stanford log-linear part-of-speech tagger[2] and then we extract all possible subsequences of phrases including up to 3 words (uni-grams, bi-grams, and tri-grams).

In order to discard keyphrases which do not have a very significant meaning a pruning process is exploited in the subsequent **stemming and stopword removing** step where: the phrases starting or ending with a stopword or a sentence delimiter are removed; plural forms and singular forms are collapsed by using the Porter stemmer algorithm (Porter, 1997); a well defined set of POS patterns is used to filter for example uni-grams that are labeled as adjective or verb.

The output of all the previous steps is constituted by three lists containing respectively the resulting candidate uni-grams, bi-grams, and tri-grams.

2.2 DIKpEW: Phrase selection

As proposed in (Pudota et al., 2010), some characteristics of the candidate keyphrases are assessed in the **feature calculation** step for identifying the most relevant keyphrases. The evaluated characteristics have been identified by taking into account how Web pages usually store meaningful information. The considered features are qualitatively described below

1. ***Phrase frequency***: this feature is the classical term frequency (TF) metric, exploited in many state of the art keyphrase extraction systems (Turney, 1999)(Hulth, 2003)(Hulth and Megyesi, 2006). In our work, the TF value is normalized and computed separately for each n-gram list.
2. ***POS value***: as observed in (Hulth, 2003)(Barker and Cornacchia, 2000), most author-assigned keyphrases for a document turn out to be noun phrases. For this reason we increase the weight of candidate phrases containing more noun phrases.
3. ***Phrase depth***: following the idea that the main concepts and information are usually reported in the first part of the document we compute the phrase depth value for each phrase as the number of words preceding a phrase's first occurrence.
4. ***Wikipedia***. The Wikipedia feature is used to identify more coherent and recognized phrases by following the idea that keyphrases that are also entries of Wikipedia are more likely associated to well-defined concepts/meaning.
5. ***Title***. It highlights keyphrases that are included in the title of the Web page (if known). We followed the hypothesis that the title summarizes meaningful concepts which are more deeply discussed in the rest of the text.
6. ***Description***. Authors of Web pages often add a short description of the main contents of the Web page by using the *DESCRIPTION* HTML tag. According to the idea that the summary provided by the author may contain very meaningful information we compute this boolean feature for each keyphrase: the feature is set to 1 if the keyphrase is in the description, 0 otherwise.
7. ***Keyword***. Even if authors of Web pages are not required to classify their published resources, they usually add some keywords in order to be properly indexed by search engines. Since these terms are labels generated by the authors themself, we consider these terms as meaningful keyphrases.

In the **scoring and ranking** step we combined the value of each feature in order to compute a score (named *keyphraseness*) for each candidate keyphrase. The keyphraseness is a weighted combination of the evaluated features where the weights of the features where experimentally computed by using the opinions of a limited set of people.

Finally, the keyphrases associated to the higher keyphraseness are filtered and recommended in the final **keyphrase filtering** step.

3 Evaluation

Web pages are usually not classified with keyphrases by their authors and this had a strong impact on our evaluation procedure. In fact there are not freely available datasets which can be used to execute an automatic evaluation of the described mechanism. For this reason we decided to exploit a live evaluation

[2] http://nlp.stanford.edu/software/tagger.shtml.

involving a set of volunteers which had the task of judging the accuracy of the results returned by our approach. Moreover, due to the lack of keyphrases associated to Web pages, we could not use KEA (Turney, 2000) for comparing our results to one of the state of the art mechanisms: in fact, the KEA mechanism needs to be trained by using a corpus of annotated documents. In order to face this issue we decided to use as baseline approach a system where keyphrases are scored and ranked according to their frequencies. This choice seems reasonable since, as our approach does, the baseline approach takes into account only the information available in a specific document (without considering the characteristics of the documents in a specific collection): the most frequent keyphrases obtain an higher score. By using such score, the baseline mechanism can extract the two top scored uni-grams, the five top scored bi-grams, and the three top scored tri-grams. The final set of keyphrases is then built by these 10 filtered keyphrases.

The results returned by both our mechanism and the baseline approach were evaluated by using a Web application where a set of volunteers judged the accuracy of the results. Since our approach is mainly aimed at supporting the users of social tagging systems, we created a Web based application which simulates the interaction of a user with a social tagging system. By using this application, the volunteers could submit an URL and then the evaluation framework returned to the users a list of keyphrases for the specific Web page. The list of returned keyphrases was built by the results produced by both the proposed approach and the baseline mechanism. However, the two sets of keyphrases were presented to the evaluators mixed in a random order. By merging the keyphrases without a specific order we avoided to bias the human evaluators since they were not able to recognize the keyphrases returned by one of the two compared approaches.

The evaluators had to vote each returned keyphrase by using the following 5-Likert scale: **Excellent** - the keyphrase is very meaningful. It reports relevant facts, people, topics or other elements which characterize the Web page; **Good** - the keyphrase is still significant for classifying the document but it is not the best. The keyphrase reports facts, people, topics or other elements which characterize the Web page, but are more weakly connected to the main content of the page; **Neutral** - you are not sure about the significance of the keyphrase for the document; **Poor** - the keyphrase does not properly describe the contents; **Very Poor** - the keyphrase does not make sense. We involved 26 volunteers (20 men and 6 women) who worked for two weeks. The volunteers were students and workers. The oldest participant was 63 years old, the youngest was 22 years old and the average age was 37 years. The volunteers evaluated the keyphrases generated for 209 Web pages written in Italian and in English.

We used the Normalized Discounted Cumulative Gain (NDCG) metric to evaluate the results of our evaluation. The NDCG metric is commonly used in the area of Information Retrieval in order to evaluate the accuracy of ranking mechanisms. This measure is specifically used in scenarios where the ranked results are associated to different relevance levels, since it takes into account both the position and the usefulness (or gain) of the results to assign a score to the evaluated ranking mechanisms. In particular, the NDCG metric is based on the assumption that an accurate ranking mechanism puts the most relevant results in the first positions of the generated ranking. This means that the accuracy of a ranking mechanism is assessed by the NDCG metric by combining information about the position of the items in the ranking and the feedback relevance provided by the users. Technically, the NDCG metric assigns a score to a ranking mechanism by taking into account a set of ranked lists of resources where, given a list of ranked resources, each resource is associated to one specific grade value of a graded relevance scale. More formally, given a ranking mechanism and a ranked list of resources returned by the mechanism where the resource (in our case the keyphrase) in position i is associated to a relevance level rel_i (in our case the position is defined by our algorithm and the relevance by the evaluators) the NDCG computes the gain for this list as follows

$$DCG = rel_1 + \sum_{i=2}^{n} \frac{rel_i}{log_2 i}$$

where n is the number of results in the ranked list and in our specific case n is equal to 10. In our evaluation the graded relevance scale is defined by the following relevance levels: Excellent = 4; Good = 3; Neutral = 2; Poor = 1; Very poor = 0. The DCG is then used to quantify the accuracy of a response generated by a ranking mechanism according to both a fixed relevance scale and the opinions of the evaluators.

By computing the DCG over each evaluation provided by our evaluators, we obtained an assessment of the accuracy for each evaluated Web page. These computed DCG are combined in the computation of the NDCG which is used to normalize the DCG values in $[0,1]$ and, finally, to compute the accuracy of the mechanism as the mean of these normalized val-

	NDCG@5	NDCG@10
Base_Ita	0.484	0.437
DIKpEW_Ita	0.558	0.614
Base_Eng	0.485	0.576
DIKpEW_Eng	0.523	0.686

Table 1: Performance of DIKpEW compared to the baseline mechanism

ues. If the evaluated keyphrase extraction mechanism returns only very relevant keyphrases then the NDCG assumes the ideal value 1. Table 1 reports the 8 different NDCG values computed for evaluating and comparing the accuracy of the top 5 and top 10 keyphrases extracted by: (i) the baseline system from Web pages written in Italian (Base_Ita); (ii) our approach from Web pages written in Italian (DIKpEW_Ita); (iii) the baseline system from Web pages written in English (Base_Eng); (iv) our approach from Web pages written in English (DIKpEW_Eng); .

According to the results showed in Table 1 our approach outperforms the baseline mechanism. Moreover, the accuracy of the results computed for the Web pages in Italian are comparable to the accuracy for the Web pages in English. This means that the noise introduced by the translation in English does not significantly lowers the accuracy of the results. This can be justified in two ways: (i) the weight of the keyphrase depends on a set of statistical features which discard possible incorrect translation; (ii) the Wikipedia feature allows us to throw out (or at least to assign to lower positions) the bi-grams and tri-grams which have not a clear meaning.

A final consideration concerns the NDCG metric: it is important to emphasize that we exploited it only for comparing our approach to a baseline reference. In fact, the choice of selecting only the top N keyphrases (where N=5 or N=10) does not tackle the possibility of working with pages with only 2 or 3 significant phrases. In this case, the NDCG@10 metric, for example, would be much lower than 1. Future work will also address this issue.

4 Conclusion

In this work we presented an approach which is aimed at supporting the users of social tagging systems in classifying Web pages. In particular, the presented approach identifies English n-grams from a Web document for suggesting meaningful labels for the specific resource. An experimental evaluation showed that the proposed approach is plausible and future analysis will investigate if the proposed approach can produce better results for specific topics or specific set of Web pages (blogs, newspapers, etc.).

The proposed approach can provide keyphrases which appear already in the given document. Future work will focus on overcoming this limitation by navigating other knowledge sources such as Wikipedia and Wordnet, producing in such a way meaningful tags which are constituted by uni-grams, bi-grams, or tri-grams not contained in the text, and that are the result of a domain reasoning activity. We also plan to integrate our approach in collaborative and content-based recommender systems following the ideas proposed in (Ferrara and Tasso, 2011) and (Ferrara et al., 2011).

REFERENCES

Barker, K. and Cornacchia, N. (2000). Using noun phrase heads to extract document keyphrases. In *Proceedings of the 13th Biennial Conference of the Canadian Society on Computational Studies of Intelligence*, pages 40–52, London, UK. Springer-Verlag.

Ferrara, F., Pudota, N., and Tasso, C. (2011). A keyphrase-based paper recommender system. In Agosti, M., Esposito, F., Meghini, C., and Orio, N., editors, *Digital Libraries and Archives*, volume 249 of *Communications in Computer and Information Science*, pages 14–25. Springer Berlin Heidelberg.

Ferrara, F. and Tasso, C. (2011). Extracting and exploiting topics of interests from social tagging systems. In *Proceedings of the International Conference on Adaptive and Intelligent Systems*, ICAIS'11, pages 285–296, Berlin, Heidelberg. Springer-Verlag.

Hulth, A. (2003). Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 216–223, Morristown, NJ, USA. Association for Computational Linguistics.

Hulth, A. and Megyesi, B. B. (2006). A study on automatically extracted keywords in text categorization. In *ACL-44: Proc. of the 21st Int. Conf. on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 537–544, Morristown, NJ, USA. ACL.

Porter, M. F. (1997). An algorithm for suffix stripping. *Readings in information retrieval*, pages 313–316.

Pudota, N., Dattolo, A., Baruzzo, A., Ferrara, F., and Tasso, C. (2010). Automatic keyphrase extraction and ontology mining for content-based tag recommendation. *International Journal of Intelligent Systems, Special Issue: New Trends for Ontology-Based Knowledge Discovery*, 25:1158–1186.

Turney, P. (1999). Learning to extract keyphrases from text. Technical Report ERB-1057, National Research Council, Institute for Information Technology.

Turney, P. D. (2000). Learning algorithms for keyphrase extraction. *Information Retrieval*, 2(4):303–336.