Extracting text keywords using WordNet

Short Paper

Catalin Cerbulescu
University of Craiova
P.O. Box 200776
Romania
ccerbulescu@electronics.ucv.ro

Georgiana Silvia Leotescu
University of Craiova
P.O. Box 200764
Romania
geo_sylvia@yahoo.com

ABSTRACT

Summarizing, extracting keywords, sorting and filtering large quantities of texts were subject of various algorithms based on lexical databases because this was a difficult and time consuming task for a human operator. Our proposed algorithm is based on WordNet lexical database. It eliminates the connection words. For the remaining words (only nouns or verbs) we build a tree with several levels of more generic terms like hypernym or lexicographer file. Using custom weights for each tree level and statistical analysis, we extract a restricted number of words which are used to define the keywords of a document. These results can be used to sort and filter a document based on relevance.

CCS CONCEPTS

• **Artificial intelligence → Natural language processing → Lexical semantics**;

KEYWORDS

Hypernymy, Lexicographer file, Text Summarization, Keyword Weight, Word sense analysis, WordNet

ACM Reference format:

Catalin Cerbulescu and Georgiana Silvia Leotescu. 2017. Extracting text keywords using WordNet. *In Proceedings of BCI '17, September 20–23, 2017, Skopje, Macedonia*, 4 pages.
https://doi.org/10.1145/3136273.3136280

1 THE HUMAN APPROACH ON DOCUMENT SUMMARIZATION.

The frequent tasks for an operator, when dealing with a huge number of documents are a) summarize a document and extract a restricted set of sentences or ideas b) extract some keywords c) sort documents based on their relevance for a user. The tasks do not consist in basic statistical operations like counting words only, but more in extracting general ideas and finding more generic terms. These tasks are also directions in the effort of automatic information filtering and classification using software tools. Extracting a document keyword set is used to a) index a document b) search for general words c) compute a linguistic distance between the document and an imposed set of words to rank the document and sort a set of documents by relevance.

Language skills and the use of a language system are very important for determining a document's relevance because we have to deal with different word forms, connection words and potential syntax and grammar issues.

When we take into account a language system, we have to refer to its fundamental function: to provide a form of expression (written or spoken) for thought and feeling. If we speak about the way in which words are combined into sentences (again, whether written or spoken), we inevitably refer to grammar. We can perceive grammar "as a coin whose two sides are expression and meaning and whose task is to systematically link the two" [6]. The knowledge of forming sentences is called competence, and being able to utter them is described as performance.

Competent speakers find it comfortable to alter the words and structure of a sentence because they know how to use the smallest units of grammar (morphemes). Syntax is also essential in conveying a message. Documents elaborated by native or competent speakers can contain linguistic style figures and they are often almost impossible to be analyzed by an automatic tool because the results are far from the expected ones.

What a word means is usually defined by its relationship to other words in its close neighborhood. There are non-hierarchical relations which "basically structure lexical items in terms of synonyms and various forms of oppositions." [13].

We can distinguish three major hierarchical relations which are taxonomies, meronomies and proportional series [13]. Taxonomies (also referred to as hyponymy) associate a hyponym (a specific item) to a superordinate (a more generic item or hypernym). This means that part of a word's meaning is not only connected to antonymy and synonymy, but also to the manner in which it fits into the vocabulary hierarchy. Nouns, verbs and adjectives can be "classified" as hyponyms (specific words) and hypernym or superordinates (generic words) according to lexical and semantic relations, so can they be studied hierarchically.

Concepts usually enter into a relationship of inclusion in a vertical sequence. By contrast, a relationship of opposition occurs in a horizontal sequence.

BCI '17, September 20–23, 2017, Skopje, Macedonia
© 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-5285-7/17/09…$15.00
https://doi.org/10.1145/3136273.3136280

Our paper is focused in analyzing the words relation in the vertical sequence. The use of a horizontal sequence can produce a redefinition of the original text and the possible finding another word in the document, defined as synonym.

The most important hierarchical relations are those of inclusion [1]. These contain generic relationships, referred to as species-genus. For example, words such as house, apartment building or firehouse are subordinated to the word building. [1]. For these words, we aim with our proposed algorithm to decide that one of the keywords for the document can be "building".

2 DOCUMENT CLASSIFICATION. WORDNET

A big step in the direction of a word's relation, in a vertical or horizontal sequence, was the development of lexical databases like WordNet because a hierarchy of words is defined based on linguistic studies. WordNet® is the largest lexical database in English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. The structure of the WordNet makes it a useful tool for computational linguistics and natural language processing. [17]. WordNet is the most important and widely used lexical resource for natural language processing [5].

"The most frequently encoded relation among synsets is the super-subordinate relation, also called hyponymy. It links more general synsets like {furniture, piece_of_furniture} to increasingly specific ones like {bed} and {bunkbed}. All noun hierarchies ultimately go up the root node {entity}. Hyponymy and hypernymy relation is also transitive." [17].

For WordNet, a hypernym is a term defined as more general or including another one. A is the hypernym for B if B is member of A or B is part of A. [17]. For example, a hypernym for bulldog can be animal but also dog. The word dog is closer as hypernym and meaning to bulldog than to animal.

The most basic part of a word, noun, verb or adjective, with no suffixes or prefixes, plural or derived form is called root word and it's a main feature in lexical databases and the starting point for building word relations.

A Part Of Speech (POS) is a category of lexical items with similar grammatical properties and behavior in terms of syntax. WordNet uses as POS only Adjective, Adverb, Noun, Verb.

[14] organize documents in meaningful clusters, based on statistical features. Fuzzy logic experiments were performed in summarizing the text based on WordNet [10]. [11] studies sentence-to-sentence semantic relatedness using the shortest path between synsets in WordNet as the core to measure the relatedness between two sentences and paraphrase detection. Various technologies like Neo4J were used aiming to improve the WordNet Graph visualization [4]. Other authors are interested in studying the word relatedness and establish if one word pair is more semantically related than another [12]. Researches in the field of identifying word sense and represent knowledge are word relations and word clustering. Knowledge from a particular domain can be represented by assembling and displaying its ontology components [9]. An ontology can be used to give order to unstructured sources and/or to provide a vocabulary of concepts and relations based on the digital source [9]. Disambiguation of the message sense can be done by using basic-level categories [15] or by using a class labeling algorithm and re-organizing data items so that they can be grouped into categories [16]. Neural algorithms in [16] reaches to semi-automatically label group items in a multi-branch hierarchy.

Word clustering can be achieved by analyzing large data amount of data. Software tools allows it by grouping words by sense and patterns [2]. Google search results were also used in [3].

The strategies for extracting semantic information from corpora can be roughly divided into two categories, knowledge-rich and knowledge-poor methods, according to the amount of knowledge they presuppose [8]. Knowledge-rich approaches require some sort of previously encoded semantic information: domain-dependent knowledge structures, semantic tagged training corpora, and/or semantic resources such as handcrafted thesauri such as WordNet [7].

3 NATURAL LANGUAGE PROCESSING

Word sense analysis, using WordNet, in order to establish word ontology can use sysnset approach to establish word proximity (a set of synonyms) or sense relations between words (hyponyms, hypernyms, etc). The right sense for a word can also be obtained using the context [3]. Using concept maps, [19] studies the problem of selecting the right sense for a polysemantic word using the context and word hypernyms.

The extracted keywords are in semantic relation of generalization with the original words, defined and used by WordNet as lexicographer file or hypernyms. The lexicographer file is a very general and restricted (less than 30) set of part of speech. There are 26 lexicographer files defined by WordNet, like: noun.animal, noun.artifact, noun.body, etc. [17] so the result of generalization will be very restricted.

Processing the document to extract the keywords will bring to the software tool some of the following challenges: *1) The language system* differs from one author to another. Focusing on words rather than word context will simplify the problem but can introduce inadequate results; *2) Potential grammar mistakes* can make the word not understandable or understood with a wrong meaning. The right meaning of a word can be extracted considering the context; *3) The Parts of speech* used by WordNet as hypernyms and lexicographer files are nouns and verbs; *4) Software API for retrieving hypernyms* and lexicographer file use root word so a root word extraction is necessary *5) Each word can have multiple* hypernyms and this can bring noise in the system or inadequate results; *6) Building and going up on too many levels of generic terms* we will depart from the original sense of the word and we will get a much generic meaning. A number of 3 to 5 will be used;

These restrictions suggest that we can only focus in our presented algorithm on a) analyzing the words without the context b) only one POS type: NOUN or VERB. Because each word will have at least one superordinate, the set of superordinate words built based on the original document can be larger than the original document.

4 PROPOSED ALGORITHM TO EXTRACT GENERIC WORDS

Our proposed algorithm for defining keywords is focused on finding more general words (lexicographer files and hypernyms) from a document. Considering the document to have level 0, the algorithm defines a next level of more general words from the current level and so on. All resulted words, from all levels, will be reunited summing their weights. The number of levels is proportional to the level of generalizations we are looking for and the time algorithm will run. Each level will have attached a weight. By statistically analyze the whole set of words, on all levels and considering their weights we will have an image about the keywords of the document.

ALGORITHM 1: Proposed algorithm for extracting keywords

```
Clean up Document for punctuation marks. Keep only POS like noun and verbs
Extract Root Words
Build WordSetToBeProcessed as reunion of root words
SetOfSuperSenses ← Empty Set
Repeat 1-5 times
   // Starts a new level of analysis
   SetOfCurrentLevelSuperSenses ← Empty Set
   ForEach word from WordSetToBeProcessed
       SetOfWordSuperSense ← word SuperSense
       Add SetOfWordSuperSense to SetOfCurrentLevelSuperSenses
   End For
   Analyze SetOfCurrentLevelSuperSenses using level word weights
   Add SetOfCurrentLevelSuperSenses to SetOfSuperSenses and sum word weights
   WordSetToBeProcessed ← SetOfLevelSuperSenses
End Repeat
Process SetOfSuperSenses. Define Rule to extract relevant keywords
Apply Rule to extract relevant keywords
```

The important steps in Algorithm 1 are 1) for each word in WordSetToBeProcessed, we create a set of supersenses and append this set to SetOfCurrentLevelSuperSenses. To avoid noise in the system and multiple combinations we will focus on only one part of speech (noun or verb) and we will extract only the general word found as NOUN or VERB. To reduce the number of possibilities, the set of supersenses for each word can be truncate to the first value returned by WordNet API, considered to be the most relevant 2) Analyze SetOfCurrentLevelSuperSenses using statistics, add corresponding level weights to words in the SetOfCurrentLevelSuperSenses 3) on Process SetOfSuperSenses we apply various statistic processing on pair word:weight like counting, sorting, computing various statistical distribution and defining the rule that will be used. A rule example can be: select the 50% of words sorted by the biggest weight.

As mentioned, each word can have multiple hypernyms. The number of hypernyms used for each word is related to the algorithm precision (if only the first one returned by WordNet is used, considered to be the most relevant, with no respect to the word context) or noise introduced in the system (if we use all the word hypernyms).

The number of iteration levels can be combined with the level weight so that the levels close to the root word will have bigger weights thus favoring first levels of generalization.

The resulted keyword set can be used for: a) building a document summary using more general terms b) indexing the document based on more general terms c) ordering a set of documents by relevance, by defining a user profile based on some root words.

5 Experimental Results on extracting keywords

The sentence used for experiments is: A Rottweiler went to an animal shelter to rescue his poodle friend from the cage. To a human operator, the original sentence can refer, for example to animals and constructions, more to animals and less to constructions. After applying the first two steps in the algorithm and keeping only the nouns, we have the list of noun root words (WordSetToBeProcessed) for tests: rottweiler animal shelter poodle friend cage.

5.1 Lexicographer files approach

The particularities of this approach are that, when querying for word super-sense, the only returned values are 26 nouns (like: noun.animal, noun.cognition, noun.quantity etc) and 15 verbs (like verb.contact, verb.cognition, verb.emotion etc.).

By using the Algorithm 1 for the mentioned WordSetToBeProcessed, after only one iteration, we get the first and only set of nouns: noun.animal, noun.Tops, noun.artifact, noun.animal, noun.person, noun.artifact refered as SetOfCurrentLevelSuperSenses. There is only level so for this level of supersenses we select a weight of 1 for each word. The final SetOfSuperSenses with the word weights: noun.animal:1 noun.Tops:1 noun.artifact:1 noun.animal:1 noun.person:1 noun.artifact:1.

The step Analyze SetOfSuperSenses can be reduced to simply sum the weights for resulted words. After sorting the result alphabetically and by weight, we get noun.artifact with weight of 2, noun.animal 2, noun.person:1, noun.Tops:1. In this case, the rule for the extraction of the keywords consists in keeping the ones with weight more than 1. We can decide that we shall keep as keywords noun.artifact and noun.animal, with the same weight (importance). The result is close enough to the result we expect although it can be considered to be too general.

One of the result alteration reasons is the dog's supersense: noun.Tops. This is the only lexicographer file for dog and according to WordNet [18] it refers to animal, entity etc.

5.2 Hypernyms approach

When querying for word hypernyms the result is a) more appropriate to what a human operator expects to obtain, b) the meaning is close related to the original word c) it is very possible for a word to have multiple hypernyms.

For simplicity, we used in our experiment only the first hypernym returned by WordNet, considered to be the most

relevant and the word context was excluded. So, the SetOfWordSuperSense had only one word. We used 3 levels of iteration. Weights were chosen considering that the first level had a closer semantic relation with the original word than the third level. The first level will have a higher weight (4), second one 2 and the third one, weight 1.
The results for each iteration are:
The first iteration: word weight 4
WordSetToBeProcessed: rottweiler animal shelter poodle friend cage
SetOfCurrentLevelSuperSenses: shepherd_dog 4, organism 4, structure 4, dog 4, person 4, enclosure 4
SetOfSuperSenses: shepherd_dog 4, organism 4, structure 4, dog 4, person 4, enclosure 4
The second iteration: word weight 2
WordSetToBeProcessed: shepherd_dog, organism, structure, dog, person, enclosure
SetOfCurrentLevelSuperSenses: working_dog 2, living_thing 2, artifact 2, canine 2, organism 2, artifact 2
SetOfSuperSenses: artifact 4, canine 2, dog 4, enclosure 4, living_thing 2, organism 6, person 4, shepherd_dog 4, structure 4, working_dog 2
The third iteration: word weight 1
WordSetToBeProcessed: working_dog, living_thing, artifact, canine, organism, artifact
SetOfCurrentLevelSuperSenses: dog 1, object 1, whole 1, tooth 1, living_thing 1, whole 1
SetOfSuperSenses: organism 6, dog 5, artifact 4, enclosure 4, person 4, shepherd_dog 4, structure 4, living_thing 3, canine 2, whole 2, working_dog 2, object 1, tooth 1

On the step Process SetOfSuperSenses, the result was sorted by weights. Weights varied from 1 to 6. The rule was defined based on the biggest weights. An example of the rule to extract relevant keywords can be: keep only the biggest 25% of weights, more specific the weights bigger than 4.75. By applying the rule we get the keywords: organism, dog. This can be considered close to the expected result but the word organism is too general (the word organism emerges from processing the word animal in the original document which makes sense for a human operator).

By relaxing the rule and keeping the biggest 50% of weights, we get the keywords: organism 6, dog 5, artifact 4, enclosure 4, person 4, shepherd_dog 4, structure 4. The result is much closer to what we expect. The presence of the word person with a weight of 4 came from the word friend from the original document, mainly associated with humans (people).

The use of hypernym approach, will need at least one iteration in the Algorithm 1. The number of iterations is very important because a number that is too big will produce results which are too general. In our example, after 3 iterations some very general terms like object already emerge so a number of 5 iterations is reasonable.

Experimental results showed that: a) for a text which is too small there is the possibility that no relevant keywords can be extracted b) if the text is too big, the number of keywords which will be extracted is too big c) applying algorithm on text groups like paragraphs seems to lead to better and more focused results maybe because human tend to define paragraphs as an envelope for an idea.

6 CONCLUSIONS

The summarization of the text aim to extract a restricted number of keywords that will describe the text and introduce a certain level of generalization. The text will be cleaned up for punctuation and, by using WordNet API only root words for nouns and verbs will be extracted. To extract a set of keywords as more generic terms of the original text, this paper proposes an algorithm based on lexicographer file and hypernyms. On each level, a new set of generic terms is built based on the previous set of generic terms. Various statistical analysis can be applied on final word set in order to extract a set of document keywords. A more accurate result is obtained using hypernyms than lexicographer files. The resulted keywords can be used to identify, footprint, index, summarize and classify the document. Although efforts to build lexical databases for other languages have been provided, English remains the only language with notable results.

REFERENCES

[1] Cabré, M. T., Terminology: Theory, Methods, and Applications, ed. by Juan C. Sager, John Benjamins Publishing Company, Amsterdam, 1999
[2] Cătălin-Constantin Cerbulescu, Claudia-Monica Cerbulescu, Wordnet And Custom User Profile In Grouping Messages By Relevance, ICCC 2007
[3] E. Agirre, and O. Lopez, 2003. Clustering wordnet word senses Proceedings of the Conference on Recent Advances on Natural Language (RANLP'03)
[4] Enrico Giacinto Caldarola, Antonio M. Rinaldi, Improving the Visualization of WordNet Large Lexical Database through Semantic Tag Clouds, BigData Congress, 2016, DOI: 10.1109/BigDataCongress.2016.14
[5] Fellbaum, C. ed. (1998). WordNet: An Electronic lexical database, Cambridge, MA: The MIT Press (Language, speech, and communication series), 1998, xxii+423 pp; hardbound, ISBN 0-262-06197-X
[6] Finegan, E., Language: Its Structure and Use, Fifth Edition, Thomson Wadsworth, USA, 2008
[7] Gamallo, P., Gasperin, C., Agustini, A., Lopes, G.P. 2001: Syntactic-based methods for measuring word similarity. In TSD-01, Springer-Verlag (2001)
[8] Grefenstette, G. 1995: Evaluation Techniques for Automatic Semantic Extraction: Comparing Syntatic and Window Based Approaches. Corpus processing for Lexical Aquisition, MIT Press, Branimir Boguraev and James Pustejovsky (eds.) (1995) 205-216
[9] JRG Pulido, R Herrera, M Arechiga, A Block, R Acosta, S Legrand, 2006, Identifying Ontology Components From Digital Archives For The Semantic Web, Advances In Computer Science And Technology, Puerto Vallarta.
[10] Jyoti Yadav, Yogesh Kumar Meena, Use of fuzzy logic and wordnet for improving performance of extractive automatic text summarization, ICACCI 2016, 21-24 Sept. 2016, DOI: 10.1109/ICACCI.2016.7732356
[11] J. C. Lee, Yu-N Cheah, Paraphrase detection using semantic relatedness based on Synset Shortest Path in WordNet, Advanced Informatics: Concepts, Theory And Application (ICAICTA), 2016 Int. Conf. On, ISBN: 978-1-5090-1636-5
[12] Mohamed Ben Aouicha, Mohamed Ali Hadj Taieb, Sameh Beyaoui, Distributional semantics study using the co-occurrence computed from collaborative resources and WordNet, INISTA, 2016, DOI: 10.1109/INISTA.2016.7571831
[13] Saint-Dizier, P. (ed.) Predicative Forms in Natural Language and in Lexical Knowledge Bases, Springer-Science+Business Media Dordrecht, B.V., 1999.
[14] Sneha S. Desai, J. A. Laxminarayana, WordNet and Semantic similarity based approach for document clustering, CSITSS, 6-8 Oct. 2016.
[15] Steve Legrand, 2006, Word Sense Disambiguation with Basic-Level Categories, Advances in NLP Research in Computing Science 18, 2006, pp. 71-82
[16] Steve Legrand, JRG Pulido, 2004, A Hybrid Approach to Word Sense Disambiguation: Neural Clustering with Class Labeling. ECML and PKDD Pisa, Italy September 24, 2004.
[17] https://wordnet.princeton.edu/
[18] MIT Java Wordnet Interface, 2015, http://projects.csail.mit.edu/jwi/
[19] Alberto J. Cañas et al., 2003, Using WordNet for Word Sense Disambiguation to Support Concept Map Construction, Lecture Notes in CS 2857.