A Reverse Dictionary Based on Semantic Analysis Using WordNet

Oscar Méndez, Hiram Calvo, and Marco A. Moreno-Armendáriz

Centro de Investigación en Computación - Instituto Politécnico Nacional
Av. Juan de Dios Bátiz, 07738, Distrito Federal, México
omendez_a12@sagitario.cic.ipn.mx

Abstract. In this research we present a new approach for reverse dictionary creation, one purely semantic. We focus on a semantic analysis of input phrases using semantic similarity measures to represent words as vectors in a semantic space previously created assisted by WordNet. Then, applying algebraic analysis we select a sample of candidate words which passes through a filtering process and a ranking phase. Finally, a predefined number of output target words are displayed. A test set of 50 input concepts was created in order to evaluate our system, comparing our experimental results against OneLook Reverse Dictionary to demonstrate that our system provides better results over current available implementations.

Keywords: reverse dictionary, semantic analysis, search by concept, vector space model.

1 Introduction

Over the years, people have used dictionaries for two well-defined purposes. Both of them are reflected on the dictionary's definition that is a collection of words listed alphabetically in a specific language, which contains their usage informations, definitions, etymologies, phonetics, pronunciations, and other linguistic features; or a collection of words in one language with their equivalents in another, also known as a lexicon. When these different ideas come together we understand why this resource hasn't lost importance and continue to be widely used around the world.

As part of the technological evolution the world has experienced during the last years, dictionaries are now available in electronic format. This resource has different advantages over the traditional printed dictionary, being the most important the easy access that it allows users and the very fast response time. Lexicographers constantly improve this resource, in order to assist language users, by increasing the number of words defined in the dictionary and adding lots more information associated with each one of them. Its performance is simple, just mapping words to their definitions, i.e. it does a lookup based on the correct spelling of the input word to find the definition.

F. Castro, A. Gelbukh, and M. González (Eds.): MICAI 2013, Part I, LNAI 8265, pp. 275–285, 2013.

This traditional approach is really helpful mostly for readers and language students, but isn't good enough taking into account the perspective of people who produce language. We all have experienced the problem of being unable to express a word that represents an idea in our mind although we are conscious of related terms, a partial description, even the definition. This may be due to a lack of knowledge in the word's meaning or a recall problem. People mainly affected by this problem are writers, speakers, students, scientists, advertising professionals, among others. For them, traditional dictionary searches are often unsuccessful because these kind of search demands an exact input, while a language producer tends to require a reverse search where the input are a group of words forming a formal definition or just a series of related terms, and the output is a target word.

The need for a different search access mode in a dictionary led to the creation of a reverse dictionary. Its basic objective is to retrieve a target word when a group of words which appear in its definition are entered. In other words, given a phrase describing a desired concept or idea, the reverse dictionary provides words whose definitions match the entered phrase. The chances of giving an exact definition of a concept is very difficult so synonym words or related words could also be considered during the search.

In this research we developed a new method to generate a reverse dictionary based on a large lexical English database known as WordNet and the implementation of different semantic similarity measures which help us in the generation of a semantic space.

2 State of the Art

Only three printed reverse dictionaries exist for English language. The reason is probably the complexity of its elaboration, especially the fact of choosing the proper form to distribute the information. The Bernstein's Reverse Dictionary [4] was the first of its kind, in this book, the definitions of 13,390 words were reduced to their most brief form and then ordered alphabetically.

With the availability of dictionaries in electronic format, the interest for a reverse lookup application has been growing during the last years. Unlike printed versions, several attempts have been made in the creation of the reverse lookup method seeking for the best performance.

In the reverse electronic dictionary presented in [7], synonyms were used to expand search capabilities. They create a dictionary database with words numerically encoded for quick and easy access; adding also synonym group numeric codes in order to extend the searching process. In every search the numeric codes of the input words are found and stored. Then, main entry words having the numeric codes of the input words within their definitions are located and displayed as output candidates.

The magnitude of this natural language application is appreciated when dictionaries for different languages are constructed like [5]. For this Japanese reverse dictionary three different databases were created, using traditional IR concepts. Each database stored all dictionary words (EDR, 1995) with their definitions as

vectors, reflecting the term frequencies in each definition, with standard similarity metrics values (tf-idf, tf, binary values) as its elements. The reverse lookup method is separated in two stages. First, they parse the input concept with a morphological analyzer and create its vector, and then compare to the definition vectors to obtain the closest matching concept in the dictionary. To calculate the similarity between vectors they used cosine measure.

A different reverse lookup method was created in [8]. Their algorithm for French language does a reverse search using two main mechanisms. The first one extracts sets of words, from their lexical database of French words, which delimit the search space. For example, in the definition 'a person who sells food' the algorithm extracts all the sets of persons. The second mechanism computes a semantic distance between each candidate word in the extracted sets and the input definition to rank the output words. This latter value is based on the distances in the semantic graph, generated by their database, between hypernyms and hyponyms of the words being analyzed.

Another proposal was based on the notion of association: every idea, concept or word is connected [14]. Given a concept (system input) and following the links (associations) between input members, a target word would be reached. They proposed a huge semantic network composed of nodes (words and concepts) and links (associations), with either being able to activate the other.

In [15] the reverse lookup method depends on an association matrix composed of target words and their access keys (definition elements, related concepts). Two different sources were selected as corpus for the databases: WordNet and Wikipedia. The one based on WordNet used as target words the words defined in the dictionary and as access keys their definitions. The corpus based on Wikipedia used the page's raw text as target words (after a filtering process) and the words co-occurrences within a given window of specific size as access keys. Finally for every input phrase, their members are identified and the reverse search results in a list of words whose vectors contain the same input terms.

The most recent reverse dictionary application we found is shown in [12]. To construct their database they created for every relevant term t in the dictionary its Reverse Mapping Set (RMS) which requires finding all words in whose definition relevant term t appears. For every input phrase a stemming process is required, then a comparison is made between the input and the RMS looking for the words whose definitions contain the input members; this generates a group of candidates that pass through a ranking phase based on similarity values computed using a similarity measure implemented on WordNet and a parser.

The systems presented above share different methodological features. All of them consider not only the terms extracted from the user input phrase, but also terms similar or related to them (synonyms, hyponyms, hyperyms) and also needed a previous dictionary processing in order to form their databases. The reverse search done by [7] [14] [15] and [12] at some point of its procedure does a comparison between the user input phrase to every definition in their databases looking for definitions containing the same words as the user input phrase, while [8] and [5] based their reverse search on the highest similarity values measuring

graph distances and cosine respectively. All of this demonstrates a tendency during reverse lookup algorithms creation until now.

Our proposal presents a new approach for reverse dictionary creation, one purely semantic. We focus on a semantic analysis of input phrases using semantic similarity measures to represent words as vectors in a semantic space previously created assisted by WordNet. Then, applying algebraic analysis we select a sample of candidate words which passes through a filtering process and a ranking phase. Finally, a predefined number of output target words are displayed. It's important to mention that this project considers only nouns as word members of the semantic space, this part of speech restriction is due to the form in which vectors are constructed and the fact that it's only possible to calculate semantic similarity or semantic relatedness with words that belong to the same part of speech. Besides, it is well known that in natural language, concepts are expressed mostly as noun phrases [13].

3 WordNet as a Resource for Semantic Analysis

WordNet is a large lexical database for English and other languages. It groups words into sets of synonyms called synsets and describes relations between them. Lexical relations hold between word forms and semantic relations hold between word meanings.

The structure of word strings in WordNet specifies a specific sense of a specific word as shown below; this is used to avoid word sense disambiguation problems:

word#pos#sense

where pos is the part of speech of the word and its sense is represented by an integer number.

WordNet has a hierarchical semantic organization of its words, also called by computer scientists as "inheritance system" because of the inherited information that specific items (hyponyms) get from their superordinates. There are two forms to construe the hierarchical principle. The first one considers all nouns are contained in a single hierarchy. The second one proposes the partition of the nouns with a set of semantic primes representing the most generic concepts and unique beginners of different hierarchies [11]. To create WordNet's semantic space this project makes use of the second form and 25 top concepts were defined as semantic primes to represent the dimensions of word vectors.

The top concepts, with its specific sense, that were chosen are:

activity#n#1, animal#n#1 artifact#n#1, attribute#n#2, body#n#1, cognition#n#1, communication#n#2, event#n#1, feeling#n#1, food#n#1, group#n#1, location#n#1, motive#n#1, natural_object#n#1, natural_phenomenon#n#1, human_ being#n#1, plant#n#2, possession#n#2, process#n#6, quantity#n#1, relation#n#1, shape#n#2, state#n#1, substance#n#1, time#n#5

This is also the order given to the top concepts during the vector representation of words mentioned further on.

WordNet also includes the implementation of similarity and relatedness measures. A semantic relatedness measure uses all WordNet's relations for its calculation meanwhile a semantic similarity measure only uses the hyponymy relation. Three measures were considered for database construction: Jiang and Conrath (JCN) [9], Lin [10] and the Lesk algorithm (Lesk) [2]. The first two are similarity measures which have demonstrated to have a good performance among other measures that use WordNet as their knowledge source [6]; the last one is an adaptation of the original Lesk relatedness measure that take advantage of WordNet's resources [1].

Jiang and Conrath: this measure combines the edge-based notion with the information content approach. It calculates the conditional probability of encountering an instance of a child-synset given an instance of a parent synset, specifically their lowest super-ordinate (lso). The formula is expressed in 1.

$$dist_{JCN}(c_1, c_2) = 2\log(p(lso(c_1, c_2))) - (\log(p(c_1)) + \log(p(c_2))) \quad (1)$$

Lin: based on his similarity theorem: "The similarity between A and B is measured by the ratio between the amount of information needed to state the commonality of A and B and the information needed to fully describe what A and B are." It uses the same elements of JCN measure but in a different way. The formula is expressed in 2.

$$sim_{LIN}(c_1, c_2) = \frac{2\log p(lso(c_1, c_2))}{\log p(c_1) + \log p(c_2)} \quad (2)$$

Lesk: the original algorithm measures the relatedness between two words by the overlap between their corresponding definitions as provided by a dictionary. Basically the steps are:

1. Retrieve from an electronic dictionary all sense definitions of the words to be measured.
2. Determine the definition overlap for all possible sense combinations.
3. Choose senses that lead to highest overlap.

In WordNet an extended gloss overlap measure is available, which combines the advantages of gloss overlaps with the structure of a concept hierarchy to create an extended view of relatedness between synsets [1].

4 Semantic Space Construction

In this section we describe the construction process of the semantic space that contains the numeric representation of all WordNet's nouns as vectors of 25 dimensions determined by the top concepts mentioned before. For every noun we create its vector measuring semantic similarity between the word and each top concept, then it is stored in the semantic space. After reading and creating the vectors for every noun, the process ends. This procedure is detailed in Figure 1.

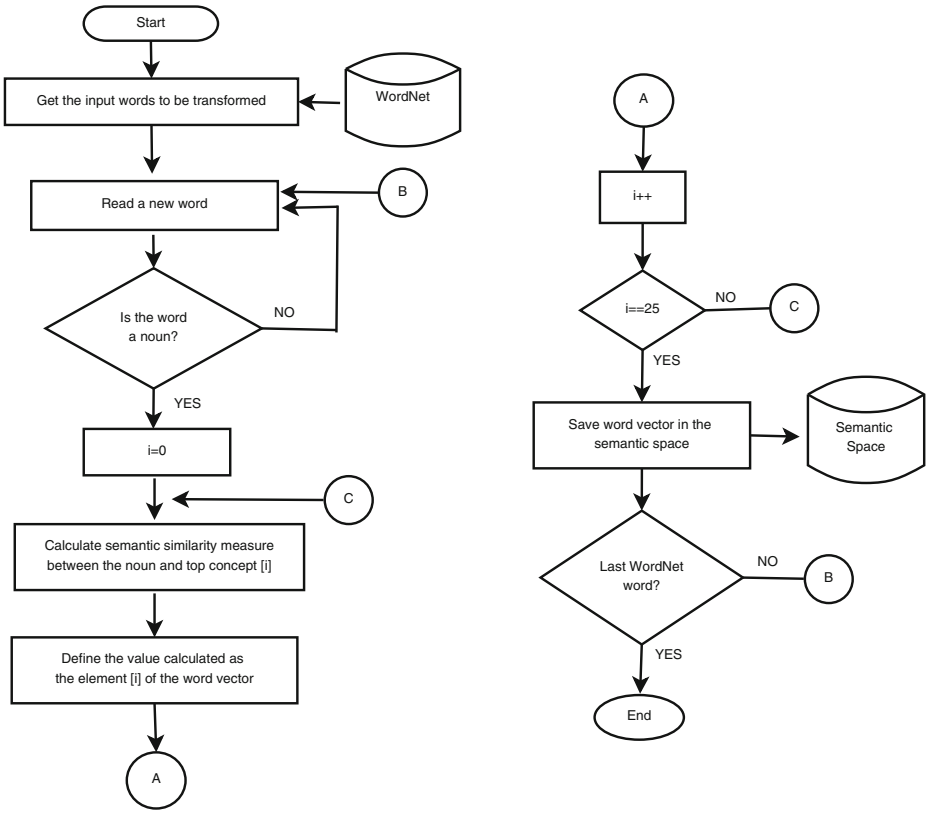

Fig. 1. Vector creation algorithm

The process was repeated for each of the different measures mentioned above, resulting on a semantic space with JCN measured vectors, another with Lin measured vectors, and the last one with Lesk measured vectors.

With the databases created, a normalization procedure was performed. For all word vectors maximum values of each dimensions were obtained in order to subsequently divide all word vectors dimensions by each respective maximum value previously obtained. Finally we have word vectors inside the semantic space with this form:

Genius → 0.05748, 0.04058, 0.09603, 0.06138, 0.06117, 0.04774, 0.07306, 0.02822, 0.06301, 0.07750, 0.05024, 0.05693, 0.03530, 0.12316, 0.01008, 0.01046, 0.00898, 0.05117, 0.03144, 0.05603, 0.04203, 0.07932, 0.03364, 0.02163, 0.07081

5 Search-by-Concept Process

A reverse dictionary receives a definition as input and gets a word that represents that concept as output. The search-by-concept dictionary proposed in this project is based on this principle.

The system input consists of a concept formed of n nouns. Once the input is defined, the system looks for the word vectors of their n components in the database and calculates their average. This gives as a result a new vector that should be located in the semantic space representing the word that combines all the characteristics given on the input concept. Regardless of whether the new vector already exists in the semantic space representing a word, a sample of twelve neighbor vectors is taken. This sample selection considers two parameters:

1. The euclidean distance value between vectors need to be:

   (a) For JCN less than 0.1
   (b) For Lin less than 0.8
   (c) For LSK less than 0.1

   These threshold values were determined after numerous testing. For vectors with euclidean distances bigger than the values mentioned above, the words they represented tend to have no relationship with the input concepts.
2. The product of the semantic similarity measure between each member of the input and the word represented by the neighbor vector is calculated; the top n words with the highest values are chosen to form the system output.

6 Results

Before showing some results, a complete example for JCN semantic space is shown below:

```
Input concept - gym_shoe#n#1 athletic_contest#n#1 race#n#2

gym_shoe#n#1 ->
0.05383, 0.03492, 0.11093, 0.05720, 0.05738, 0.04458, 0.06833, 0.02628,
0.05933, 0.07286, 0.04694, 0.05249, 0.03347, 0.11451, 0.00944, 0.00927,
0.00782, 0.04819, 0.02938, 0.05237, 0.03932, 0.07490, 0.03153, 0.02020,
0.06700

athletic_contest#n#1 ->
0.08950, 0.03136, 0.07729, 0.07229, 0.05214, 0.06832, 0.08528, 0.04630,
0.07227, 0.07297, 0.05879, 0.04805, 0.04601, 0.10280, 0.00946, 0.00849,
0.00708, 0.05869, 0.02943, 0.06550, 0.04902, 0.09036, 0.02934, 0.02281,
0.08023

race#n#2 ->
0.09333, 0.03214, 0.07960, 0.07480, 0.05331, 0.07113, 0.08805, 0.04859,
0.07433, 0.07472, 0.06073, 0.04942, 0.04732, 0.10539, 0.00970, 0.00866,
0.00724, 0.06035, 0.03020, 0.06766, 0.05061, 0.09279, 0.03003, 0.02350,
0.08229
```

```
Average vector ->
0.07888, 0.03280, 0.08927, 0.06809, 0.05427, 0.06134, 0.08055, 0.04039,
0.06864, 0.07351, 0.05548, 0.04998, 0.04226, 0.10756, 0.00953, 0.00880,
0.00738, 0.05574, 0.02967, 0.06184, 0.04631, 0.08601, 0.03030, 0.02217,
0.07650
```

After the search-by-concept process these are the results:

The seven output words with highest ranking are shown in Table 1. The most relevant result is `meet\#n\#1`. The proximity of its vector's dimensions values with the ones of the average vector previously calculated is notable.

```
meet#n#1 ->
0.08617, 0.03065, 0.07523, 0.07008, 0.05108, 0.06587, 0.08282, 0.04433,
0.07043, 0.07140, 0.05706, 0.04682, 0.04483, 0.10047, 0.00925, 0.00833,
0.00693, 0.05719, 0.02873, 0.06359, 0.04762, 0.08818, 0.02873, 0.02220,
0.07838
```

Table 1. System output for concept: gym_shoe#n#1 athletic_contest#n#1 race#n#2

Product of semantic similarity values	Euclidean distance	Word	Gloss
0.02642	0.02015	meet#n#1	a meeting at which a number of athletic contests are held
0.00580	0.02755	Olympic_Games#n#1	the modern revival of the ancient games held once every 4 years in a selected country
0.00426	0.02755	horse_race#n#1	a contest of speed between horses
0.00426	0.02755	footrace#n#1	a race run on foot
0.00387	0.05936	game#n#2	a single play of a sport or other contest
0.00325	0.03846	track_meet#n#1	a track and field competition between two or more teams
0.00293	0.04428	race#n#1	any competition

This process is done with the three different semantic spaces for every input concept. Table 2 and Table 3 show the reverse search of three different concepts with the two highest ranked output words from our system and the two highest ranked output words from an existing reverse dictionary [3] (OneLook Reverse Dictionary Online) respectively for comparison terms.

Table 2. Reverse search for three different concepts - System output

Concept	System results		
nature evolution life	JCN	growth#n#2	A progression from simpler to more complex forms.
		chemical_reaction#n#1	(Chemistry) a process in which one or more substances are changed into others.
	Lesk	oxidative_phosphorylation#n#1	An enzymatic process in cell metabolism that synthesizes ATP from ADP.
		blooming#n#1	The organic process of bearing flowers.
	Lin	growth#n#2	A progression from simpler to more complex forms.
		heat_sink#n#1	A metal conductor specially designed to conduct (and radiate) heat.
antenna screen broadcast	JCN	serial#n#1	A serialized set of programs.
		wide_screen#n#1	A projection screen that is much wider than it is high.
	Lesk	rerun#n#1	A program that is broadcast again.
		receiver#n#1	Set that receives radio or tv signals.
	Lin	electrical_device#n#1	A device that produces or is powered by electricity.
		surface#n#1	The outer boundary of an artifact or a material layer constituting or resembling such a boundary.
thunderbolt cloud water	JCN	atmospheric_electricity#n#1	Electrical discharges in the atmosphere.
		precipitation#n#3	The falling to earth of any form of water.
	Lesk	atmospheric_electricity#n#1	Electrical discharges in the atmosphere.
		cumulus#n#1	A globular cloud.
	Lin	atmospheric_electricity#n#1	Electrical discharges in the atmosphere.
		atmospheric_phenomenon#n#1	A physical phenomenon associated with the atmosphere.

Table 3. Reverse search for three different concepts - OneLook Reverse Dictionary output

Concept	OneLook Reverse Dictionary results
nature evolution life	natural Huxley
antenna screen broadcast	set-top box tv-antenna
thunderbolt cloud water	thunder cloud

Table 4. Evaluation

Output source	Aspect 1	Aspect 2	
Our system	94%	JCN	42%
		Lin	6%
		Lesk	32%
OneLook Reverse Dictionary	74%	20%	

At first sight, the results of our system seem to be correct answers for each concept, but in which way could we measure the quality of our results? We create a test set with 50 different concepts and for each concept we show the two highest ranked output words from our system and the two highest ranked output words from OneLook Reverse Dictionary, as in Table 2 and Table 3. A group of 10 people evaluated the test set under the following considerations:

1. Indicate if the output words converges with their associative reasoning.
2. Indicate which one of the sources gave the best results. And in case our system output was selected, specify the source of semantic space.

We resume the evaluation information in Table 4. Analyzing its content, it is clear that the performance of our system is better than OneLook Reverse Dictionary. Not only in the proximity with human associative reasoning capacity, it also gave the best results during the reverse search; where the concepts obtained from JCN semantic space demonstrate to combine better the characteristics of meaning of the input phrases.

7 Conclusions

In this paper, we described a new method for reverse dictionary construction with a semantic approach. We proposed the creation of three different semantic spaces, each one containing vectors created from different sources of semantic similarity measures. Also we described the different parts that constitute our reverse search together with an example. Our experimental results show that our system provides better results over current available implementations, including an improved system output providing also the gloss of every output word. This is very helpful in terms of evaluation because the user doesn′t have to waste time looking for a definition in order to verify the quality of the output.

As future work we propose the creation of two new semantic spaces based on different resources, a distributional thesaurus and latent Dirichlet allocation(LDA). A distributional thesaurus is a thesaurus generated automatically from a corpus by finding words which occur in similar contexts to each other. Meanwhile LDA is a generative probabilistic model for collections of discrete data. This enables an analysis of reverse search from different approaches to determine which one is the closest to human associative reasoning. A supervised approach (WordNet), semi-supervised approach (distributional thesaurus) and unsupervised approach (LDA).

Acknowledgements. The authors wish to thank to Instituto Politécnico Nacional (SIP-IPN grants 20130018 and 20130086, COFAA-IPN and PIFI-IPN) and the government of Mexico (SNI and CONACYT) for providing the necessary support to carry out this research work.

References

1. Banerjee, S., Pedersen, T.: An adapted lesk algorithm for word sense disambiguation using wordnet. In: Gelbukh, A. (ed.) CICLing 2002. LNCS, vol. 2276, pp. 136–145. Springer, Heidelberg (2002)
2. Banerjee, S., Pedersen, T.: Extended gloss overlaps as a measure of semantic relatedness. IJCAI 3, 805–810 (2003)
3. Beeferman, D.: Onelook Reverse Dictionary (2013), `http://www.onelook.com/reverse-dictionary.shtml` (accessed January-2013)
4. Bernstein, T., Wagner, J.: Bernstein's reverse dictionary. Quadrangle/New York Times Book Co. (1975)
5. Bilac, S., Watanabe, W., Hashimoto, T., Tokunaga, T., Tanaka, H.: Dictionary search based on the target word description. In: Proc. of the Tenth Annual Meeting of The Association for NLP (NLP 2004), pp. 556–559 (2004)
6. Budanitsky, A., Hirst, G.: Evaluating wordnet-based measures of lexical semantic relatedness. Computational Linguistics 32(1), 13–47 (2006)
7. Crawford, V., Hollow, T., Crawford, J.: Reverse electronic dictionary using synonyms to expand search capabilities. Patent, 07 1997; US 5649221 (1997)
8. Dutoit, D., Nugues, P.: A lexical database and an algorithm to find words from definitions (2002)
9. Jiang, J.J., Conrath, D.W.: Semantic similarity based on corpus statistics and lexical taxonomy. arXiv preprint cmp-lg/9709008 (1997)
10. Lin, D.: An information-theoretic definition of similarity. In: ICML, vol. 98, pp. 296–304 (1998)
11. Miller, G.A.: Nouns in wordnet: a lexical inheritance system. International Journal of Lexicography 3(4), 245–264 (1990)
12. Shaw, R., Datta, A., VanderMeer, D., Dutta, K.: Building a scalable database-driven reverse dictionary. IEEE Transactions on Knowledge and Data Engineering 25(3), 528–540 (2013)
13. Sowa, J.F.: Conceptual structures: Information processing in mind and machine (1984)
14. Zock, M., Bilac, S.: Word lookup on the basis of associations: from an idea to a roadmap. In: Proceedings of the Workshop on Enhancing and Using Electronic Dictionaries, ElectricDict 2004, pp. 29–35. Association for Computational Linguistics, Stroudsburg (2004)
15. Zock, M., Schwab, D.: Lexical access based on underspecified input. In: Proceedings of the workshop on Cognitive Aspects of the Lexicon, pp. 9–17. Association for Computational Linguistics (2008)