On Similarity of Word Senses in Explanatory Dictionaries*

Alexander Gelbukh[1], Grigori Sidorov[1], and Yoel Ledo-Mezquita[1,2]

Center for Computing Research (CIC),
National Polytechnic Institute (IPN),
Av. Juan de Dios Bátiz, esq. Miguel Othón de Mendizábal,
Mexico D. F., Zacatenco, CP 07738, Mexico
{gelbukh,sidorov,ledo}@cic.ipn.mx, www.gelbukh.com

[2] Telematics Department, CUJAE, Cuba
ledo@tesla.ispjae.edu.cu

Abstract. Quality machine translation (MT) as well as of some other applications (such as information retrieval, IR) require word sense disambiguation (WSD) in the source text. However, WSD is only possible if the word senses specified in the dictionaries are really different and clearly distinguishable. We investigate the semantic closeness of different senses of the same word in a Spanish explanatory dictionary. We define the closeness between two senses as the relative number of equal or synonymous words in their definitions. We show that a considerable part of dictionary definitions (ca. 90%) are different enough to be distinguished in MT and IR. On the other hand, a considerable number of definitions (ca. 10%) are too similar to be reliably distinguished. These results suggest that MT and IR can take advantage of WSD algorithms, but for this, the similar senses reflecting too subtle meaning nuances should be clustered together to form coarser but easier distinguishable senses. The proposed method for detecting too similar senses can be incorporated into the lexicographer's workbench to be used in development and improvement of dictionaries.

1 Introduction

Words in a typical explanatory dictionary have different meanings (senses); this phenomenon is known as polysemy. However, in real texts words appear in a specific sense. The problem of choosing a specific word sense for a given word occurrence in the text is known as word sense disambiguation (WSD) problem. Various WSD methods discussed in literature can be classified into statistical [1, 5, 7, 12] and knowledge-based [4, 8, 10, 6] ones. The WSD techniques are believed to be useful in improving the quality of machine translation (MT) and other tasks related to information processing, such as information retrieval (IR).

Our motivation for this research was the following. We planned to compile a bilingual dictionary by (semi-)automatic alignment of the definitions in two explanatory dictionaries, one Spanish and another one English. The idea was to compare each pair of definitions in these dictionaries (using an existing Spanish-English dictionary) and to consider mutual translations the words that match best. Of course, actually a unit of

* Work done under partial support of Mexican Government (CONACyT, SNI, SRE), IPN (CGPI, PIFI, COFAA), and CyTED (RITOS-2).

comparison is not a word but a word sense. However, such an idea would work only if the senses in the dictionaries are clearly distinguishable, at least with respect to the similarity measures we used. Thus, we were interested in the question of whether, and how many of, the definitions in the given explanatory dictionary are clearly distinguishable.

The same question is important to verify the intuition that underlies the classic scheme of MT, which consists in that the first step in translation is the (automatic or manual) choice of the correct sense of each word in the source language, with a subsequent lookup of the translation of the found sense in the bilingual dictionary.

Similar question arises in other applications. As far as IR is concerned, given a query *bill*$_1$ 'financial document', the system aware of word senses would not present to the user the documents about *bill*$_2$ 'garden instrument', *bill*$_3$ 'part of a bird', etc. However, for very similar senses their discrimination would only compromise recall rather than improving precision. For example, for the query *solution*$_1$ 'result of solving a problem' the documents containing *solution*$_2$ 'process of solving a problem' are also relevant, so that an IR system should not distinguish such sense nuances. How often are the former and the latter situation? That is, to what extent WSD is really useful for IR?

In this paper, we investigate the semantic closeness of different senses of the same word. We define the closeness between two given senses through the number of equal or synonymous words in their definitions in the explanatory dictionary (in our experiments we used the Anaya explanatory dictionary of Spanish). By equal words we mean equal lexemes, i.e., we do not distinguish different morphological forms (we consider *go*, *goes*, *went*, *gone*, *going* to represent the same word). For detecting synonyms, we use a Spanish synonym dictionary.

We assume that the word senses that are close in our similarity measure express the same (or very similar) information content (meaning). We assume also that the users or the MT or IR systems would have difficulty specifying which of the two similar senses satisfies the information need. Similarly, WSD algorithms would frequently confuse such senses. Therefore, distinguishing between such similar senses would only compromise the recall of an IR system or create redundancy in a MT system.

On the other hand, we assume that the word senses that are distant in our measure are easily distinguishable by the humans, MT or IR applications, and WSD algorithms. Distinguishing between such senses would improve the MT quality and IR system's precision.

We show that a considerable part of dictionary definitions fall in the former category: they should not be distinguished. On the other hand, another considerable part of dictionary definitions fall in the latter category and thus should be distinguished. This presents an argument for usefulness of WSD for MT and IR given that too detailed word senses listed in a traditional dictionary are clustered into coarser senses.

We present the algorithm (and software) that can automatically calculate the semantic closeness. It can be, for example, incorporated into lexicographer's workbench: the algorithm can show the potentially similar senses to the lexicographer, who can then make a decision about merging the similar senses or changing their definitions.

In the rest of the paper, we first discuss the data (dictionary) used in our experiments and the experimental methodology, then present and discuss the obtained results, and finally draw some conclusions.

Table 1. Distribution of number of senses per word.

Words	Senses	Words	Senses	Words	Senses	Words	Senses
13077	1	85	9	6	15	2	17
8884	2	54	10	5	21	1	44
4103	3	33	11	4	18	1	41
1792	4	17	12	3	22	1	35
774	5	16	13	3	19	1	28
426	6	13	14	2	30	1	27
235	7	7	16	2	29	1	24
144	8	6	20	2	23		

2 Data Description

We used the Anaya explanatory dictionary of Spanish as a source for words and their senses. We preferred this dictionary to Spanish WordNet [11] because Spanish WordNet has definitions in English while the linguistic tools (such as stemmer and tagger) at our disposal work with Spanish. For stemming, we used our Spanish morphological analyzer and generator [1].

This dictionary has more than 30 thousand headwords, with more than 60 thousand senses in total. The distribution of number of senses per word is presented in Table 1.

All definitions in the dictionary were normalized and part-of-speech (POS)-tagged as described in [9]. Here is an example of a normalized and POS-tagged definition. The original definition of one of the senses of the word *abad* 'abbot' was as follows:

Abad: *Título que recibe el superior de un monasterio o el de algunas colegiatas*.
'Abbot: Title that receives a superior of a convent or of some churches.'

The normalized version of this definition is as follows:

Abad = $título_{noun}$ que_{conj} $recibir_{verb}$ el_{art} $superior_{noun}$ de_{prep} un_{art} $monasterio_{noun}$ o_{conj} el_{art} de_{prep} $alguno_{adj}$ $colegiata_{noun}$. $_{punct}$

'Abbot = $title_{noun}$ $that_{conj}$ $receive_{verb}$ a_{art} $superior_{noun}$ of_{prep} a_{art} $monastery_{noun}$ or_{conj} of_{prep} $some_{adj}$ $church_{noun}$.'

where *conj* stands for conjunction, *art* for article, *prep* for preposition, *adj* for adjective, and *punct* for punctuation mark. There were some words (about 3%) that were not recognized by our morphological analyzer; we marked such words as *unknown*.

We also used a synonym dictionary of Spanish that contains about 20 thousand headwords. This dictionary is applied at the stage of measuring of the similarity between senses for detecting synonymous words in definitions; see Section 3.

For comparison, we ignored the auxiliary words because usually they do not add any semantic information.

We considered homonymous headwords as different words (groups of senses) rather than different senses of the same word, as they are represented in the dictionary. Homonyms usually have very different meanings, so we supposed all senses of one homonym should be considerably different from those of another one.

3 Experimental Methodology and Main Algorithm

In the experiment, we measured the similarity between two different word senses of the same word. We used the standard measure of similarity between two texts analogous to the well-known Dice coefficient [2, 8]. Dice coefficient is defined as follows:

$$D(t_1, t_2) = 2\frac{|W_1 \cap W_2|}{|W_1| + |W_2|} \quad \textbf{(1)}$$

where W_1 and W_2 are sets of words in the texts t_1 and t_2, respectively. This value characterizes the relative number of the words the texts have in common. In this case, the words are compared literally.

However, we wanted also to take into account synonyms of the words from the two texts (the two sense definitions in question). Different ways to treat the synonyms can be suggested:

- Treat the synonyms in the same way as morphological forms of the words, or
- Weight the synonyms with some coefficient K, or
- Ignore the synonyms.

The middle option can be realized by the following modification of (1):

$$S(t_1, t_2) = \frac{|W_1 \cap W_2| + K|W_1 \circ W_2|}{\max(|W_1|, |W_2|)} \quad \textbf{(2)}$$

where the symbol "$\circ$" denotes the intersection through synonyms, see below, and K is a weighting coefficient. We use the maximum value for normalization since all words from the larger definition can be synonyms of the words from the other one. Note that the first option above can be thought of as the use of the weighting coefficient $K = 1$ and the last one as the use of $K = 0$.

Given this similarity measure, the processing procedure for our experiments was as follows. For each word in the dictionary, we measured the similarity between each pair of its senses (we ignored the words with only one sense). Note that our similarity measure is symmetric, so we calculated it only once for each sense pair.

Given two senses (dictionary definitions) s_1 and s_2, we measured the similarity between them as follows; see Fig. 1. At the beginning, the similarity score is set to the number of equal significant words in the two definitions. By equal words, we consider the words in any morphological form, i.e., we only required that the stem and the part of speech be equal. For this, we used a stemmer and a POS tagger. We took into account only significant words and discarding auxiliary words (prepositions, etc.). Unknown words were processed as significant words provided that their length was greater than a given threshold. We used the threshold equal to two letters, i.e., nearly all unknown words participate.

An auxiliary list of words was used to prevent the same word from being counted twice. When a word was counted, it was added to the list; before counting a word, it was checked that it is not already on the list.

For a word present only in one definition, we searched for its synonyms in the other one using a dictionary of synonyms. If a synonym was found, the score was incremented by the weighting coefficient K. In this case, the synonym was also checked against the list of already counted words to avoid repetitions and added to this list if it was not yet there.

```
ForEach word from the dictionary
    ForEach pair of its senses s1, s2
        list = s1 ∩ s2;
        score = the number of significant words in list;
        call Synonyms (s1, s2);
        call Synonyms (s2, s1);
        n1 = the number of significant words in s1;
        n2 = the number of significant words in s2;
        similarity (s1, s2) = score / max (n1 , n2);

procedure Synonyms (sa, sb):
    ForEach significant word w ∈ sa
        ForEach synonym u of w
            If u ∈ sb and u ∉ list then
                score = score + K;
                Add u to list;
```

Fig. 1. Calculation of the similarity between pairs of senses.

After the synonyms of the words from the first definition have been looked for in the second one, we apply the same procedure to look for the synonyms of the words from the second definition in the first one. This is because we cannot guarantee that our synonym dictionary is symmetrical, i.e., that if the word *A* is synonym for *B* then *B* is synonym for *A*. Probably it should be so in an ideal synonym dictionary, but in a real-world dictionary, it is not always the case. However, even if it is so, this second step does not result in wrong matches due to the use of the list to avoid repetitions.

Finally, we apply the formula (2) to calculate the similarity. The program implementing this algorithm takes less than 3 minutes for Anaya dictionary.

4 Experimental Results and Discussion

We conducted three experiments with different weighting coefficients, i.e. the different manner of treating synonyms, according to the options listed in Section 3. In the first experiment, we did not count the contribution of synonyms in the similarity ($K = 0$) and in other two experiments we counted the synonyms with coefficients $K = 0.5$ and $K = 1$, respectively. The results of the experiments are shown in Table 2 and represented graphically in Fig. 2.

Table 2. Number of sense pairs for different contribution of synonyms.

Interval of distances	$K = 0$		$K = 0.5$		$K = 1$	
	Number	%	Number	%	Number	%
Exactly 0	57257	83.56	46205	67.45	46205	67.43
0.01 to 0.25	8423	12.29	18240	26.62	14725	21.49
0.25 to 0.50	2675	3.90	3853	5.62	6655	9.71
0.50 to 0.75	153	0.22	205	0.30	600	0.88
0.75 to 1.00	13	0.02	18	0.03	336	0.49

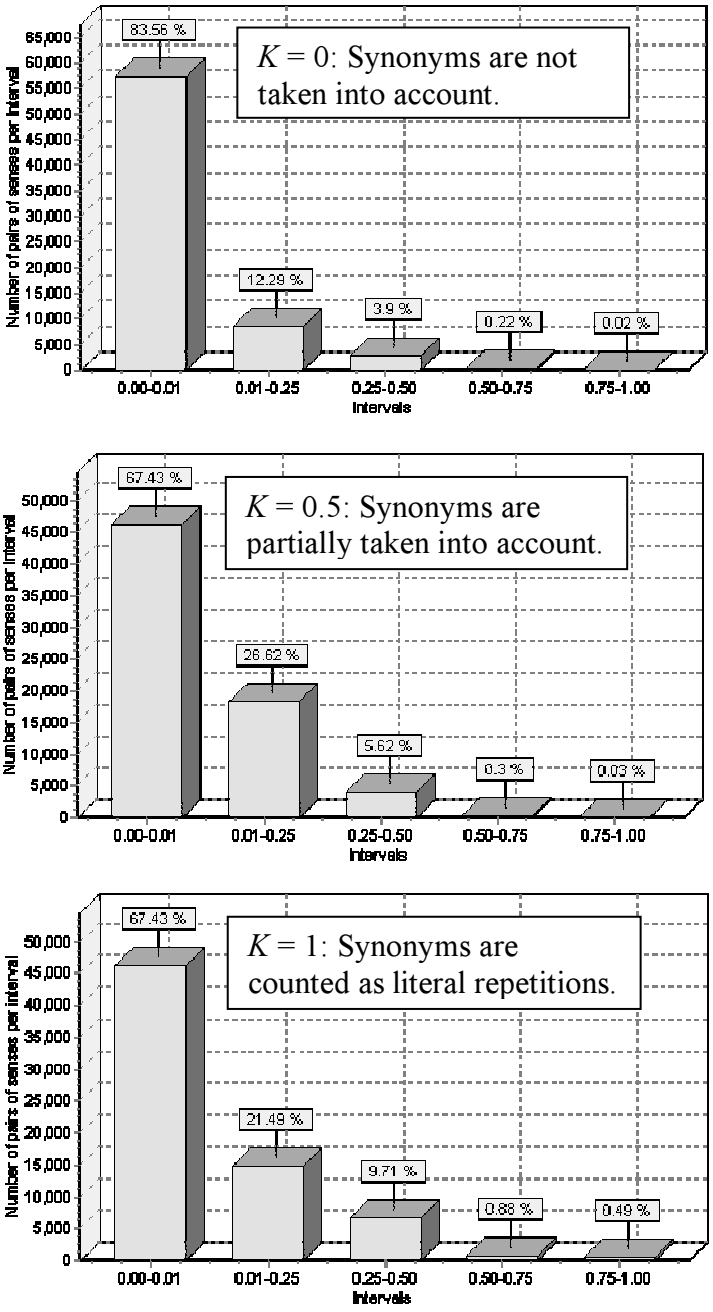

Fig. 2. Number of sense pairs for different contribution of synonyms.

Since the similarity is a fraction, some fractions are more probable than others; specifically, due to a high number of short definitions, there were many figures with small denominators, such as 1/2, 1/3, 2/3, etc. To smooth this effect, we represented the experimental results by intervals of values and not by specific values. We use four equal intervals for the percentage of similarity between senses. We present the results for zero similarity separately because we knew in advance that there were many senses without any intersection.

It can be observed that synonyms had important contribution to the similarity between word senses. In our opinion, synonyms should be used in such calculation

because they represent basically the same meaning and differ only in outer shape. Thus, though usually one synonym cannot be substituted with another in the text, they do express more or less the same user's information need. Thus, we believe that the weight $K = 1$ is most adequate, though we do not have solid arguments to prefer some specific value.

Another problem is the interpretation of the intervals: what values correspond to "far" or "near", i.e., "similar" or "different" senses. We suggest that the values greater than 0.5 indicate very similar senses. Indeed, in this case at least 50% of the words used in both definitions are synonymous or identical. The values between 0.25 and 0.50 can be considered indicating substantial similarity, because in this case at least 25% of words are equivalent. Smaller vales indicate little similarity.

The experimental data show that the majority of sense pairs (67%) do not intersect, i.e., are different, and only about 1% are very similar. Still relatively large number of pairs has significant similarity – about 10%. Another 21% have little similarity. Therefore, as we have discussed in Section 1, applying WSD in MT is obligatory and IR systems can benefit from WSD applied to the majority of senses, which are different. However, a considerable part of the pairs of senses in the dictionary, that we experimented with, should not be distinguished by WSD procedures. The presented algorithm can show to the lexicographer the potentially similar senses so that he or she can make a decision on merging the similar senses or changing their definitions.

There is no need to give an example of different senses since these are simply the senses that have neither common significant words nor synonyms. An example of similar senses is:

Agobiarse ('to worry too much, get worked up'):

1. $causar_{\text{verb}}$ $molestia_{\text{noun}}$ o_{conj} $fatiga_{\text{noun}}$ ('cause bother or fatigue'),
2. $causar_{\text{verb}}$ $angustia_{\text{noun}}$ o_{conj} $abatimiento_{\text{noun}}$ ('cause anguish or depression').

The similarity between these two senses is 2/3 = 0.67. Indeed, both *molestia* ('bother, trouble') and *fatiga* ('fatigue') have the synonym *angustia* ('anguish, affliction, distress'), but it is counted only once in the first definition. The verb *causar* is literally repeated, so 2 words (not counting the auxiliary words) of 3 are similar.

These senses are indeed very much similar, so that it is difficult to imagine the context in which they would be easy to distinguish even for a human.

5 Conclusions

We have defined a similarity measure for short texts and applied it to dictionary definitions of different senses of the same word. We found that a considerable part of the senses in the dictionary we used for our experiments were too similar to justify their separation. Thus, for more reliable application of WSD algorithms and for better quality of MT or better precision of IR systems, such similar senses should be treated by the system as the same sense. On the other hand, the majority of the senses are really different and should be distinguished wherever possible in MT systems or in indexing and search in IR.

The presented algorithm (and the corresponding software) can be, for example, used as part of the lexicographer's workbench showing to the lexicographer potentially

similar senses. Of course, the final decision of merging the sense, changing their definitions or ignoring the situation should be made by the human (the lexicographer).

In our previous work [9] we obtained not only the stemmed forms and POS tags for the words in the definitions but also distinguished their word senses. In this paper, we did not use this information. This is due to that the algorithm presented in [9] is rather unreliable. Probably in the future work comparison of dictionary definitions taking into account word senses is to be tried.

References

1. Gelbukh, A. and G. Sidorov (2002). Morphological Analysis of Inflective Languages Through Generation. *Procesamiento de Lenguaje Natural*, N 29, September 2002, Spain. pp. 105–112.
2. Jiang, J.J. and D.W. Conrad. From object comparison to semantic similarity. In: *Proc. of Pacling-99* (Pacific association for computational linguistics), August, 1999, Waterloo, Canada, pp.256-263.
3. Karov, Ya. and Edelman, Sh. (1998) Similarity-based word-sense disambiguation. *Computational linguistics*, Vol. 24, pp. 41-59.
4. Lesk, M. (1986) Automatic sense disambiguation using machine-readable dictionaries: how to tell a pine cone from an ice cream cone. *Proceedings of ACM SIGDOC Conference*. Toronto, Canada, pp. 24-26.
5. Manning, C. D. and Shutze, H. (1999) *Foundations of statistical natural language processing*. Cambridge, MA, The MIT press, 680 p.
6. McRoy, S. (1992) Using multiple knowledge sources for word sense disambiguation. *Computational Linguistics*, Vol. 18(1), pp. 1-30.
7. Pedersen, T. (2002) A baseline methodology for word sense disambiguation. In: A. Gelbukh (Ed.) *Computational linguistics and intelligent text processing*, Lecture Notes in Computer Science N 2276, Springer-Verlag, 2002, pp 126-135.
8. Rasmussen E. Clustering algorithms. In: Frakes, W. B. and Baeza-Yates, R. *Information Retrieval: Data Structures and Algorithms*. Prentice Hall, Upper Saddle River, NJ, 1992, pp. 419-442.
9. Sidorov G. and A. Gelbukh (2001). Word sense disambiguation in a Spanish explanatory dictionary. Proc. of *TALN-2001*, Tours, France, July 2–5, 2001, pp 398-402.
10. Wilks, Y. and Stevenson, M. (1999) Combining weak knowledge sources for sense disambiguation. Proceedings of *IJCAI-99*, 884-889.
11. *WordNet: an electronic lexical database*. (1998), C. Fellbaum (ed.), MIT, 423 p.
12. Yarowksy, D. (1992) Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. Proceeding of *Coling-92*, Nante, France, pp. 454-460.