Automatic Selection of Defining Vocabulary in an Explanatory Dictionary*

Alexander Gelbukh and Grigori Sidorov

Center for Computing Research (CIC),
National Polytechnic Institute (IPN),
Mexico City, Mexico.
{gelbukh, sidorov}@cic.ipn.mx

Abstract. One of the problems in converting a conventional (human-oriented) explanatory dictionary into a semantic database intended for the use in automatic reasoning systems is that such a database should not contain any cycles in its definitions, while the traditional dictionaries usually contain them. The cycles can be eliminated by declaring some words "primitive" (having no definition) while all other words are defined in terms of these ones. A method for detecting the cycles in definitions and selecting a minimal (though not the smallest) defining vocabulary is presented. Different strategies for selecting the words for the defining vocabulary are discussed and experimental data for a real dictionary are presented.

1 Introduction

A natural method to define the meaning of the words for an automatic reasoning system is to define some words through other words, the way it is done in the traditional explanatory dictionaries. To build such definitions, automatic conversion of existing explanatory dictionaries into "computer-oriented" dictionaries looks attractive. However, existing human-oriented dictionaries have a feature that does not permit to directly use them as logical systems: their definitions have logical cycles. For example:

(1) bee: an insect that produces honey.
honey: a substance produced by bees.[1]

There can appear longer cycles: a word *a* is defined though a word *b*, which is defined through a word *c*, etc., which is defined through the word *a*. Obviously, any dictionary that defines all words it mentions must contain cycles; thus, cycles are an inevitable feature of a human-oriented dictionary that tries to define all words existing in the given language.

* Work done under partial support of CONACyT, CGEPI-IPN, COFAA-IPN, and SNI, Mexico. Implementation by Gabriela Rivera-Loza. We thank Graeme Hirst and Ted Pedersen for useful discussion.

[1] This is a slightly simplified real example from the Anaya dictionary of Spanish: "*abeja: insecto que segrega miel*"; "*miel: sustancia que producen las abejas*."

To convert such a dictionary into a "computer-oriented" logical database for a reasoning system, the cycles are to be eliminated by declaring some words "primitive," i.e., not defined in this dictionary (their meaning is to be stated in a different way).

For example, in school geometry it is possible to expand the definition of any term (say, *bisectrix*) substituting any word in its definition (say, *angle*) with its definition. This system of definitions is constructed in such a way that if such substitution is repeated iteratively, the definition of any term can be expanded into a (very long) definition composed only of the primitive concepts and logical operators. For school geometry, the primitive concepts are *point*, *line*, and *incidence*: these words do not have any definitions in the formal logical system of geometry.[2] We call such a set of primitive words *defining vocabulary*.

DEFINITION. Given a dictionary D, a *defining vocabulary* for D is a set of words such that if they are declared primitive (i.e., their definitions are removed from the dictionary), the rest of the dictionary does not contain cycles.[3] A defining vocabulary is *minimal* if no its subset is a defining vocabulary. A defining vocabulary is the *smallest* if there is no defining vocabulary for D consisting of a smaller number of words.

Indeed, for the same set of words, different defining vocabularies can be chosen. For instance, in the example (1) above, either *bee* can be chosen primitive and *honey* defined, or vice versa. Without going deep into discussion about the nature of semantic primitives (see, for example, [2–4]) or defining vocabulary (such as Longman defining vocabulary), we just note that the problem of selection of a defining vocabulary has so far no widely accepted theoretical solution. For the purposes of this paper it is important that since the meaning of the primitive words is explained in an "expensive" way (say, procedurally), it is highly desirable to minimize their number.

In this paper we present an algorithm that, given a dictionary, selects a minimal defining vocabulary for it. Our method, though, does not build the smallest defining vocabulary (this is the topic of our current investigation).

Below we present the algorithm, then discuss four strategies it can use, and compare them basing on the experimental data obtained with a large Spanish dictionary.

2 Algorithm and Strategies for Selection of Defining Vocabulary

We represent the dictionary as a directed graph $\boldsymbol{G}$, where the nodes are words[4] and there is an arc from w_1 to w_2 iff w_2 is used in the definition of w_1. Since $\boldsymbol{G}$ has cycles,

---

[2] Their meaning is explained to the students (the "users" of this formal system) by examples or procedurally (showing how to *draw a line* or how to observe that *two lines are incident*).

[3] The formal way we use the term "primitive word" does not completely correspond to the traditional use of this term in semantics [4]. In particular, the words selected by our algorithm as primitive might not be acceptable semantic primitives for a linguist.

[4] By a word, (1) literal string, (2) lemma, or (3) specific word meaning can be understood. The former variant nearly does not make sense. The results obtained for the latter two variants are very similar. Here we present the third variant (words as specific meanings). We used a disambiguation procedure similar to the Lesk algorithm [1].

our task is to select a minimal (not necessarily the smallest) set $\boldsymbol{P}$ of nodes such that removing from $\boldsymbol{G}$ all arcs leaving these nodes makes the resulting graph $\boldsymbol{G'}$ acyclic.

Initially, both $\boldsymbol{P}$ and $\boldsymbol{G'}$ is empty. We consider the nodes of $\boldsymbol{G}$ one by one in some order (see below) and insert them into either $\boldsymbol{P}$ or $\boldsymbol{G'}$: if insertion of the node (and all arcs incident to it in $\boldsymbol{G}$) into $\boldsymbol{G'}$ does not cause any cycles in it,[5] the node is inserted into $\boldsymbol{G'}$, otherwise into $\boldsymbol{P}$. At the end of this process, $\boldsymbol{G'}$ is the desired acyclic graph and $\boldsymbol{P}$ is the corresponding minimal defining vocabulary.

There are different possible strategies to define the order of consideration of the nodes of $\boldsymbol{G}$ in the algorithm. The nodes considered first tend to belong to $\boldsymbol{G'}$, while the ones considered last tend to belong to $\boldsymbol{P}$, i.e., to be declared primitive (cf. example (1) above: if *honey* is considered first, *bee* is declared primitive, and vice versa). In our experiments we used four different strategies.

Strategy 1: *random, uniform*. At each iteration, the next node is chosen randomly (with a uniform distribution) from the nodes not yet processed.

Strategy 2: *by frequencies*. The nodes are ordered by the number of incoming arcs (i.e., frequencies in the definitions[6]), from smallest to greatest. We expected that with this, $\boldsymbol{P}$ would be smaller because the chosen nodes break more cycles in $\boldsymbol{G}$.

Strategy 3: *random, by frequencies*. This is a combination of 1 and 2. The random order is used, but with distribution inverse to the frequencies. We expected that some alternations of the rigid order of method 2 might produce better results.

Strategy 4: *by random voting*. $N = 20$ different sets $\mathbf{P}_i$ were generated with the strategy 1. Then for each node w we counted the number $0 < n(w) < N$ of sets $\mathbf{P}_i$ to which it belonged. In the algorithm, we considered first the nodes w with $n(w) = 0$ in the order of their frequencies, as in the strategy 2. Then we considered the rest of the elements in the order inverse to $n(w)$, in each group with the same $n(w)$ using the order inverse to the frequencies. We expected that the nodes with a greater $n(w)$ were better defining words and, thus, should belong to the best defining set.

4 Experimental Results

We applied these strategies to a large explanatory dictionary of Spanish (Anaya dictionary). Before this, auxiliary words had been removed and the content words in the definitions had been lemmatized, POS-tagged, and marked with sense numbers; see [1] for details. The dictionary contained 30725 headwords; 10359 words were used in the definitions (i.e., these words had incoming arcs).

Table 1. Number of primitives obtained with different strategies.

Strategy	Size of $\boldsymbol{P}$
1. Random, unified	2789, $s = 25$
2. By frequencies	2302
3. Random, by frequencies	2770
4. By random voting	**2246**

[5] A special data structure is used to quickly verify this, which guarantees linear complexity of the whole process.

[6] Without repetitions in the same definition, i.e. several occurrences of a word in the same definition is counted as one occurrence.

Table 1 shows the experimental results. The values given for the first strategy are the average calculated during 20 experiments and the mean quadratic deviation *s*. It is interesting that the deviation is rather small, which means that there is little difference in size of the sets generated in different experiments according to this strategy. As one can see, the best strategy is 4.

Surprisingly, the size of the defining vocabulary we found is very near to 2000, which is considered an approximate number of primitives in human languages. For example, this is the size of the Longman Defining Vocabulary, the number of basic glyphs in Chinese, etc.

5 Conclusions and Future Work

We have presented a method for selecting, given a dictionary D, a minimal (though not the smallest) defining vocabulary. The strategies of ordering to be used in the algorithm have been discussed.

Construction of defining vocabularies is helpful in analysis of the structure of dictionaries by the lexicographers, in particular, to evaluate the optimality of relations between words in the dictionary.

The possible future work is the following:

- To give a linguistic interpretation of the obtained defining vocabularies,
- To elaborate linguistic (semantic, rather than statistical) criteria of preferences of inclusion of words into the defining vocabulary (see footnote 3 above),
- To improve the algorithm to find the smallest defining vocabulary; different techniques—for example, genetic algorithms—can be used,
- To develop the software that would allow using the obtained information to help lexicographers in improving (traditional) dictionaries.

References

1. Gelbukh, A., and G. Sidorov (2001). Algorithm of word sense disambiguation in an explanatory dictionary. Proc. of *COMPLEX-2001, Workshop on Computational Lexicography*, Birmingham, Great Britain, June 28-30, 2001, pp. 35-40.
2. Kozima, H., and A. Ito (1997). Context-sensitive word distance by adaptive scaling of a semantic space. In: Mitkov, R. and , N. Nicolov (eds.). *Recent Advances in Natural Language Proceedings: Selected Papers from RANLP'95*, pp. 111-124.
3. Saint-Dizier, P., and E. Viegas (eds.). (1995) *Computational lexical semantics*. Cambridge: Cambridge University Press, 447 p.
4. Wierzbicka, A. (1996) *Semantics: Primes and Universals*. Oxford: Oxford Univ. Press.