Towards a methodology for automatic identification of hypernyms in the definitions of large-scale dictionary

Inga Gheorghita[1,2], Jean-Marie Pierrel[1]
[1] ATILF, Université de Lorraine & CNRS, Nancy, France
[2] XILOPIX, 2 rue de Nancy, 88000 Épinal, France
inga.gheorghita@atilf.fr, jean-marie.pierrel@atilf.fr

Abstract

The purpose of this paper is to identify automatically hypernyms for dictionary entries by exploring their definitions. In order to do this, we propose a weighting methodology that lets us assign to each lexeme a weight in a definition. This fact allows us to predict that lexemes with the highest weight are the closest hypernyms of the defined lexeme in the dictionary. The extracted semantic relation "is-a" is used for the automatic construction of a thesaurus for image indexing and retrieval. We conclude the paper by showing some experimental results to validate our method and by presenting our methodology of automatic thesaurus construction.

Keywords: hypernymy, dictionary, thesaurus

1. Introduction

Linguistic resources such as dictionaries, computational lexicons, semantic taxonomies and thesauri are an important source of knowledge for natural language processing applications.

The information contained in linguistic resources, depending on their type, includes semantic relations (eg. *thrush* is a kind of *bird*), text definitions (eg. the Oxford dictionary defines the "lion" as "a large tawny-colored cat that lives in prides"), examples on the usage domain, and so on. Unfortunately, not all of which provide structured information that can be used by applications of natural language processing (Harabagiu, Miller, & Moldovan, 1999). A human understands the meaning of a word just by reading its definition in the dictionary, but it's not the case for a computer system. The main cause is that the semantic information, such as definitions, contained in the lexical resources is not very explicit and is provided in the form of free text.

Even WordNet (Miller, 1995), one of the most popular lexicons for the English language, uses definitions to explain the meaning of ambiguous words. However, compared with other electronic dictionaries and thesauri, which represent only an electronic transcription of their paper version, WordNet contains explicit information in the form of semantic relations such as, meronymy and hypernymy.

Over the last several decades much research has been done on the automatic construction of resources from corpora (Hearst, 1992), (Yarowsky, 1992), in particular by creating hypernym hierarchies. Various techniques as Machine Learning (Snow, Jurafsky, & Ng, 2005), Hidden Markov Model (Ritter & Soderland, 2009) and resources as dictionaries (Nakamura & Nagao, 1988) and thesauri (Kennedy & Szpakowicz, 2007) are used for identification of hypernymy or other semantic relations in the text.

In this paper, we aim to make explicit the information that is implicitly contained in the definitions of "Trésor de la Langue Française informatisé"[1] (TLFi) (Dendien & Pierrel, 2003). We are interested in determining from a definition of TLFi the hypernymy relation that will be used for automatic construction of a thesaurus for image indexing and retrieval (Gheorghita, 2011). More precisely, we determine the possible hypernyms for a particular dictionary entry by exploring its definitions. In order to do this, we propose a weighting methodology that lets us assign to each lexeme the weight it has in a definition. This fact allows us to predict that the lexemes with the highest weight are the closest hypernyms of the defined lexeme in the dictionary.

2. Hypernymy in lexical models

Hypernymy is a lexical function that for a term *t* associates one or more other general terms. Logical definitions (or Aristotelian) are generally composed of a "genus" and "differentiae". In most of the definitions of this type, the hypernymy is represented by the relation "is-a". *A* is a hypernym of *B* if *B* is an *A* (a kind / type / kind of *A*) and if *A* is a classifier of *B*. This means that concept *B* is a specialization concept of *A*, and concept *A* is a generalization concept of *B*. For example, « mammal » is a generalization of « lion, wolf».

In linguistic resources like thesauri, WordNet, lexical entries are linked to other lexical entries by semantic relationships, so in WordNet the entry for *big* would somehow represent that its antonym is *small*. In this type of lexical model the relations that a word has to others partly determine the word's sense. In dictionaries, the meaning of lexemes is divided into several parts (Murphy, 2010). The information necessary for determining the semantic relations among words in dictionaries is contained in theirs definitions. Thus, we can determine the semantic relations between lexemes by a set of rules such as "*A* is the hyponym of *B* iff it has the same components as B, plus at least one more".

Compared to WordNet, the TLFi defines the meaning of a word only by a definition. The single information that can disambiguate the meaning of an input of TLFi is the domain[2] of definition. But only 31% of definitions have a domain. The definitions without a domain are assigned

[1] Treasury of the French Language Computerized

[2] There are a total of 7 786 domains

to the "generic" domain. It means that the sense of the word is also valid in the other domains. The majority of definitions of TLFi for nominal entries are logical where usually the first word of the definition is the hypernym of the entry. In the TLFi the semantic relations are not explicit. To determine the possible hypernyms of a TLFi entry, we calculate the weight of each noun in the definitions for a given domain. We assume that the nouns with the highest weight are the best hypernyms of the TLFi entry.

3. The word weighting method in the dictionary definitions

Our approach based on the analysis of the structure, the size and the meta-language of dictionary definitions, has allowed us to define a weighting method, which estimates the importance of lexemes in a definition. Thus, to calculate the final weight of the lexeme, we take into account the importance of the lexeme in a definition (local weighting), the importance of the lexeme in the collection of definitions for a given domain (overall weight) and the position of the lexeme in the chain of characters of the definition.

The importance increases proportionally to the number of times a word appears in the definition, and to the number of times a word appears in the collection of definitions for the given domain but is offset by the position in the definition.

The weight of a term *t* in a definition *d* for the domain *D* is defined as follows:

$$p_{t=} \frac{freq(t,d)}{\sum_i freq(t_i,d)} * \frac{N(d_t,c)}{N(d,c)} * \log_2 \frac{N_{pos}}{N_{pos}(t,ch)}$$

where:

$freq(t,d)$: frequency of a term *t* in the definition *d*

$\sum_i freq(t_i,d)$: frequency of all terms t_i in the definition *d*

$N(d_t,c)$: number of definitions in the collection for the domain *D* that contain the term *t*

$N(d,c)$: number of definitions in the collection for the domain *D*

N_{pos} : number of positions in the string of a definition *d*

$N_{pos}(t,ch)$: number of position of term *t* in the string *ch* of a definition *d*

The position of the lexeme is a very important indicator since the definitions of the dictionary are written by lexicographers according to some rules and using a specific meta-language. In the definitions, the meta-language terms occurred very often. Their weight is quite high compared with the weight of the other lexemes. It is for this reason that we created the specific classes for each type of meta-language terms. This fact allows us to distinguish the meta-language term from the lexeme and to increase or decrease the weight of the lexeme in dependence of its position with the meta-language term. Contrary to other weighting formulas as TF.IDF (Spark Jones, 1972) which favor the discriminants and rarest terms, our goal is to give more weight to the lexemes located at the beginning of the definition, considered as class representatives, and to the discriminant terms in the collection of definitions for a given domain, considered as specific characteristics.

According to the hypothesis made before, that the term with the higher weight is considered to be a best hypernym for the input *e* of TLFi, the weighting method is used to determine the list of possible hypernyms for the given term *e*. Jointly used with the inclusion model, which defines a set of rules of inheritance of properties from one class by a subclass, we build a thesaurus as a hierarchical tree where the terms are related by the relation “is-a”.

4. Evaluation of results and discussion

We applied our approach to 132 743 definitions that correspond to 51 778 nominal entries in a dictionary.

Table 1 shows examples of possible hypernyms for dictionary entries ranked by their weight in the definition. We noticed that the lexeme with the highest weight is not always the best hypernym of the dictionary entry. It is usually a meta-language term like *family of*, *form of* or a lexeme very characteristic of a given domain like *system* for *medical domain* and *tribunal* for *law domain*. This fact is explained by their high frequency in the collection of definitions for the given domain. However, the lexeme that can be considered as the best hypernym, like *fruit* for *avocado*, is in the list of the first three possible hypernyms. To determine it precisely, the frequent terms must be filtered and eliminated from the list of possible hypernyms.

We evaluated the quality of our methodology, by using the structured definitions of TLFi within the Definiens project (Barque, Nasr, & Polguère, 2010). In these definitions, the semantic markers are a central component (CC) and peripheral components (CP), which have been annotated manually. We assumed that the lexemes, with the highest weight in the list of possible hypernyms, must be located in the central component of the structured definitions. To prove our hypothesis, we calculated the precision. The precision of our results is the proportion of lexemes with the highest weight in the definitions determined as the central components in the structured definitions of Definiens project. Since the Definiens project has not been finished yet, we could only test our hypothesis for 15 000 dictionary entries.

Figure 1 shows the precision for the first three lexemes of maximum weight.

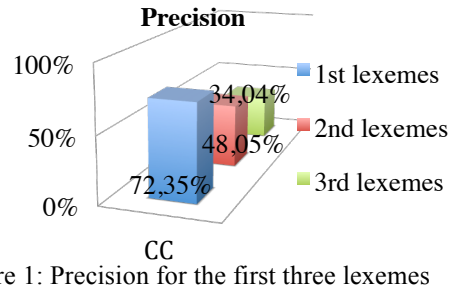

Figure 1: Precision for the first three lexemes

Avocat *lawyer*			Avocat *avocado*			Avocatier *avocado*		
k	Droit *Law domain*	Weight	k	Botanique *Botanical domain*	Weight	k	Botanique *Botanical domain*	Weight
1.	Tribunal *court*	0.0009	1.	Forme *shape*	0.006	1.	Famille *family*	0.038
2.	**Profession** *profession*	0.00018	2.	**Fruit** *fruit*	0.0009	2.	**Arbre** *tree*	0.02
3.	Intérêt *interest*	0.00013	3.	Poire *pear*	0.0002	3.	Région *region*	0.003
4.	Vie *life*	0.00011	4.	Pulpe *pulp*	0.00019	4.	Fruit *fruit*	0.002
5.	Ecrit *written*	0.00010	5.	Matière *flesh*	0.00017	5.	Lauracées *Lauraceae*	0.0003
6.	Barreau *bar*	0.00007	6.	Comestible *edible*	8.3E-5	6.	Nom *name*	0.00023
7.	Honneur *honor*	0.00006	7.	Avocatier *avocado*	4.72E-6	7.	Avocat *avocado*	7.9E-6
8.	Liberté *liberty*	0.00005						
9.	Justiciables *litigants*	0.00001						
10.	Eclairer *light up*	0.000005						
Belladone *belladonna*			Belladone *belladonna*			Aigle *eagle*		
k	Médecine *Medical domain*	Weight	k	Botanique *Botanical domain*	Weight	k	Générique *Generic domain*	Weight
1.	Système *system*	0.001	1.	**Plante** *plant*	0.13	1.	**Oiseau** *bird*	0.0004
2.	Sécrétion *secretion*	0.0006	2.	Famille *family*	0.03	2.	Famille *family*	0.0002
3.	**Alcaloïde** *alkaloid*	0.0004	3.	Partie *part*	0.002	3.	Taille *size*	0.00008
4.	Sensibilité *sensitivity*	0.0001	4.	Propriété *propertie*	0.0008	4.	Proie *prey*	0.00004
5.	Tonique *tonic*	1.3E-5	5.	Poison *poison*	0.00004	5.	Bec *bill*	0.00001
6.			6.	Atropine *atropine*	1.5E-6	6.	Bout *tip*	0.00001
7.			7.			7.	Envergure *span*	6.7E-5
8.			8.			8.	Tarse *tarsus*	5.5E-6
9.			9.			9.	Serre *claws*	4.5E-6

Table 1: Example of possible hypernyms ranked by their weight in the definitions of dictionary entries

For the first three possible hypernyms, we obtained a high precision that decreases with the rank of the hypernym. It proves that the first lexemes with the highest weight represent the best possible hypernyms for a dictionary entry.

These experiments demonstrate that our weighting methodology estimates correctly the importance of lexeme in a definition and allows us to determine with a best precision the first three possible hypernyms for the defined lexeme.

5. Exploitation of results

Extracted semantic relations from dictionary definitions are usually used to enrich existing taxonomies (Navigli & Velardi, 2008). Our aim is to use the hypernymy relations obtained from dictionary to construct a hierarchy of type "is-a".

In this section we present the algorithm of automatic construction of thesaurus, which is based on our methodology of hypernyms identification. The idea of the algorithm is to create a thesaurus from the words by using their definitions of TLFi. The algorithm is based on two processes. The first process aims to transform words into the thesaurus nodes. The second process allows those nodes created from the first process to be hierarchized.

5.1 Creation of nodes of the thesaurus

The objective of this process is the transformation of words in the thesaurus nodes. This process involves two steps:

a) Extracting data from the TLFi for a given word X.

For each given *word X* we get from the TLFi a list of data composed of the domains of its definitions, lexemes with their weights and positions in the definitions of each domain.

b) *Transformation of the data list of a given word X in nodes of the thesaurus*.

From the data extracted for a given *word X* we proceed to the creation of nodes of thesaurus. A node is a data structure. We distinguish 3 types of nodes: domain node, word node and lexeme node. Depending on the type of node, the data structure is different. For the domain node the data structure is limited to the domain name and its identifier. The structure of the word node consists only of word and that of the lexeme node contains the word, the lexeme, its weight, its position, the identifier of the definition, the domain and its identifier. Thus, we consider as the nodes of the thesaurus each *word X* as well as its domains and extracted lexemes.

5.2 Construction of hierarchy of nodes

The goal of this process is to organize in a hierarchy the created nodes. The hierarchical tree is built by comparing the data structures of each node with the other. This process is realized in several steps:

a) Determination of parent nodes for a given word X node.

Using the created nodes, we determine the parent nodes of word X node. To do this, we group the lexeme nodes by their definition identifier and for each created group we determine the node whose weight is maximum. This node becomes the parent node for word X node. Thus, several different parent nodes (as much as different definitions of the word in the TLFi) are created having as child node the word X node. Then the following steps are executed:

If in a group there are two nodes of the same maximum weight, but having different lexemes
  then the two nodes become different parent nodes for word X node.

If for two groups, emerge two nodes of the same maximum weight having the same domain and lexeme
  then
    if these nodes have the same position
      then only one node becomes the parent node of the word X node;
    else we determine for each group the second node whose weight is maximum and these nodes become parent nodes of word X node.

b) Determination of child nodes for the created parent nodes.

In order to determine the child nodes of the created parent nodes, we proceed to the creation of the other nodes. New nodes are created from existing lexemes nodes. This procedure consists of executing the first process (5.1) for each lexeme node. The structures of new obtained nodes (named lexeme nodes II) are compared with the created parent nodes by identifying the lexeme nodes II, which have the same lexeme as the created parent nodes.

If such nodes are found, we check:
  if their position is minimal (1-3) and they have the same domain as the parent nodes or generic domain
    then these lexeme nodes are replaced by words nodes corresponding to words of lexemes nodes and they become the child nodes of the created parent node.

c) Transformation of child nodes in the parent nodes of word X node.

In order to allow the growth of thesaurus in depth we determine for word X node the new parent nodes. Thus, we compare the parent nodes with lexemes nodes II.

If the existing lexeme nodes II have the same lexeme as the word of the word X node and the same word as the child node, we check:
  if their position is minimal (1-3) and they have the same domain as the word X node or generic domain
    then these nodes become parent nodes of word X node.

d) Determination of the hierarchy for domain nodes.

For each domain node, we determine its parent node by exploiting the thesaurus[3] of the TLFi's domains. Thus, we compare domain nodes with parent nodes determined during the step one of the second process. If these nodes have the same domain, then the domain nodes become parent nodes for these.

e) Assignment of associative nodes to the word X node.

The constructed thesaurus will be used for indexing and search of images. Thus, the associative nodes are the nodes that during the search of images will be used to direct the user to nodes situated at the bottom of the hierarchy. Associative nodes are nodes that have not been used for the creation of the thesaurus and, compared to other nodes of the thesaurus, they are not used for indexing.

The assignment of these associative nodes corresponds to the following process:

If the domains and the definitions' identifiers of lexeme nodes not used for the creation of the thesaurus are identical to those of the parent node of the word X node
  then these nodes are assigned to the word X node.

6. Conclusion

In this paper, we have proposed a new methodology to calculate the weight of lexemes in dictionary definitions. We have showed that our method allows us to determine precisely the possible hypernyms for the defined lexeme. The first evaluation of our method has given the best precision of 72,35% for the first hypernyms, which weight is the highest in the definition.

The utility of our approach is that it can be used to determine the hypernymy relation in the machine-readable dictionaries where usually the semantics relations are not explicit. Thus, based on determined hypernymy relation we have presented our algorithm of automatic construction of thesaurus using dictionary definitions. The constructed thesaurus will allow the disambiguation of the sense of words by improving the precision of the image search. For example, the system will be able to provide for the query "ananas" 3 types of images corresponding to 3 senses (plant, fruit, color) of the lexeme *ananas* in the TLFi.

Currently, we are working on the implementation of the algorithm of automatic construction of thesaurus. This fact will allow a second evaluation of our methodology for automatic identification of hypernyms in the

[3] This resource was created by normalizing the domains of dictionary and using the documentation on the thesaurus of techniques of TLFi, that contains all domains and subdomains used during the writing of TLFi's definitions.

definitions of dictionary. We also plan to compare the extracted hypernyms with those already available in the existing thesauri or computational lexicons.

With the addition of some improvements such as the filtration of meta-language terms, we believe that this automatic method of identifying hypernyms will allow us to construct a truly hierarchical tree of type "is-a".

7. References

Barque, L., Nasr, A., & Polguère, A. (2010). From the Definitions of the Trésor de la Langue Française to a Semantic Database of the French Language. *Proceedings of the 14th EURALEX International Congress*. Leeuwarden.

Dendien, J., & Pierrel, J.-M., (2003). Le Trésor de la Langue Française Informatisé: un exemple d'informatisation d'un dictionnaire de langue de reference. *TAL (Traitement Automatique des Langues),* 44(2), Hermès Sciences Edition, pp. 11--37. Paris.

Gheorghita, I. (2011). Méthodologie de construction automatique du thésaurus pour l'indexation et la recherche des images. *TALN&RECITAL*, *2*, pp. 221--228. Montpellier.

Harabagiu, S., Miller, G. A., & Moldovan, D. (1999). WordNet 2 - A Morphologically and Semantically Enhanced Resource. *Actes de SIGLEX'99*, pp. 1--8.

Hearst, M. A. (1992). Automatic acquisition of hyponyms from large text corpora. *Proc 14th Conference on Computational linguistics*, pp. 539--545.

Kennedy, A., & Szpakowicz, S. (2007). Disambiguating hypernym relations for Roget's thesaurus. *TSD'07 Proceedings of the 10th international conference on Text, speech and dialogue*. Heidelberg : Springer-Verlag Berlin.

Miller, G. A. (1995). WordNet: A lexical Database. *Communications of the ACM*.

Murphy, L. M. (2010). *Lexical Meaning*. Cambridge: University Press.

Nakamura, J., & Nagao, M. (1988). Extraction of semantic information from an ordinary en- glish dictionary and its evaluation. *Proceedings of the 12th Conference on Computational linguistics,* pp. 459--464. Morristown: Association for Computational Linguistics.

Navigli**,** R., & Velardi, P. (2008). From Glossaries to Ontologies: Extracting Semantic Structure from Textual Definitions. In *Ontology Learning and Population: Bridging the Gap between Text and Knowledge* (P. Buitelaar and P.Cimiano, Eds.), Series information for Frontiers in Artificial Intelligence and Applications, IOS Press, pp. 71--87.

Ritter, A., & Soderland, S. (2009). What Is This, Anyway: Automatic Hypernym Discovery. *Proceedings of the 2009 AAAI,* pp. 88--93. Spring.

Snow, R., Jurafsky, D., & Ng, A. Y. (2005). Learning syntactic patterns for automatic hy-pernym discovery. *NIPS*.

Spark Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation, 28*(1), pp. 11--20.