Conceptual document indexing using a large scale semantic dictionary providing a concept hierarchy

Martin Rajman, Pierre Andrews, María del Mar Pérez Almenta, and Florian Seydoux

Artificial Intelligence Laboratory, Computer Science Department
Swiss Federal Institute of Technology
CH-1015 Lausanne, Switzerland
(e-mail: Martin.Rajman@epfl.ch, pierre.andrews@cs.york.ac.uk, mariadelmar.perezalmenta@epfl.ch, Florian.Seydoux@epfl.ch)

Abstract. Automatic indexing is one of the important technologies used for Textual Data Analysis applications. Standard document indexing techniques usually identify the most relevant keywords in the documents. This paper presents an alternative approach that aims at performing document indexing by associating concepts with the document to index instead of extracting keywords out of it. The concepts are extracted out of the EDR Electronic Dictionary that provides a concept hierarchy based on hyponym/hypernym relations. An experimental evaluation based on a probabilistic model was performed on a sample of the INSPEC bibliographic database and we present the promising results that were obtained during the evaluation experiments.

Keywords: Document indexing, Large scale semantic dictionary, Concept extraction.

1 Introduction

Keyword extraction is often used for documents indexing. For example, it is a necessary component in almost any Internet search application. Standard keyword extraction techniques usually rely on statistical methods [Zipf, 1932] to identify the important content bearing words to extract. However it has been observed that such extractive techniques are not always efficient, especially in situations where important vocabulary variability is possible.

The aim of this paper is to present a new algorithm that does not extract keywords from the documents, but associates them with concepts representing the contained topics [Rajman et al., 2005]. The use of a concept ontology is necessary for this process. In our work, we use the EDR Electronic Dictionary (developed by the Japan Electronic Dictionary Research Institute [Institute (EDR), 1995]), a semantic database that provides associations between words and all the concepts they can represent, and organizes these concepts in a concept hierarchy based on hyponym/hyperonym relations.
In our approach, the indexing module first divides the documents into topically homogeneous segments. For each of the identified segments, it selects all the concepts in EDR that correspond to all the terms contained in the segment. The conceptual hierarchy is then used to build the minimal sub-hierarchy covering all the selected concepts and this sub-hierarchy is explored to identify a set of concepts that most adequately describes the topic(s) discussed in the document. A "most adequate" set of concepts is defined as a cut in the sub-hierarchy that jointly maximizes specific genericity and informativeness scores.

An experimental evaluation, based on a probabilistic model, was performed on a sample of the INSPEC bibliographic database [INSPEC, 2004] produced by the Institution of Electrical Engineers (IEE). For this purpose, an original evaluation methodology was designed, relying on a probabilistic measure of adequacy between the selected concepts and available reference indexings.

The rest of this contribution is organized as follows: in section 2, we describe the EDR semantic database that we use for concept extraction. In section 3, we present the necessary text pre-processing steps that need to be applied for concept extraction to be performed. In section 4, we present the concept extraction algorithm. In section 5, we describe the evaluation framework and the obtained results. Finally, in section 6, we present conclusions and future works.

2 The Data

The EDR Electronic Dictionary [Institute (EDR), 1995] is a set of linguistic resources that can be used for natural language processing. It consists of several parts (dictionaries). For our work, we used the Concept dictionary that provides about 400'000 concepts organized on the basis of hyponym/hyponym relations (See figure 1), and the English word dictionary that provides grammatical and semantic information for each of the dictionary entries. Dictionary entries can be either simple words or compounds.

At the semantic level, the EDR word dictionary provides relations between words and concepts. Notice that, in the case of polysemy, one word can be associated with more than one concept.

![Fig. 1. An example of Concept classification in the EDR Concept dictionary.](image-url)
3 Pre-processing the texts

Document segmentation The first pre-processing step is document segmentation. Segmentation is necessary because it allows not to have to process simultaneously all the concepts that might be potentially associated with a large document, in which case concept extraction would be computationally inefficient. However, to preserve the quality of the extracted concepts, the used segments must be topically homogeneous. For this purpose, we implemented a simple, well known, Text Tiling technique [Hearst, 1994], where segmentation is based on a measure of proximity between the lexical profiles representing the segments. For the rest of this document, we will consider that the segmentation step has been preformed and the elementary unit for concept association will be the segment, not the document. Once concepts are associated with all the segments corresponding to a document, they are simply merged to produce the set of concepts associated with the document itself.

Tokenization The next pre-processing step is tokenization, which is necessary to decompose the document in distinct tokens that will serve as elementary textual units for the rest of the processing. For this purpose, we used the Natural Language Processing library SLPtoolkit developed at LIA [Chappelier, 2001]. In this library, tokenization is driven by a user defined lexicon, and the resulting tokens can therefore be simple words or compounds. For this purpose, the used lexicon had to be adapted to EDR, so as to contain every possible inflected form of any EDR entry. As EDR does not directly provide these inflected forms, but only the lemmas with inflexion information, we had to write a specific program that exploits the available information to produce the required inflected forms.

Part of Speech Tagging and Lemmatization This pre-processing step consists in identifying, for each token, the lemma and Part Of Speech (POS) category corresponding to its context of use in the document. For our experiments, we used the Brill POS tagger [Brill, 1995].

The output of all the pre-processing steps is the decomposition of each of the identified segments in sequences of lemmas corresponding to EDR entries and associated with the POS category imposed by their context of occurrence. However, because of the polysemy problem already mentioned earlier, this is not sufficient to associate each of the triggered EDR entries with one single concept corresponding to its contextual meaning. Some technique performing semantic disambiguation would be required for that. However, as semantic disambiguation is currently not yet efficiently solved, at least for large scale applications, we decided to keep the ambiguity by triggering all the concepts potentially associated with the (lemma, POS) pairs appearing in the segments. The underlying hypothesis is that some semantic disambiguation will be implicitly performed as a side-effect of the concept selection algorithm. This aspect should however be further investigated.
4 Concept Extraction

The goal of the concept extraction algorithm is to select a set of concepts that most adequately represents the content of the processed document. To do so, we first trigger all the possible concepts that are associated, with the EDR word entries identified in the document. Then, we extract out of the EDR hierarchy the minimal sub-hierarchy that covers all the triggered concepts. This minimal hierarchy, hereafter called the ancestor closure (or simply the closure), is defined as the part of the EDR conceptual hierarchy that only contains, either the triggered concepts themselves, or any of the concepts dominating them in the hierarchy. Notice that the only constraint imposed on the conceptual hierarchy for the definition of a closure to be valid is that the hierarchy corresponds to a non cyclic directed graph. In such a hierarchy, we call leaves (resp. roots) all the nodes connected with only incoming (resp. outgoing) links. The EDR hierarchy indeed corresponds to a non cyclic directed graph and, in addition, each of its two distinct parts (the technical concepts and the normal concepts) contains only one single root (hereafter called the root).

Once the closure corresponding to the triggered concepts is produced, the candidates for the possible set of concepts to be considered for representing the content of the document are the different possible cuts in the closure. For any non cyclic directed graph, we define a cut as a minimal set of nodes that dominates all the leaves of the graph. Notice that, by definition, the set of the roots of the graph, as well as the set of its leaves, both correspond a cut.

4.1 Cut Extraction

The idea behind our approach is to extract a cut that optimally represents the processed document. To do so, our algorithm explores the different cuts in the closure, scores them, and select the best one with respect to the used score. As a cut can be seen as a more or less abstract representation of the
leaves of the closure, the score of a cut is computed relatively to the covered leaves. In our algorithm, a local score is first computed for the concepts in the cut, and a global score is then derived for the cut from the obtained local scores. Notice also that, as the number of cuts in a closure might be exponential, evaluating the scores of all possible cuts is not realistic for real size closures. A dynamic programming algorithm was therefore developed to avoid intractable computations [Rajman et al., 2005].

In this algorithm, the local score $U$ (the definition of $U$ is given in section 4.2) is computed for each concept $c$ in the cut. This local score measures how much the concept $c$ is representative of the leaves of the closure. The global score of the cut is then computed as the average of $U$ over all concepts in the cut.

4.2 Concept Scoring

The local score $U$ is decomposed into two specific components, genericity and informativeness.

**Genericity** It is quite intuitive that, in a conceptual hierarchy, a concept is more generic than its subconcepts. At the same time, the higher a concept lays in the hierarchy, the larger is the number of the leaves it covers. Following this, a simple score $S_1$ was defined to describe the genericity of a concept. We made the assumption that this score should be proportional to the total number $n(c)$ of leaves covered by the concept $c$. Because of the linearity assumption, the score $S_1$ of a concept $c$ can therefore be written as:

$$S_1(c) = \frac{n(c) - 1}{N - 1},$$

where $N$ is the total number of leaves in the closure.

**Informativeness** If only genericity would be taken into account, our algorithm would always select the roots of the closure as the optimal cut. Therefore, it is important to also take into account the amount of information preserved about the leaves of the closure by the concepts selected in the cut. To quantify this amount of preserved information, we defined a second score $S_2$ for which we made the assumption that the score $S_2(c)$ defined for a concept $c$ in a cut should be linearly dependent on the average normalized path length $d(c)$ between the concept $c$ and all the leaves it covers in the closure. Because of the linearity assumption, the score $S_2$ of a concept $c$ can therefore be written as: $S_2(c) = 1 - d(c)$.

**Score Combination** As two scores are computed for each concept in the evaluated cut, a combination scheme was necessary to combine $S_1$ and $S_2$ into a single score. A weighted geometric mean was chosen:

$$U(c) = S_1(c)^{1-a} \times S_2(c)^a.$$

The parameter $a$ offers a control over the number of concepts returned by the selection algorithm. If the value of $a$ is close to one, then it will favor the score $S_2$ over $S_1$, and the algorithm will extract a cut close to the leaves, whereas a value close to zero will favor $S_1$ over $S_2$ and therefore yield more generic concepts in the cut.
5 Evaluation

The evaluation of the Concept Extraction algorithm was made on a sample from the INSPEC Bibliographic database, a bibliographic database about physics, electronics and computing [INSPEC, 2004]. The sample was composed of short abstracts manually annotated with keywords extracted from the abstracts. For the evaluation, a set of 238 abstracts was randomly selected in database, and these abstracts were manually associated with two sets of concepts: the ones corresponding to a simple keyword in the reference annotation, and, the ones corresponding to compound keywords.

In our case, only the concepts of the first kind were considered and all compound keywords were first decomposed into their elementary constituents and then associated with the corresponding concepts.

To measure the similarity between the concept derived from the reference annotation and the ones produced by our algorithm, we used the standard Precision and Recall measures. For any indexed document, Precision is the fraction of identified correct concepts in all concepts associated with the document by the algorithm, while Recall is the fraction of the identified correct concepts in all concepts associated with the document in the reference annotation. For any set of documents, the quality of the concept association algorithm was then measured by the average Precision and Recall scores over all the documents in the sample.

However, if applied directly, an evaluation based on Precision and Recall scores would be quite inadequate, as it does not take at all into account the hyponym/hypernym relations relating the concepts. For example if a document is indexed by the concept "dog" and the algorithm produces the concept "animal", this should not be considered as a total failure as it would be the case with the standard definition of Precision and Recall. To take this into account, we replaced the binary match between produced and reference concepts by a similarity measure based on the available concept hierarchy. The selected similarity measure was the Leacock-Chodorow similarity [Leacock and Chodorow, 1998] that corresponds to the logarithm of the normalized path length between two concepts. The probabilistic model then used for the evaluation was the following: the normalized version of the concept similarity between a produced concept \( c_i \) and a reference concept \( C_k \), denoted by \( p(c_i, C_k) \), is interpreted as the probability that the concepts \( c_i \) and \( C_k \) can match. Then, if \( Prod = \{c_1, c_2, ..., c_n\} \) is the set of concepts produced for a document and \( Ref = \{C_1, C_2, ..., C_N\} \) is the corresponding set of reference concepts, for each concept \( c_i \) (resp. \( C_k \)) the probability that it matches the reference set \( Ref \) (resp. the produced set \( Prod \)) is:

\[
p(c_i) = 1 - \prod_{k=1}^{N} (1 - p(c_i, C_k)) \quad (\text{resp.} \quad p(C_k) = 1 - \prod_{i=1}^{n} (1 - p(c_i, C_k))),
\]

and the expectations for Precision and Recall can therefore be computed as:

\[
E(P) = \frac{1}{n} \times \sum_{i=1}^{n} p(c_i) \quad \text{and} \quad E(R) = \frac{1}{N} \times \sum_{k=1}^{N} p(C_k).
\]

For the obtained expected values for \( P \) and \( R \), the usual F-measure can then be computed.
5.1 Results and Interpretation

A first experiment was carried out to select which value of $a$ should be used for the evaluation. Observing the average results obtained for each value of $a$ (see figure 3), one can see that $a$ has a very limited impact on the algorithm performance (the F-measure is quasi constant until $a = 0.6$). The obtained results therefore seem to indicate that the value of $a$ can be chosen almost arbitrarily between $a=0.1$ and $a=0.7$.

In a second step, the following procedure was applied to compute the average Precision and Recall: (1) all the probabilities $p(c_i)$ and $p(C_k)$ were computed for each document in the evaluation corpus; (2) the concepts $c_i$ in Prod and $C_k$ in Ref were sorted by decreasing probabilities; (3) for each value $\Theta$ in an equi-distributed set of threshold values in [0,1[ an average (Precision, Recall) pair was computed, taking only into account the concepts $c$ for which $p(c) > \Theta$; (4) average values of Precision, Recall and F-Measure were computed over all the produced pairs.

The obtained curves shown in figure 4 display an interesting behavior: when Recall increases, Precision first starts to raise and then falls down. This might be explained by the fact that cuts corresponding to higher Recall values contain more concepts and that there is therefore a good chance that these concepts are lower in the hierarchy and have more chances to be close to the concepts in the reference. Then, when the number of produced concepts is too large, it exceeds what is necessary to cover the reference concepts and the added noise therefore entails a drop in Precision. A second interesting
observation is that, for a=0.6 and a=0.7, there are no (Precision, Recall) pairs with Recall larger than 0.8. This might be explained by the fact that, for small values of a, there is only a small chance that the extracted cut is specific enough to have a good probability to match all the reference concepts, and therefore makes it hard to reach high values of Recall.

6 Conclusion

Current approaches to automatic document indexing mainly rely on purely statistical methods, extracting representative keywords out of the documents. The novel approach proposed in this contribution gives the possibility of associating concepts instead of extracting keywords. For that, the construction of the ancestor closure over the segment’s concepts is used to choose the best representative set of concepts to describe the document’s topics. The novel evaluation method developed to measure the proposed concept extraction algorithm lead to promising results in terms of Precision and Recall, and also gave the opportunity to observe interesting features of the concept association mechanism. It proved that extracting concepts instead of simple keywords can be beneficial and does not require intractable computation.

As far as future works are concerned, more sophisticated methods to solve the ambiguity in concept association related to word polysemy should be investigated. A more general theoretical framework providing some well grounded justification for the scoring scheme should also be worked out.

References