Designing an Electronic Reverse Dictionary Based on Two Word Association Norms of English Language

Jorge Reyes-Magaña[1,2], Gemma Bel-Enguix[1], Gerardo Sierra[1], Helena Gómez-Adorno[3]

[1] Instituto de Ingeniería, Universidad Nacional Autónoma de México, Circuito Escolar s/n, Ciudad Universitaria, Delegación Coyoacán, Ciudad de México, México
[2] Facultad de Matemáticas, Universidad Autónoma de Yucatán, Anillo Periférico Norte, Tablaje Cat. 13615, Colonia Chuburná Hidalgo Inn, Mérida, Yucatán, México
[3] Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México, Circuito Escolar s/n, Ciudad Universitaria, Delegación Coyoacán, Ciudad de México, México
E-mail: jorge.reyes@correo.uady.mx, gbele@iingen.unam.mx, gsierram@iingen.unam.mx, helena.gomez@iimas.unam.mx

Abstract

This work introduces the exploitation of some language resources, namely word association norms, for building lexical search engines. We used the Edinburgh Associative Thesaurus and the University of South Florida Free Association Norms for the construction of knowledge graphs that will let us execute algorithms over the nodes and edges in order to do a lexical search. The aim of the search is to perform an inverse dictionary search that, given the description of a concept as a query in natural language, will retrieve a target word. We evaluated two graph approaches, namely Betweenness Centrality and PageRank, using a corpus of human-definitions. The results are compared with the BM25 text-retrieval algorithm and also with an online reverse dictionary– OneLook Reverse Dictionary. The experiments show that our lexical search method is competitive with the IR models in our case study, even with a slight outperformance. This demonstrates that an inverse dictionary is possible to build with these kind of resources, no matter the language of the Word Association Norm.

Keywords: inverse dictionary; norm association words; graph theory

1. Introduction

Two types of dictionaries can be distinguished in order to link a concept with its meaning: semasiological and onomasiological. The former provide meanings, i.e. given a word, the user obtains the meaning of the word. The latter work in the opposite way, given the description of a word, the user obtains the related concept (Baldinger, 1970). The problem of building an onomasiological dictionary has been tackled in diverse ways, since in printed onomasiological dictionaries the words are not isolated, but usually arranged by shared semantic or associated features grouped under headwords (Sierra, 2000b; Sierra & McNaught, 2003). The main disadvantage in this type of search is that a really specific idea of the concept is needed in order to search in the right place of the index or headwords. Currently, an onomasiological dictionary can be thought of as a simple internet search, thanks to the information accessible through different digital resources for almost any kind of topic. Unfortunately, the outcome of the search tends to be even more confusing, or it simply shows other results that do not correspond to the concept.

The present paper presents two algorithms that perform a lexical search over a knowledge graph in a similar way onomasiological dictionaries help to find a concept, starting from a definition or a set of clue words. We developed a model based on graph-based techniques, the *Betweenness centrality* and *PageRank*, to perform the search of a given concept on a dataset of word association norms for English, the Edinburgh Associative Thesaurus (EAT) (Kiss et al., 1973b), and the University of South Florida Free Association Norms (Nelson et al., 2004).

We used an evaluation corpus consisting of seven concepts. Although this is a small evaluation corpus, it can be considered as an illustrative example on how our method allows the building of reverse dictionaries using WAN. For each concept, 10 definitions were provided by human native speakers. In most cases, the definitions are very different from the ones found in dictionaries; they lack specialized terms and include cultural references and connotations. This allows us to design a more realistic electronic application, that will help people find a target word even with a limited knowledge of specific details. We used the 70 definitions as queries in our search model and compared the results with an information retrieval (IR) model (BM-25) and the online reverse dictionary *OneLook*[1]. Our model achieved better results than the baseline IR model for this case of search scenario.

2. Related Work

2.1 Onomasiological searching

There are some specialized texts that aim to help writers who need to go from a meaning or concept to a corresponding word. These resources are gathered according to their behaviour in the following three features: a) the type of information they contain, b) the structure of the wordbook, and c) the type of search undertaken. We distinguish four different groups: *Thesauri*, *Reverse dictionaries*, *Synonymy and antonymy dictionaries* and *Pictorial dictionaries*.

The whole scenario of onomasiological searches changed with the universalization of the internet and language technologies, that allowed building online resources powered by the huge corpus the world wide web provides. In the last two decades, several online dictionaries have been designed that allow natural language searches. The users enter their own definition in natural language and the engine looks for the words that match the definition.

[1] https://www.onelook.com/thesaurus/

One of the first online dictionaries allowing this type of search was the one created for French by Dutoit and Nugues (2002). Another interesting contribution was introduced by Bilac et al. (2004), who designed a dictionary for Japanese. El-Kahlout and Oflazer (2004) built a similar resource for Turkish. For English, there exists an online onomasisological dictionary, OneLook Reverse Dictionary,[2] that retrieves acceptable results. One of the main works in Spanish is the one by Sierra (2000a), which was improved by Hernández (2012).

2.2 Free word associations

Free word associations (WA) are commonly collected by presenting a stimulus word (SW) to the participant and asking him to produce in a verbal or written form the first word that comes to his mind. The answer generated by the participant is called a response word (RW).

Compilations of WA are called Word Association Norms. Many languages have this type of resources, which are time-consuming to collect and need many volunteers.

In recent years, the web has become a natural way to get data to build such resources.

Jeux de Mots[3] provides an example in French (Lafourcade, 2007), whereas the *Small World of Words*[4] contained datasets in 14 languages at the time of writing. Nevertheless, the norms are only available in German. The authors (De Deyne et al., 2013) will make the other languages available as soon as they finish collecting the material. Such repositories have the problem of being collected without control over who is actually adding to the content, the linguistic proficiency of the users, and their age, gender or level of studies.

For Spanish, there exist several datasets of word associations. Algarabel et al. (1998) integrate 16,000 words, including statistical analyses of the results. Macizo et al. (2000) build norms for 58 words based on the responses of children, and Fernández et al. (2004) derived the free-association norms for the Spanish names of Snodgrass and Vanderwart pictures (Sanfeliu & Fernández, 1996).

The use of free word associations to compute relationality between words is not new. Borge-Holthoefer and Arenas (2009) describe a model (RIM) to extract semantic similarity relations from free association information. In recent years, Bel-Enguix et al. (2014) used techniques of graph analysis to calculate associations from large collections of texts. Additionally, Garimella et al. (2017) published a model of word associations

---

[2] https://www.onelook.com/reverse-dictionary.shtml.

[3] http://www.jeuxdemots.org/.

[4] https://smallworldofwords.org/.

that was sensitive to the demographic context.

The only resource designed and compiled for Mexican Spanish is the *Corpus de Normas de Asociación de Palabras para el Español de México*[5] (Arias-Trejo et al., 2015).

Among the available compilations, the best-known in English are the *Edinburgh Associative Thesaurus*[6] (EAT) (Kiss et al., 1973a) and the collection of the University of South Florida (USF) (Nelson et al., 1998)[7]. This work proposes the use of these datasets to be the basis of the design of a lexical search system that works from the clues or definitions to the concept, i.e., from the responses to the stimuli in order to build the reverse dictionary.

3. Word Association Norms datasets and graph

The EAT was mainly collected with undergraduate students from different British universities. The participants were between 17 and 22 years old, among which 64% were males and 36% were females. Every informant gave responses for 100 words, and every word was given to 100 participants. The resource was elaborated between 1968 and 1971 and published in 1973.

We used an XML version of the resource[8], prepared by the University of Montreal, that consists of 8,211 stimulus words, and 20,445 different words including both, stimuli and responses.

The USF norms were collected with more than 6,000 participants that produced nearly three-quarters of a million responses to 5,019 stimulus words. Participants were asked to write the first word that came to mind that was meaningfully related or strongly associated with the presented word on the blank shown next to each item. The norms are distributed as plain text files separated by commas [9] so that the document can be opened in a variety of different programs and databases. In this format, data for 5,019 normed words and their 72,176 responses can be found.

The graph representing the WAN's datasets has been elaborated with lemmatized lexical items. It is formally defined as: $G = \{V,E,\varphi\}$ where:

- $V = \{v_i | i = 1,\ldots,n\}$ is a finite set of nodes of length n, $V \neq \emptyset$, that corresponds to the *stimuli* and their *associates*.

[5] http://www.labpsicolinguistica.psicol.unam.mx/Base/php/general.php

[6] http://www.eat.rl.ac.uk/

[7] http://web.usf.edu/FreeAssociation

[8] http://rali.iro.umontreal.ca/rali/?q=en/Textual%20Resources/EAT

[9] http://w3.usf.edu/FreeAssociation/AppendixA/index.html

- $E = \{(v_i,v_j)/v_i,v_j \in V, 1 \leq i,j \leq n\}$, is the set of edges.
- $\varphi : E \rightarrow \mathrm{R}$, is a function over the weight of the edges.

We built separate graphs, each one is undirected so that every *stimulus* is connected to every associated word without any precedence order.

For the weight of the edges we used one of the following functions:

Frequency (F) Counts the number of occurrences of every associate to its *stimulus* in the whole dataset. For the system to work in the shortest paths, we need to calculate the *IF*, inverse frequency, that is defined in the following way: being F the frequency of a given associated word, and ΣF the sum of the frequencies of the words connected to the same *stimulus*, $IF = \Sigma F - F$

Association Strength (AS) Establishes a relation between the frequency (F) and the number of associations for every stimulus. It is calculated as follows: being F the frequency of a given associated word, and ΣF the sum of the frequencies of the words connected to the same *stimulus* (the total number of responses), the association strength (AS) of the word W to such *stimulus* is given by the formula:

$$AS_W = \frac{F}{\Sigma F}$$

For our experiments, we need to calculate the inverse association strength, *IAS*, in order to prepare the system to work with graph-based algorithms:

$$IAS_W = 1 - \frac{F}{\Sigma F}$$

Figure 1 depicts a subgraph of the EAT dataset, containing only four stimuli with their corresponding associates. It can be observed that there are some associate words that are common to different stimuli, even for this small subgraph. We can also find relationships between two stimuli; for example, *hamburger* and *lion.* Figure 2 depicts a subgraph of the USF dataset, containing the same four stimuli presented in Figure 1, but in this case the corresponding responses were the available in the American resource. We can observe that the associate word *food* is shared by *spaghetti* and *hamburger.*

4. Graph algorithms and the reverse dictionary

Given a definition, we search in the graph for the word that better matches it. For this purpose we considered centrality measures, because these type of algorithms identify the most important nodes in a graph; for example, the degree centrality assumes that important nodes have many connections. The degree centrality is not suitable for our purposes because we need to find the most important nodes for a specific subset of nodes (the nodes that represent the words in a definition). In order to build the inverse dictionary we choose two algorithms, the *Betweenness Centrality* and *PageRank*, described in the following sections.

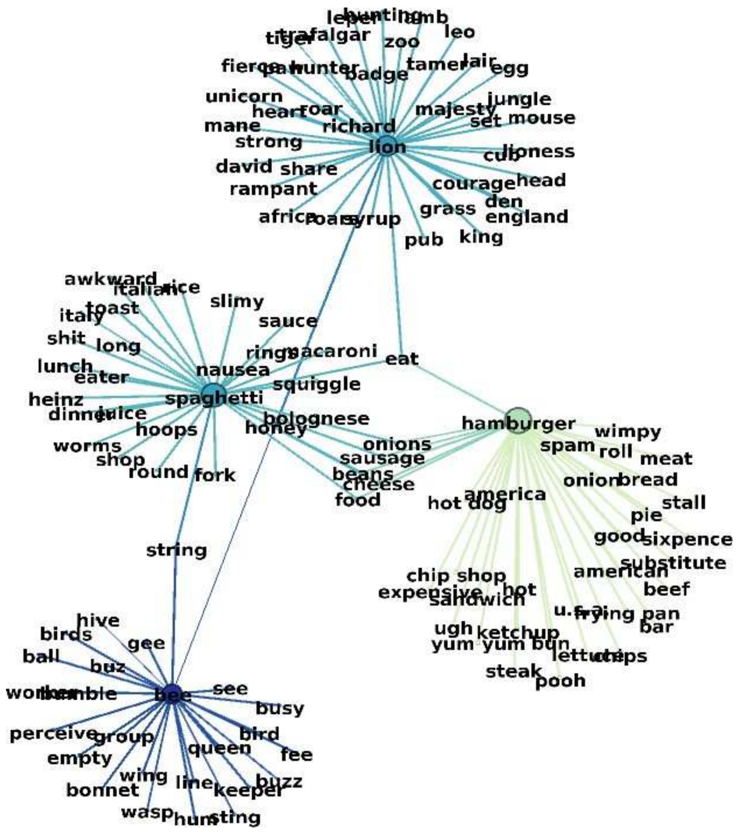

Figure 1: Subgraph based on EAT with the stimuli *bee, lion, hamburger,* and *spaghetti* with their corresponding associates.

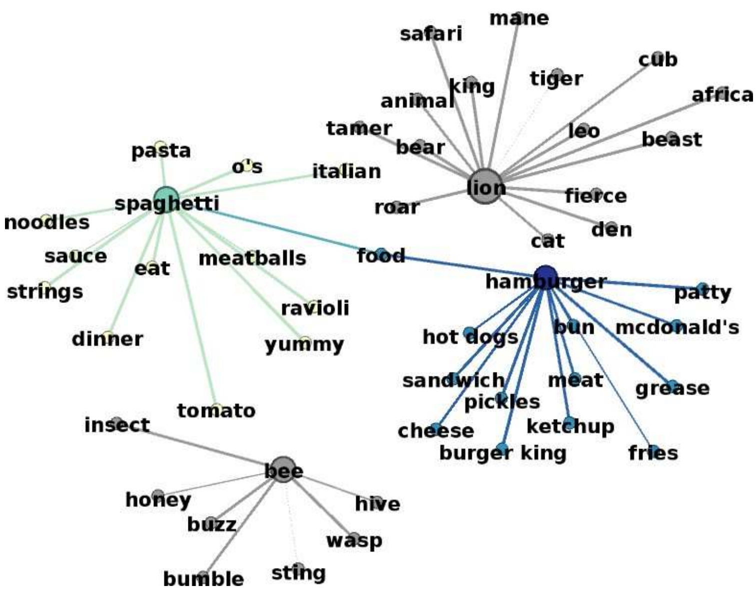

Figure 2: Subgraph based on Florida Free Association Norms with the stimuli *bee, lion, hamburger* and *spaghetti* with their corresponding associates.

4.1 Betweenness Centrality

We choose a variation of the *Betweenness Centrality* (BT) algorithm (Freeman, 1977) which instead of computing the BT of all pairs of nodes in a graph, calculates the centrality based on a sample (subset) of nodes (Brandes, 2008). The traditional betweenness algorithm assumes that important nodes connect other nodes. For a given node (v) in a graph (G), the BT is calculated as the relation between the number of shortest paths between nodes i and j that pass through v and the number of shortest paths between i and j. It is formally described as follows:

$$C_{btw}(v) = \sum_{i,j \in V} \frac{\sigma_{i,j}(v)}{\sigma_{i,j}} \tag{1}$$

where:

$V =$ is the set of nodes, $\sigma_{i,j}$ is the number of shortest paths between i and j, and $\sigma_{i,j}(v)$ is the number of those paths that pass through some node v that is not i or j.

In a non-weighted graph, the algorithm looks for the shortest path. In a weighted graph, like the one we have built, it finds the path that minimizes the sum of the weights of the edges.

The BT algorithm was introduced based on the general idea that when a particular person in a group is strategically located on the shortest communication path connecting pairs of others, that person is in a central position (Bavelas, 2002). Noting the importance of the shortest paths, we adapted the information available in WAN, letting the most important nodes and their relations be represented as minimal values, as explained before. This is why we have adopted the weighting function based on the inverse frequency and the inverse association strength.

We employ the approximation of the BT algorithm in order to search for the concept related to a given definition. This is because it only uses a subset of nodes to find the most central ones in the graph. Our hypothesis is that, if we use a subset, the nodes of the WAN graph (WG) that represent the words of a definition as initial and final nodes in the BT algorithm, and calculate the centrality of the other nodes in WG taking these nodes as pairs, then the more central nodes will be the concept of such a definition. This approximation is formally described as follows:

$$C_{btw\_aprox}(v) = \sum_{i \in I, f \in F} \frac{\sigma_{i,f}(v)}{\sigma_{i,f}} \tag{2}$$

where: I is the set of initial nodes, F is the set of final nodes, $\sigma_{i,f}$ is the number of shortest paths between i and f, and $\sigma_{i,f}(v)$ is the number of those paths that pass through some node v that is not i or f.

Therefore, we define a subgraph composed by the words (nodes) of the definition. This subgraph is used as both initial and final nodes, for calculating the shortest paths from each of the nodes of the initial nodes set to each one of the nodes of the final nodes set. Finally, the nodes are ranked taking the measure of BT as a parameter for the comparison of the most important nodes found by the algorithm.

4.2 PageRank

PageRank computes a ranking of the nodes in a graph G based on the structure of the incoming links. It was originally designed as an algorithm to rank web pages. It was developed by Page et al. (1999), it is formally described as:

Let u be a web page. Then let F be the set of pages u points to and B be the set of pages that point to u . Let $N_u = |F_u|$ be the number of links from u and let c be a factor used for normalization (so that the total rank of all web pages is constant).

R represents the computation of PageRank, as follows:

$$R(u) = c \sum_{u \in B_u} \frac{R(v)}{N_v} \tag{3}$$

The rank of a page is divided among its forward links evenly to contribute to the ranks of the pages they point to. The equation is recursive but it may be computed by starting with any set of ranks and iterating the computation until it converges. In the most general and intuitive manner, PageRank corresponds to the standing probability distribution of a random walk on the graph of the Web.

Figure 3 shows Mathematical PageRanks for a simple network, expressed as percentages. Each value in the nodes represents the probability of a random walker finishing the path in it. The highest value is seen in node B, as it is the one with the most connections in the graph.

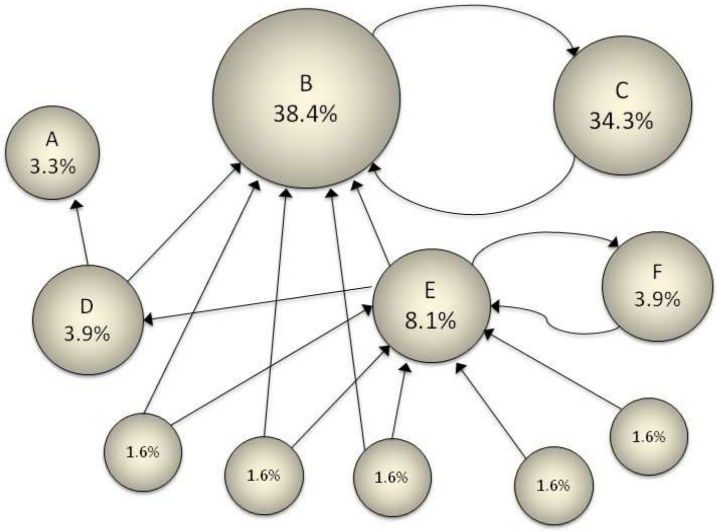

Figure 3: PageRank percentage in a simple network.

In our case, the pages described above are the words in the WAN datasets, the webpage links correspond to all the relations given by the stimuli-response between words. The hypothesis here is that the higher scores returned by the PageRank algorithm correspond to a target word being matched with a suitable definition. In this case, we didn't need the original graph to be tested with the algorithm because it will return the most relevant node of all the WAN dataset, instead, we pruned the graph considering some aspects described in the following subsection.

4.3 Algorithm description

Algorithm 1: Reverse dictionary

```
Data: WAN datasets, definitions to search
Result: list of ranked concepts
pre-process(WAN datasets);
pre-process(definitions to search);
GraphWAN = build-graph(WAN datasets);
GraphWAN = prune-graph(GraphWAN);
for each definition do
    definition = remove-StopWords(definition);
    definition = filter-WordsInWAN(definition);
    build_subgraph(definition);
    ranking_nodes_BT = BT(GraphWAN,subgraph);
    ranking_nodes_PR = PR(GraphWAN);
    ascending_order(ranking_nodes_BT);
    ascending_order(ranking_nodes_PR);
```

Algorithm 1 presents the overall schema of our model. The WAN datasets used here as input refer to both EAT or USF norms. First, we perform some pre-processing steps. All the *stimuli* and the *responses* are lemmatized, leaving each word as the most representative of the flexed forms. The same pre-processing is applied to the definitions to be searched by the model. This process provides us with more matches in the case when the definition contains *table*, *tables*, etc. because it will be transformed into its lemma, *table*. For this purpose, we use the lemmatization process available in *spacy*[10].

Later, we built GraphWAN with the Python package Networkx (Hagberg et al., 2005). Due to various experiments carried out with the original graph we discovered that compression was needed in order to get a more compact graph to be processed, and for this purpose we prune the original graph taking all the neighbours for each word of the definition to be searched, i.e. all nodes that have a connection with the words of the definition were selected considering the original graph structure.

Then, for each definition to be searched we removed all the functional words using the stop words list available in the *NLTK* package (Bird & Loper, 2004). Next, with the list of words with lexical meaning, we kept only the ones that belong to the vocabulary in WAN. With this we built a subgraph to be the input in the Betweenness Centrality algorithm. Finally, the nodes were sorted out according to the highest centrality measure, which corresponds to the words that are closer to the ones of the definition.

5. Experiments and results

5.1 Evaluation corpus

For the experiments, a small corpus containing 10 definitions for seven concepts was used, and these definitions were taken from Sierra and McNaught (2000), originally used for evaluating their work. These definitions are reported to be gathered with a small group of twenty undergraduate students in the area of terminology. From two sets, each student was asked to take a set and write on a blank sheet of paper, similar to an onomasiological search, a concept, a definition or the ideas suggested to them by each word. After exchanging the sheets, the other students participating in the experiment wrote the word or words designating the concepts identified or written on the blank sheets by the previous student.

The selected words used for evaluating our system are: *water, squirrel, bench, hurricane, lemon, bucket* and *clothes*. Table 1 presents an example of 10 definitions of the same concept given by different students.

[10] https://spacy.io/

It's a little rodent and can be red or grey, it has a big bushy tail
A small rodent living in trees with a long bushy tail
A small rodent which lives in trees, collects nuts and has a bushy tail
Animal, grey/red, bushy tail, lives in trees, buries nuts
Small animal, lives in trees, eats acorns, has a bushy tail
Animal, bushy tail, eats nuts, builds nests in trees called dreys
Small funny animal with big, bushy tail, likes nuts, likes trees
Animal that lives in trees and collects acorns, has a long tail
A small-sized animal, habitat in trees
Small grey mammal, relative to the rodent, found in both countryside and town

Table 1: Definitions of *squirrel* given by the students.

5.2 Results with the inverse dictionary and graphs

The experiments were performed taking into account weighted graphs with the two previously mentioned functions: Inverse Frequency (IF) and Inverse Association Strength (IAS). Considering separated graphs with each of the WAN datasets.

For the evaluation of the inference process, we used the technique of precision at k ($p@k$) from Manning et al. (2009). For example, $p@1$ shows that the concept associated with a given definition was ranked correctly in the first place, in $p@3$ the concept was in the first three results, and the same applies to $p@5$ and $p@10$.

The results are shown in Tables 2 and 3. As a general statement when the model searches over the graphs weighted with IAS the results are higher than when searching on the graph weighted with IF in both datasets. Psychologists agree that Association Strength (AS) is the measure that implies a cognitive relationship between two terms, and this idea is reflected in our results. Frequency is closely related to AS, but it lacks the generalization of the latter function.

Regarding the WAN datasets, the best results are achieved using USF Word Association Norms processed with Betweenness Centrality. We consider this is because this algorithm lets us create a source and target of nodes that exactly correspond to the words given by a user in the definition, compared to PageRank that analyses the graph built with the neighbourhood around this words.

Weighting function	Graph Algorithm	p@1	p@3	p@5	p@10
Inverse Frequency (IF)	Betweenness Centrality (BT)	**0.152**	0.186	0.220	0.237
Inverse Association Strength (IAS)	Betweenness Centrality (BT)	**0.152**	**0.220**	**0.237**	**0.254**
Inverse Frequency (IF)	PageRank (PR)	0.000	0.074	0.129	0.129
Inverse Association Strength (IAS)	PageRank (PR)	0.000	0.0740	0.129	0.129

Table 2: Results in terms of precision of our model with EAT dataset

Weighting function	Graph Algorithm	p@1	p@3	p@5	p@10
Inverse Frequency (IF)	Betweenness Centrality (BT)	0.236	0.309	**0.418**	0.436
Inverse Association Strength (IAS)	Betweenness Centrality (BT)	**0.290**	**0.363**	**0.418**	**0.5272**
Inverse Frequency (IF)	PageRank (PR)	0.037	0.074	0.129	0.222
Inverse Association Strength (IAS)	PageRank (PR)	0.037	0.074	0.148	0.222

Table 3: Results in terms of precision of our model with USF dataset

5.3 Results

In order to evaluate the relevance of our method, we performed experiments with other well-known IR methods.

First, we compared the performance of our method with the results of a reverse dictionary. To do that, we used the OneLook Thesaurus that allows you to describe a concept and returns a list of words and phrases related to that concept. The definitions were manually checked using the OneLook web application[11].

Secondly, we performed experiments with one of the most successful text-retrieval algorithms, Okapi BM25, based on probabilistic models and developed in the seventies by Stephen E. Robertson and Karen Spärck Jones (1976). The algorithm implemented following Robertson and Zaragoza (2009) is based on the bag-of-words method. Given a query, it ranks a list of documents according to their relevance for such query. We have applied it considering as a document every definition and every set of responses to a stimulus.

[11] https://www.onelook.com/thesaurus/

Method	**P@1**	**P@3**	**P@5**	**P@10**
OneLook	0.202	0.347	0.376	0.434
Reverse Dictionary with USF (IAS)	**0.290**	0.363	0.418	**0.5272**
BM25 with EAT	0.257	0.357	0.414	0.471
BM25 with USF	0.257	**0.400**	**0.457**	0.514

Table 4: Comparative precision results

The results achieved using the two baselines, OneLook and BM25, are reported in Table 4, where they are compared with the best result obtained by the inverse dictionary with our model. The BM25 algorithm showed better performance than the OneLook reverse dictionary when the search was performed over the WAN datasets. The BM25 was implemented using both WAN datasets. For each *stimuli* we built a document containing all the *responses* established in the resource. The better results are consistent with the ones seen in the reverse dictionary, USF norms show the best performance with this *IR* algorithm. It is observed that this algorithm is the most competitive against our model, but we outperformed the results in p@1 and p@10, while we unperformed in p@3 and p@5.

The system is fast, efficient and demonstrates high performance. However, the structure of the resource we have built favours the fact that two words that are not really related by association could have a short path between them because they share a connected word. This is expected to be a problem of our reverse dictionary based on WANs, although it can be minimized by performing some kind of lexical filter in the future.

6. Conclusions and future work

This paper introduces a model for onomasiological searches that has some novelties; among them the simplicity, the use of graph-based techniques and the WAN datasets the method is based on. However, we observed that the graph built with all the nodes and edges contained in the datasets tends to be not so good, due to the number of paths that lead to the wrong results. In order to solve this problem, we had to make a graph reduction keeping the most relevant nodes and their paths.

We have shown how descriptions of concepts that are made by ordinary people with non-scientific specifications can retrieve accurate results using our method. This is possible thanks to the nature of the dataset. Indeed, word association norms group words that are closely related in a cognitive way, and taking advantage of the metrics in the original resource that can be used to produce weighted edges in the graph that is built.

The success of the system with non-scientific input can drive new lines of applied research, and the implementation of different assistant writing systems especially oriented to people with a range of aphasias, like dysnomia and Alzheimer's disease.

Our algorithm has shown competitive performance compared with other baseline systems.

7. Acknowledgments

This research has been supported by projects PAPIIT IA401219 from the Universidad Nacional Autónoma de México; and CONACYT Fronteras de la Ciencia 2016-01-2225.

8. References

Algarabel, S., Ruíz, J. C. & Sanmartín, J. (1998). *The University of Valencia's computerized Word pool*. Instruments & Computers.: Behavior Research Methods.

Arias-Trejo, N., Barrón-Martínez, J.B., Alderete, R.H.L. & Aguirre, F.A.R. (2015). *Corpus de normas de asociación de palabras para el español de México [NAP]*. Universidad Nacional Autónoma de México.: UNAM.

Baldinger, K. (1970). *Teoría semántica: hacia una semántica moderna*, volume 12. Alcalá.

Bavelas, A. (2002). A mathematical model for group structure. *Social networks: critical concepts in sociology, New York: Routledge*, 1, pp. 161–88.

Bel-Enguix, G., Rapp, R. & Zock, M. (2014.). A Graph-Based Approach for Computing Free Word Associations. In *Proceedings of the 9th edition of the Language Resources and Evaluation Conference*. pp. 221–230.

Bilac, S., Watanabe, W., Hashimoto, T., Tokunaga, T. & Tanaka, H. (2004). Dictionary search based on the target word description. In *Proceedings of the Tenth Anual Meeting of the Association for Natural Language Processing*. pp. 556–559.

Bird, S. & Loper, E. (2004). NLTK: the natural language toolkit. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions*. Association for Computational Linguistics, p. 31.

Borge-Holthoefer, J. & Arenas, A. (2009). Navigating Word Association norms to Extract Semantic Information. In *Proceedings of the 31st Annual Conference of the Cognitive Science Society.*

Brandes, U. (2008). On variants of shortest-path betweenness centrality and their generic computation. *Social Networks*, 30(2), pp. 136–145.

De Deyne, S., Navarro, D.J. & Storms, G. (2013). Better explanations of lexical and semantic cognition using networks derived from continued rather than single-word associations. *Behavior research methods*, 45(2), pp. 480–498.

Dutoit, M. & Nugues, P. (2002). A Lexical Database and an Algorithm to Find Words from Definitions. In *Proceedings of the 15th European Conference on Artificial Intelligence*. pp. 450–454.

El-Kahlout, I. & Oflazer, K. (2004). Use of Wordnet for Retrieving Words from Their Meanings. In *2nd Global WordNet Conference.*

Fernández, A., Díez, E., Alonso, M.A. & Beato, M.S. (2004). Free-association norms form the Spanish names of the Snodgrass and Vanderwart pictures. *Behavior*

Research Methods, Instruments & Computers, 36, pp. 577–583.

Freeman, L.C. (1977). A set of measures of centrality based on betweenness. *Sociometry*, pp. 35–41.

Garimella, A., Banea, C. & Mihalcea, R. (2017). Demographic-aware word associations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. pp. 2285–2295.

Hagberg, A., Schult, D. & Swart, P. (2005). Networkx: Python software for the analysis of networks. *Mathematical Modeling and Analysis, Los Alamos National Laboratory*.

Hernández, L. (2012). *Creación semi-automática de la base de datos y mejora del motor de búsqueda de un diccionario onomasiológico*. Universidad Nacional Autónoma de México.

Kiss, G., Armstrong, C., Milroy, R. & Piper, J. (1973a). *An associative thesaurus of English and its computer analysis*. Edinburgh.: Edinburgh University Press.

Kiss, G.R., Armstrong, C., Milroy, R. & Piper, J. (1973b). An associative thesaurus of English and its computer analysis. *The computer and literary studies*, pp. 153–165.

Lafourcade, M. (2007). Making people play for Lexical Acquisition. *In Proceedings of the th SNLP 2007, Pattaya, Thaïland*, 7, pp. 13–15.

Macizo, P., Gómez-Ariza, C.J. & Bajo, M.T. (2000). Associative norms of 58 Spanish for children from 8 to 13 years old. *Psicológica*, 21, pp. 287–300.

Manning, C., Raghavan, P. & Schutze, H. (2009). *Introduction to Information Retrieval*. Cambridge University Press.

Nelson, D.L., McEvoy, C.L. & Schreiber, T.A. (1998). *Word association rhyme and word fragment norms*. The University of South Florida.

Nelson, D.L., McEvoy, C.L. & Schreiber, T.A. (2004). The University of South Florida free association, rhyme, and word fragment norms. *Behavior Research Methods, Instruments, & Computers*, 36(3), pp. 402–407.

Page, L., Brin, S., Motwani, R. & Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Robertson, S. & Sparck Jones, K. (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27(3).

Robertson, S. & Zaragoza, H. (2009). The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval*, 3(4), pp. 333–389.

Sanfeliu, C. & Fernández, A. (1996). A set of 254 Snodgrass' Vanderwart pictures standardized for Spanish: Norms for name agreement, image agreement, familiarity, and visual complexity. *Behavior Research Methods, Instruments, & Computers*, 28, pp. 537–555.

Sierra, G. (2000a). Design of an onomasiological search sistem: A concept-oriented tool for terminology. *Terminology*, 6(1), pp. 1–34.

Sierra, G. (2000b). The onomasiological dictionary: a gap in lexicography. In *Proceedings of the Ninth Euralex International Congress*. pp. 223–235.

Sierra, G. & McNaught, J. (2000). Extracting semantic clusters from MRDs for an onomasiological search dictionary. *International Journal of Lexicography*, 13(4), pp. 264–286.

Sierra, G. & McNaught, J. (2003). Natural Language System for Terminological Information Retrieval. In A. Gelbukh (ed.) *Computational Linguistics and Intelligent Text Processing*. Berlin, Heidelberg: Springer, pp. 541–552.