Using Semantic Classes as Document Keywords*

Rubén Izquierdo[1], Armando Suárez[1], and German Rigau[2]

[1] GPLSI Group, University of Alicante, Spain
{ruben,armando}@dlsi.ua.es
[2] IXA NLP Group, EHU, Donostia, Spain
german.rigau@ehu.es

Abstract. Keyphrases are mainly words that capture the main topics of a document. We think that semantic classes can be used as keyphrases for a text. We have developed a semantic class–based WSD system that can tag the words of a text with their semantic class. A method is developed to compare the semantic classes of the words of a text with the correct ones based on statistical measures. We find that the evaluation of semantic classes considered as keyphrases is very close to 100% in most cases.

1 Introduction

Keyphrases are mainly words that capture the main topics of a document. Keyphrases can be single words or compound words or multiwords. They can provide a high level view of the content of a document. For example, they can be used by a reader to know if some document is relevant for him or not. In this sense, the authors of a lot of scientific papers are required to include some keyphrases describing the paper. These topics can be used also for classifying the paper.

There are more applications of keyphrases. For example in Document Summarization, keyphrase represent the meaning of a whole document in a few topics, providing a very concise summary. Also in clustering and Information Retrieval, keyphrases have an important role. In general, keyphrases provide a powerful way for representing the meaning and content of a document in a bunch of topics. There are a lot of documents that have no keyphrases pre–assigned, and doing it manually is a hard task. Hence, the development of automatic techniques for perform keyphrase identification is a very interesting research area. There are two sub-types of this task: keyphrase extraction (selecting some relevant phrases from a document) and keyphrase assignment (assign a predefined list of categories as keyphrases).

The work in [6] presents one of the first developed systems for keyphrase extraction. It was implemented by means of a genetic algorithm. This algorithm

* This paper has been supported by the European Union under the project KYOTO (FP7 ICT-211423), the Valencian Region Government under projectss PROMETEO/2009/119 and ACOMP/2011/001 and the Spanish Government under the project TEXT MESS 2.0 (TIN2009-13391-C04-01) and KNOW2 (TIN2009-14715-C04-01).

R. Muñoz et al. (Eds.): NLDB 2011, LNCS 6716, pp. 225–229, 2011.

adjusts a set of heuristic rules to extract the correct keyphrases. [7] developed the KEA system, a machine learning system for keyphrase extraction. Recently, the interest in keyphrase extraction has reemerged, leading to the development of several new systems and techniques. Moreover, in last SemEval conference[1], a specific task was proposed in order to evaluate the performance of keyphrase extraction systems. The task was called *Automatic Keyphrase Extraction from Scientific Articles*.

2 Using Semantic Classes as *Keyphrases*: Evaluation

Our idea is to use semantic classes as keyphrases for representing the main topics of a document. We think than the semantic classes of the words can be informative about the topics of a text. This effect could be due to the semantic classes provide a high semantic level of abstraction, not considering individual words but general concepts. To obtain these semantic classes, we employ a semantic class based *Word Sense Disambiguation* (from now on WSD) system that we have developed to tag each noun and verb with its proper semantic class.

We have developed a WSD based on machine learning and semantic classes. Our system uses an implementation of a Support Vector Machine algorithm to train the classifiers using SemCor [4] Corpus for acquiring examples. The system uses set of traditional features for representing these examples, expanded with two semantic features. More details about the WSD system can be found in [2].

Semantic classes are concepts that subsume sub–concepts and words with a related meaning and common characteristics. We use two semantic class sets: the first one automatically extracted from WordNet, the Basic Level Concepts, and the second one, SuperSense, was created manually by the developers of WordNet. On the one hand, **Basic Level Concepts**[2] (BLC) [1] are small sets of meanings representing the whole nominal and verbal part of WN. **SuperSenses** (SS) are based on open syntactic categories (nouns, verbs, adjectives and adverbs) and logical groupings, such as person, phenomenon, feeling, location, etc.

Our method of keyphrase extraction consists of considering the semantic classes of the words within a document, and select the most frequent ones as keyphrases for the document. One important criterion is the selection of what words are considered to calculate the overall set of semantic classes. Semantic classes of domain general words are maybe not informative and can introduce noise, so they must not be taken into account. To remove this general words, we use a filter, implemented by means of an statistical measure traditionally used in the field of Information Retrieval: the TF-IDF measure [3]. To apply this formula we need a big corpus belonging to a general domain, and The British National Corpus[3] (BNC) is chosen[4].

[1] `http://semeval2.fbk.eu`

[2] `http://adimen.si.ehu.es/web/BLC`

[3] `http://www.natcorp.ox.ac.uk`

[4] The BNC is a 100 million word collection of samples of written and spoken language from a wide range of sources, designed to represent a wide cross-section of current English.

To evaluate our approach we use the corpora from the WSD All Words tasks from SensEval–2[5], SensEval–3[6] and SemEval–1[7]. We use these corpora due to the texts contained on them are supposed to be domain specific and show a good coherence to perform keyphrase extraction and topic detection. Moreover, existing corpora for keyphrase extraction are not well designed, and are mainly focused on keyphrase assignment, and are not designed for keyphrase extraction. Specifically, each test corpus (SensEval–2, SensEval–3 and SemEval–1) consists of three independent documents.

We weigh each word in a document according to the TF–IDF formula. The formula is calculated combining two values: the frequency of a word within a document (SenEval of SemEval document), and the number of BNC files in which the word appears. So, words that are related with the specific domain of the document (words with a high document frequency) and are not general domain words (words contained on a low number of BNC files), will have a high TF–IDF value. Once we have all words of a document weighted, we sort them according to this weight. We propose two experiments for calculating the semantic classes of a document: (1) consider the semantic classes of all words within the document, and (2) consider the semantic classes of the 99% words with higher weight. As explained before, in the second experiment we expect to remove general domain words.

3 Results

We have three ranks of semantic classes: the guessed by our system (G), the correct rank (C), and the most frequent rank (M). We compare pairs of ranks by means of the Spearman's rank correlation[5]. This is a non-parametric measure of statistical dependence between two variables. In other words, it gives a measure of the similarity or two sorted ranks, and this fits very well with our case.

We first show the evaluation using **BLC–20[8] semantic classes**. In table 1 the values for the Spearman's rank correlation are shown, including the experiment considering all words (no filtering), and the experiment considering only the 99% words with higher weight (TF–IDF filtering). We can see that in the most cases, the values are very close to 1, what indicates identical ranks. Moreover, in a lot of cases, our rank (G) is more similar to the correct (C) than the most frequent rank (M). The behaviour of the system in the three corpora is similar, obtaining higher results in SensEval–3 corpus. We can also see that the results considering TF–IDF filtering are not always higher that without filtering. This is maybe due to the filtering process is not well tuned, and some general domain words removed by the filter contain actually relevant semantic information about the document. Anyway, the results are very close to 1, indicating

[5] `http://86.188.143.199/senseval2`
[6] `http://www.senseval.org/senseval3`
[7] `http://nlp.cs.swarthmore.edu/semeval`
[8] BLC–20 is a kind of BLC where each BLC concept must subsume at least 20 subconcepts.

Table 1. Spearman values for BLC–20

Corpus	File	No filtering			TF–IDF filtering		
		G–C	G–M	M–C	G–C	G–M	M–C
SV2	*d00*	0.9974	0.9961	0.9906	0.9948	0.9894	0.9773
	d01	0.9906	0.9901	0.9697	0.9761	0.9661	0.9930
	d02	0.9899	0.9983	0.9905	0.9835	0.9979	0.9828
SV3	*d000*	0.9981	0.9995	0.9978	0.9969	0.9987	0.9962
	d001	0.9885	0.9962	0.9897	0.9786	0.9966	0.9637
	d002	0.9977	0.9998	0.9981	0.9966	0.9991	0.9971
SEM1	*d00*	0.9515	0.9893	0.9103	0.9253	0.9940	0.8593
	d01	0.9514	0.9995	0.9514	0.9040	1	0.9040
	d02	0.9721	0.9444	0.9690	0.9560	0.9995	0.9679

Table 2. Spearman's values for SuperSense

Corpus	File	No filtering			TF–IDF filtering		
		G–C	G–M	M–C	G–C	G–M	M–C
SV2	*d00*	0.7994	0.7325	0.6150	0.8464	0.6091	0.3275
	d01	0.7069	0.6151	-0.2917	0.7309	0.6029	-0.3221
	d02	0.7875	0.9754	0.7911	0.5182	0.9231	0.4818
SV3	*d000*	0.8029	0.9201	0.8431	0.8765	0.9456	0.9235
	d001	-0.4923	0.9701	-0.2352	0.5099	0.9868	0.5099
	d002	0.8820	0.9802	0.8323	0.9657	0.9816	0.9397
SEM1	*d00*	0.4524	0.6426	0.5	0.4405	0.9818	0.5
	d01	0.9182	0.5412	0.9318	0.9030	0.9955	0.9212
	d02	0.5955	-0.1448	0.9126	0.3333	0.7394	0.9273

that we can use our BLC–20 semantic class WSD system to detect and extract relevant concepts representing the main topics of a document. In other words, we can use our semantic class WSD system as keyphrase extractor.

Similarly, we perform the same experiment considering the **SuperSense**–based WSD system. The results of the Spearman's rank correlation are shown in table 2. In this case the results are considerably lower, although SuperSense semantic classes have a higher level of abstraction and the average polysemy is lower, and the results could be expected to be higher. This can indicate that the BLC concepts work very well as keyphrases. The words within a document seem to maintain a certain semantic coherence considering BLC classes, whereas this coherence is not hold in the case of SuperSense, maybe due to this classes are too coarse–grained.

4 Conclusions

In this paper we propose the use of a semantic class–based WSD system in order to perform keyphrase extraction. To do this, we use different kinds of semantic classes as keyphrases, expecting semantic classes to represent the topic

information of a document. We represent the topic information of a document starting from the semantic classes assigned automatically by the WSD system to the words within the document. Then we use an statistical measure, the Spearman's rank correlation, to evaluate our system, and compare the semantic classes as keyphrases with the correct ones.

In general, we obtain very good results with the Basic Level Concepts–based system, better than using Supersense. Therefore, the performance of the keyphrase extractor based on a WSD system not depends only in the abstraction level of the semantic classes used, but also in the discriminative power and the coherence of the set.

References

1. Izquierdo, R., Suarez, A., Rigau, G.: Exploring the automatic selection of basic level concepts.In: Angelova, G., et.al. (eds.) International Conference Recent Advances in Natural Language Processing, Borovets, Bulgaria, pp. 298–302 (2007)
2. Izquierdo, R., Suárez, A., Rigau, G.: An empirical study on class-based word sense disambiguation. In: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics, Athens, Greece, EACL 2009, pp. 389–397. Association for Computational Linguistics, Stroudsburg (2009)
3. Jones, K.S.: A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation 28, 11–21 (1972)
4. Miller, G., Leacock, C., Tengi, R., Bunker, R.: A Semantic Concordance. In: Proceedings of the ARPA Workshop on Human Language Technology (1993)
5. Spearman, C.: The proof and measurement of association between two things. The American Journal of Psychology 15(1), 72–101 (1904)
6. Turney, P.D.: Learning algorithms for keyphrase extraction. Inf. Retr. 2, 303–336 (2000), http://portal.acm.org/citation.cfm?id=593957.593993
7. Witten, I.H., Paynter, G., Frank, E., Gutwin, C., Nevill-Manning, C.G.: KEA: Practical Automatic Keyphrase Extraction. In: Proceedings of Digital Libraries 1999 (DL'99), pp. 254–255 (1999), http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.55.3127