Information Retrieval with a Simplified Conceptual Graph-Like Representation

Sonia Ordoñez-Salinas,[1] Alexander Gelbukh [2]

[1] Universidad Distrital F.J.C and Universidad Nacional, Colombia.
[2] Center for Computing Research (CIC), National Polytechnic Institute (IPN), Mexico.
sordonez @ udistrital.edu.co, gelbukh @ cic.ipn.mx

Abstract. We argue for that taking into account semantic relations between words in the text can improve information retrieval performance. We implemented the process of information retrieval with simplified Conceptual Graph-like structures and compare the results with those of the vector space model. Our semantic representation, combined with a small simplification of the vector space model, gives better results. In order to build Conceptual Graph-like representation, we have developed a grammar based on the dependency formalism and the standard defined for Conceptual Graphs (CG). We used noun pre-modifiers and noun post-modifiers, as well as verb frames, extracted from VerbNet, as a source of definition of semantic roles. VerbNet was chosen since its definitions of semantic roles have much in common with the CG standard. We experimented on a subset of the ImageClef 2008 collection of titles and annotations of medical images.

Keywords: Information Retrieval, Conceptual Graph, Dependency Grammar.

1 Introduction

The language used in medical literature, as well as in other domains, has its own grammatical peculiarities concerning the usage of noun phrases and terminology. For processing of this type of language it is preferable to use structures that allow representing semantic relations between words. Conceptual Graphs structures allow retaining the relations between words.

However, it is difficult to transform natural language text to Conceptual Graphs structures. We present method for transforming text into simplified conceptual graph-like structure, close to syntactic dependency structure. As a case study, we used these structures in the process of information retrieval and found that it improves the retrieval performance as compared with the standard vector-space model as a baseline.

Our procedure for transforming text to simplified Conceptual Graphs (CG) is based on an adapted grammar, which we manually built for this purpose. This grammar is based on two elements: construction of concept nodes, usually noun phrases, and assigning them specific roles defined by the standards of Conceptual Graphs.

We tested our method on a collection of annotations of medical images ImageClef 2008 [22].

The paper is organized as follows. In Section 2, we explain our motivation. In Section 3, we discuss the importance of CG and its advantages as a computational structure and shortly present the state of the art both of knowledge representation structures and automatic parsing of natural language into Conceptual Graphs. In Section 4, we give more details about conceptual structures. In Section 5, we describe the experimental methodology and experimental results. Finally, Section 6 we concludes the paper and presents future work.

2 The Problem

In medical science, information systems are very important for managing information on human health conditions. However, medical information is presented in natural language. People working in medical institutions choose different words when filling forms; it is difficult to standardize all the vocabulary related with health. In order to be able to analyze medical information, retrieve necessary details, or answer specific queries, natural language processing methods are to be applied. It is desirable that these methods represent natural language information in computational structures.

There are several computational structures used in natural language processing. Simple structures like bag of words, which work only on words without preserving their relations, make processing easier; however, they lose the semantics of the text. Other structures like Conceptual Graphs can preserve many semantic details, but they are complex in management and processing, as well as difficult to obtain.

In this paper we use a simpler structure than full standard Conceptual Graphs and show that it is still useful in an applied task, namely, in Information Retrieval task.

3 Related Work

In this section, we give a brief overview of computational structures, Conceptual Graphs (CGs) in particular, and summarize the state of the art in automatic transformation of natural language texts into CGs.

3.1 Computational Structures for NLP

There are many computational structures used in natural language processing (NLP).

Statistical Computational Structures These include structures ranging from simplest structures, such bag of words and vector space model, to more complex structures, such as graphs or trees. Most frequently mentioned in the literature are vector space model structures and graphs.

The vector space model [32] is simple and most common in practice. This technique consists in extracting words from texts (tokenizing), removing stop words, and reducing the dimensionality (e.g., stemming). The documents are represented as vec-

tors, where each word represents a feature (coordinate) and its value can be either frequency of the given word in the text or presence or just binary: presence or absence of this word in the text.

The documents can be represented as graphs in many ways [34]. These methods used for this are classified into standard, single, *n*-distance, *n*-single distance, absolute distance, and relative frequency. Each method determines terms and adjacencies used as vertices and arcs of the graph, correspondingly. Rege *et al*. [33] describe many forms of representing documents as graphs and Badia and Kantardzic [2] propose a methodology for construction of graphs via statistical learning. Graphs are widely used in natural language processing, for example, is question answering [26], text classification [4, 10], named entity recognition [5], or information representation [3, 31] (in combination with vector space techniques). In other cases the documents are represented by probabilistic functions [25] or composite function of probabilistic function on words [12, 35].

Linguistic Computational Structures To represent linguistic knowledge, grammar structures are most commonly used. Grammar structures include syntactic structure, morphology, and other linguistic details. Morphology provides the part of speech of each word in the text, as well as its dictionary form. Syntax describes the relations among words. In general, these structures are based on a grammar or a set of structural rules which is language-specific and depend on syntax theories: for example, dependency grammar, link grammar, or constituent-based (phrase structure) grammar.

A structure based on the dependency grammar is determined by the relation between a word which functions as the head and the other words dependent on it. In this structure, the order of the words or their position in the sentence does not matter [39]. The link grammar builds undirected relations between pairs of words [36]. In the constituents-based grammar, runs of adjacent words that express a text unit are determined and labeled with constituent category, such as Noun Phrase (NP), Verb Phrase (VP), Noun, etc. This technique builds a tree by iterative process of segmentation and grammatical classification of runs of words in the sentence.

Knowledge Representation Languages These languages are designed to describe semantics and concepts. Examples of this type of languages are *Frame Representation Language* (FRL) and *Descriptions Logics* (DL) [7]. The language of first type (FRL) is defined with meta-language and is based on frames. The frames are oriented to the recognition of objects and classes. Frames have names and features, numeric or otherwise. FRL heavily relies of inheritance mechanisms.

Conceptual Structures The structures of this category, unlike the statistical-oriented structures, are intended for knowledge representation. As examples we can mention Semantic Networks, Conceptual Graphs, the Knowledge Interchange Format (KIF), the Resource Description Framework (RDF) of World Wide Web Consortium (W3C), an ontological language Web Ontology Language (OWL) of W3C [9], and Common Logic (CL).

RDF is a language for information representation in Web resources. It represents documents' metadata such as the title, author, and date. OWL is a markup language for publishing and sharing ontologies on the Web. Semantic network [13, 37] is a mathematical model that shares many features with Conceptual Graphs.

3.2 Conceptual Graphs and their Representation

Conceptual Graphs (CGs) for representing text were introducing by Sowa [37]. They are bipartite digraphs. They have two types of nodes: concepts and relations. A relation node indicates the semantic role of the incident concepts. Since CGs are semantically very rich, they are suitable for knowledge representation, including knowledge bases and ontologies. There are relatively few works, however, aimed at construction of CGs. Three trends can be mentioned: (1) methodology for manual development of CGs; (2) automatic transformation of natural language text into CGs using deterministic approaches, and (3) automatic transformation using statistical approaches.

Deterministic Automatic Transformation In one of his pioneering works, Sowa [38] proposed a procedure to build Conceptual Graphs based in four elements: (a) Type label (concepts and relations); (b) Canonical graph. The Canonical Graphs corresponds to the graphs that connect relation and concept nodes with their restrictions; (c) Type definition. Some concepts and relations are defined with primitives, while others can be defined by lambda abstractions; (d) Schema. A concept type may have one o more schemas that specify the corresponding knowledge.

Other works present step by step construction of each element of the graphs. Hernández Cruz [18] presents a converter or Spanish text into Conceptual Graphs, based on previous syntactic analysis. Amghar *et al*. [1] describes how to convert French texts into Conceptual Graphs using cognitive patters. In medical context, Rassinoux *et al*. [28, 29] generate annotations for the text, which they use to construct CGs.

Reddy *et al*. [30] present an implementation of a CG-like data structure. Conceptual graphs serve as knowledge representation in the systems LEAD (Learning Expert system for Agricultural Domain) and XLAR (Universal Learning Architecture). There, CGs are constructed via frames, which represent features of objects. Castro-Sanchez and Sidorov [8] extract semantic role and valency information from human-oriented dictionaries.

Hensman *et al*. [15, 16, 17] use WordNet and VerbNet for identifying the semantic roles. All documents are converted into XML format and then parsed with Charniak's probabilistic parser, which produces trees in Penn Treebank-style formalism based on constituent grammar. Then the roles are identified using VerbNet. For each clause in the sentence, the main verb is identified and a sentence pattern is built using the parse tree. For each verb in the sentence, they extract all possible semantic frames from VerbNet, taking into account the constraints of the roles in VerbNet.

Statistical Automatic Transformation Hensman [14] first transforms documents into Extensible Markup Language (XML). Then she identifies semantic roles using VerbNet, WordNet and a parser. Barrière and Barrière [6] describe the construction of

CGs using tag words and a parser. Then they disambiguate the CGs. For transforming the grammatical rules into CGs they use heuristics methods. Other researchers use link grammar [19, 40] or dependency grammar [21]. In the latter work the authors use supervised learning to classify concepts, relations, and structures.

4 Building the Structure

We used a simplified structure, which is basically syntactic structure minimally adapted to semantics represented in conceptual graphs.

For assignment of semantic roles, we used the verb lexicon VerbNet [20]. It is organized into verb classes. Each class contains a set of syntactic descriptions that include a verb and the elements that depend on it, along with their semantic roles. Basing on this information, we built a dependency grammar, which included verb classifications, their syntactic descriptions, and frame descriptions. Table 1 shows a sample of the rules. The first rule has the roles *agent* and *theme,* the class of the verb is *V_ACCOMPANY-51-7*, and LIS_NP correspond to a list of noun phrases. The @ in the rule marks the head; this allows producing a dependency tree using a context-free parser. The elements within square brackets are optional.

Table 2. Example of alternative rules for the non-terminal SENTENCE

agent:LIS_NP @:V_ACCOMPANY-51-7 theme:LIS_NP
agent:LIS_NP @:V_ACCOMPANY-51-7 theme:LIS_NP [spatial] destination:LIS_NP
actor1:LIS_NP @:V_ACQUIESCE-95 [DEST_DIR] actor2:LIS_NP
agent:LIS_NP @:V_ADDICT-96 patient:LIS_NP
agent:LIS_NP @:V_ADDICT-96 patient:LIS_NP [DEST_DIR] stimulus:LIS_NP
agent:LIS_NP @:V_ADDICT-96 patient:LIS_NP [DEST_DIR] stimulus:LIS_NP
agent:LIS_NP @:V_ADJUST-26-9 patient:LIS_NP

For parsing the obtained grammar, we used the parsing tool [11] developed in the Natural Language Processing Laboratory, CIC-IPN, available from nlp.cic.ipn.mx/tools/parser. The tool produces a dependency tree labeled with the semantic roles indicated in the grammar. Note that in our case the grammar was specially designed for the obtained trees to resemble CGs, and the labels on its arcs were semantic roles. We expect to add in the future more elaborated post-processing of the dependency trees to better approximate semantic graph structures.

5 Information Retrieval with Simplified Conceptual Graphs

Since we represent the documents and the queries as graphs, the main issue for an information retrieval application is the similarity measure between two graphs. The system produces ranking of documents for a given query according to this similarity measure between the query and each document.

To measure the similarity between two graphs $\mathbf{G}_1$ and $\mathbf{G}_2$ as the relative size of their maximum overlap, i.e., the maximum common sub-graph. To find the maximum common sub-graph, we build all maximal common sub-graphs and then choose the largest one. To find maximal common sub-graphs, we use the following procedure.

A vertex mapping between two labeled graphs $\mathbf{G}_1$ and $\mathbf{G}_2$ is a one-to-one correspondence $\varphi : \mathbf{S}_1 \leftrightarrow \mathbf{S}_2$, $\mathbf{S}_i$ is a subset of vertices of $\mathbf{G}_i$, such that the labels on the corresponding vertices (which in our case are the stems of the corresponding words) coincide. We require the corresponding subsets $\mathbf{S}_1$ and $\mathbf{S}_2$ to be maximal in the sense that no supersets of them can be mapped. For example, in *a fat cat sat on a mat and a fat dog slept* and *a fat cat slept and a fat dog sat on a mat*, the first *fat* from the first sentence can be mapped to either first or second occurrence of *fat* in the second sentence, and then the second *fat* is mapped to the other occurrence; the similarly there are six possible mappings of *a*'s, which gives 12 possible mappings in total.

Either one of the isomorphic sets $\mathbf{S}_1 \cong \mathbf{S}_2$ is the vertex set of a maximal common sub-graph. The arcs of this common sub-graph are those arcs that are present, with the same labels, in both graphs between the corresponding vertices of $\mathbf{S}_1$ and $\mathbf{S}_2$, i.e., such arcs that $u \xrightarrow{x} v$, $u,v \in \mathbf{S}_1$, is an arc in $\mathbf{G}_1$, and $\varphi(u) \xrightarrow{x} \varphi(v)$, $f(u), f(v) \in \mathbf{S}_2$, is an arc in $\mathbf{G}_2$. This completes building of a maximal common sub-graph $\mathbf{G}_{12}$.

We score a maximal common sub-graph very similarly to the standard vector similarity score, but combining the counts for words and relations separately:

$$sim(d_1, d_2) = \frac{\alpha \sum_w idf_w + \beta \sum_r idf_r}{\exp\left(\alpha \log \sum f_{w,1}^2 \sum f_{w,2}^2 + \beta \log \sum f_{r,1}^2 \sum f_{r,2}^2\right)}, \quad (1)$$

where w runs over the mapped vertices, that is, the words in common between the two documents (nodes of $\mathbf{G}_{12}$); r runs over arcs (relations) present in both graphs between the corresponding vertices (arcs of $\mathbf{G}_{12}$); *idf* is the standard inverse document frequency measure, calculated both for vertices and for arcs. The frequency for an arc is measured by a triple of a label on the source vertex, label on the relation, and label on the target vertex; for example, $love \xrightarrow{agent} John$ is a unit for counting the *idf*.

The denominator is a standard vector space model normalizing factor, modified to reflect both vertices (words) and arcs (relations). In fact we found that for this dataset it is better not to include this denominator; see below. Finally, α, β are importance weights given to the intersection of the words and the arcs, correspondingly; see below. In fact, only the ratio α/β is what matters, so only one of the two parameters can be chosen independently.

We consider all possible vertex mappings (maximal common sub-graphs $\mathbf{G}_{12}$) between $\mathbf{G}_1$ and $\mathbf{G}_2$; the best score for a mapping is considered as the similarity measure between the two documents.

6 Experimental Results

We experimented with both the proposed representation and with the usual vector space model as a baseline.

Dataset As the test collection we used a subset of annotated collection of medical images of ImageClef 2008, only using the title of the image and its annotation, but not the image itself. By joining the title and annotation of each image, we obtained a collection of 67115 records. We only considered the documents that contained any text, and ignored the documents that only contained an image.

Of these documents, we only experimented with a subset of first 1,000 documents (from 0000003 to 0001684) and first 15,603 documents (from 0000003 to 0026687), due to time limitations. We used 9 queries, namely 22 to 30, because these queries are intended to be answered not only by analyzing the image but also the textual part of the collection. The sample of 1,000 documents contained 160 relevant answers to all questions (counting twice the same answer to two different questions), and the sample of 15,603 documents contained 1,187 relevant answers.

The collection consists of very short documents and even shorter queries. For example, query 25 reads "*Merkel cell carcinoma*," and the first document marked in the collection as relevant for this query is document 79: "*Eight single-level dynamic CT scans (A H) of the abdomen of a 32-year-old woman with abdominal pain. Scans were obtained during injection of 150 mL of nonionic contrast medium (iohexol) at 5.0 mL/sec. Scans show that the pancreas reaches peak enhancement before the liver. Effect of injection rate of contrast medium on pancreatic and hepatic helical CT*". Our initial hypothesis was that for so short texts the usual vector model may prove to be inaccurate and additional semantic information would be useful.

Building Conceptual Graphs We developed an English grammar with the following peculiarities, as described above. In addition to the usual syntactic structure, the grammar includes thematic roles, such as *agent* or *attribute*. These roles were taken from FrameNet, and were selected on the lexical basis, for each verb individually. This gave us more semantic-oriented analysis than a general-purpose grammar that only uses morphosyntactic information.

This grammar recognizes all the words that occur in this collection. To include in it the words for which we did not find morphosyntactic information in WordNet, we used the UMLS tool [24] to determine their part of speech. The labels of the nodes (words) in the graphs were obtained with Porter stemmer [27]; the labels of the arcs were specified in the grammar.

Performance measure To evaluate our system against the gold baseline, we used the Mean Average Precision [23] measure. This is one-number measure defined as

$$MAP = \frac{1}{Q}\sum_{q=1}^{Q}\frac{1}{m_q}\sum_{d=1}^{m_q}P_{qd}\ , \qquad (2)$$

where Q is the total number of queries (9 in our case), m_q is the number of relevant documents for a query, P_{qd} is the precision on the set of documents ranked by the system, for the query q, higher or equally as the document d. The summation is over all relevant documents for the given query.

The ranking (ordering) was defined by the scores calculated according to (1). However, there were many documents scored equally, so the ranking was ambiguous. For the purpose of calculating precision in (2), we used the following formula:

$$P_{qd} = \frac{P(R_{qd}^{>}) + P(R_{qd}^{\geq})}{2}, \quad (3)$$

where P is the precision, $R_{qd}^{>}$ is the set of documents scored higher than d, and $R_{qd}^{\geq}$ is the set of documents scored higher or equally as d.

Baseline: the Vector Space Model To build the vector representation of the documents, we used wordInd of Lexical Tools of UMLS [24] for tokenizing and Porter stemmer [27] for stemming. The vector coordinates were the frequencies (not binary vectors), and the similarity measure was cosine.

To illustrate the behavior of the collection, we show in Figure 1 the precision and recall on each query (query numbers from 21 to 30) for binary retrieval by threshold of cosine > 0.5 and cosine > 0.

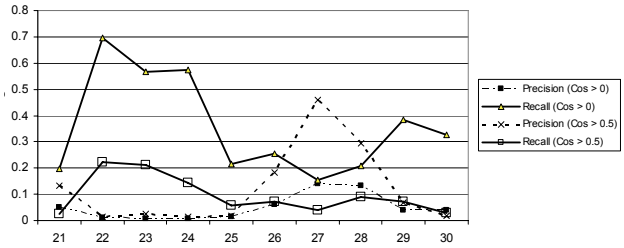

Figure 1. Precision and recall on vector space model for different queries of the collection.

Rather poor performance can be explained by very small size of the queries and documents, as well as by semantically hard nature of the queries, which would require a good medicine domain ontology. We did not use any synonym dictionary or ontology, because the purpose of this work was not to achieve good performance but to compare the options of using or not semantic relations in the text.

However, we do not use precision and recall figures for comparison with our method, since these are set-based measures, while both vector space model and our method produce rankings. The Mean Average Precision for the baseline vector model without taking into account the relations can be observed on the figures below with the zero value of the parameter.

Information Retrieval with Simplified Conceptual Graphs For each document and for each query, we built its semantic representation, varying the weights α, β present in (1). We also considered the possibility not to include the normalizing denominator in (1), i.e., we considered a similarity measure that consisted only of the numerator. Figure 2 shows the value obtained for the performance evaluation measure described above on the sample of 15,603 documents described above, for the parame-

ter $\alpha = 1$ (coincidence of words) and varying β (coincidence of arcs). The left plot shows the experiments without the normalizing denominator in (1), and the right one shows the results for both complete formula (1) and without the denominator (same as on the left, but at larger scale).

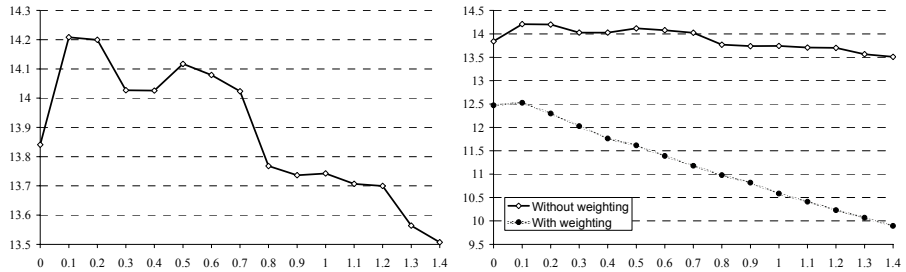

Figure 2. Left: Mean average precision for the large sample, without normalizing weighting in (1), as function of relative weight of graph arcs. Right: The same, together with the results with normalization.

One can observe that, contrary to the assumptions of the vector space model, the formula without the normalizing denominator performs considerably better; we attribute this to the small size of the documents.

The standard vector space model similarity is one with the parameter $\beta = 0$. With small nonzero β the results improve, mainly for the non-normalized variant of (1) (the normalized variant also improves very slightly for β around 0.1). The improvement is not impressive but clearly observable for β everywhere between 0 and 0.7. As the parameter β grows, the results decline. With an infinite β, that is, with $\alpha = 0$, $\beta = 1$, the result was 45%. This is still better than random baseline, which gives 50%, so relations alone, without taking into account words at all, still can be used for retrieval, but the performance is much poorer than that for the words without relations. We believe that this may be due to low recall: for many documents there were no relations in common with the queries, because of too small size of both queries and documents.

Figure 3 present the same data but separately for each query. As expected [23], the results vary greatly from query to query.

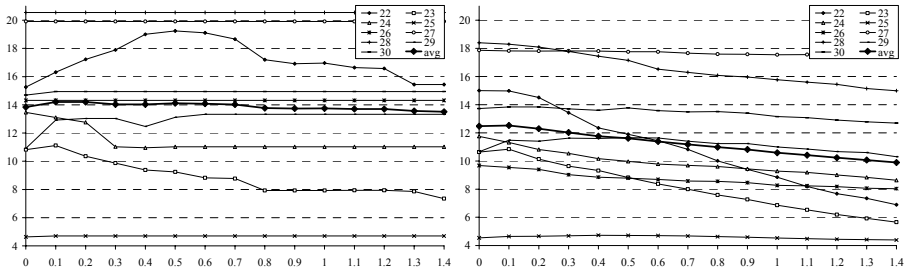

Figure 3. Mean average precision for the large sample, for individual queries as well as averaged over queries (same as Figure 2). Left: with normalization, right: without.

Figure 4 shows the results for a smaller sample (1,000 documents). The sample also shows improvement of the formula with $\beta > 0$ over the baseline $\beta = 0$; in this case the improvement is observable for the variant of the formula with the normalizing denominator.

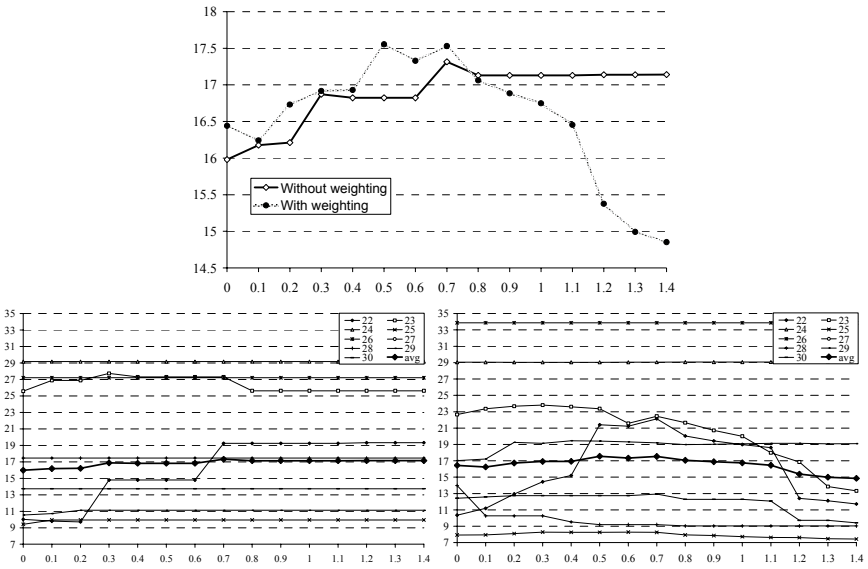

Figure 4. Plots as in Figures 2 and 3, over the small sample.

7 Conclusions and Future Work

We have shown that taking into account semantic relation between words improves the results of information retrieval task. We also briefly presented a methodology for transforming short phrases expressed in natural language into Conceptual Graphs via automatic semantic analysis using lexical resources such as VerbNet.

In the future, we plan to work on better post-processing of the dependency tree into a conceptual graph-like structure and on improvements to our grammar that produces the semantic roles. We will also experiment with other text collections to see whether the method gives greater improvement on collections with larger documents.

Acknowledgements. The work was done during the first author's research stay at the *Laboratorio de Lenguaje Natural y Procesamiento de Texto* of the *Centro de Investigación en Computación* of the *Instituto Politécnico Nacional*, Mexico, partially funded by the *Universidad Nacional de Colombia* and *Universidad Distrital F.J.C.*, Bogota, Colombia, and with partial support of Mexican Government (SNI, CONACYT grant 50206-H, CONACYT scholarship for Sabbatical stay at Waseda U., COFAA-IPN, and SIP-IPN grant 20100773) to the second author.

References

1. Amghar, T.; Battistelli, D. & Charnois, T. Reasoning on aspectual-temporal information in French within conceptual graphs. 14th IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2002, 315–322, 2002.
2. Badia, A. Kantardzic, M. Graph building as a mining activity: finding links in the small. In: Proceedings of the 3rd international workshop on Link discovery LinkKDD'05, ACM pp. 17–24, 2005.
3. Baeza-Yates R., Ribeiro-Neto B., Modern Information Retrieval, ACM Press, Pearson Addison Wesley, 1999.
4. Barbu, E., Heroux, P., Adam, S. and Trupin, E. Clustering document images using a bag of symbols representation. In: Proceedings, Eighth International Conference on Document Analysis and Recognition, Vol. 2, pp. 1216–1220, 2005.
5. Barceló, G., Cendejas, E., Bolshakov, I., and Sidorov G. Ambigüedad en nombres hispanos. Revista Signos. Estudios de Lingüística 42 (70), pp. 153–169, 2009.
6. Barrière, C. and Barrière, N. C. From a Children's First Dictionary to a Lexical Knowledge Base of Conceptual Graphs. St. Leonards (NSW): Macquarie Library, 1997.
7. Barski, C. The enigmatic art of knowledge representation. www.lisperati.com/tellstuff/ind-ex.html. Accessed March 2010.
8. Castro-Sánchez, N. A., and Sidorov, G. Analysis of Definitions of Verbs in an Explanatory Dictionary for Automatic Extraction of Actants based on Detection of Patterns. Lecture Notes in Computer Science, N 6177, pp 233–239, 2010.
9. Delugach, H. S. Towards. Conceptual Structures Interoperability Using Common Logic Computer. Science Department Univ. of Alabama in Huntsville. Third Conceptual Structures Tool Interoperability Workshop, 2008.
10. Figuerola G. C., Zazo F A., Berrocal J. L. A. Categorización automática de documentos en español: algunos resultados experimentales. Universidad de Salamanca, Facultad de Documentación, Salamanca España, pp. 6–16, 2000.
11. Gelbukh A., Sidorov, G., Galicia, S., Bolshakov, I. Environment for Development of a Natural Language Syntactic Analyzer. Acta Academia, Moldova, pp. 206–213, 2002.
12. Griffiths T. L. and Steyvers M. Finding scientific topics. In Proceedings of the National Academy of Sciences, 101 Suppl. 1, pp. 5228–5235, 2004.
13. Helbig H. Knowledge Representation and the Semantics of Natural Language. Springer, 2006.
14. Hensman, S. Construction of Conceptual Graph representation of texts. Department of Computer Science, University College Dublin. Belfield, Dublin 4. Proceedings of Student Research Workshop at HLT-NAACL, 2004.
15. Hensman, S. and Dunnion, J. Automatically building conceptual graphs using VerbNet and WordNet. 2004 international Symposium on information and Communication Technologies, Las Vegas, Nevada, June 16–18, 2004. ACM International Conference Proceeding Series, vol. 90. Trinity College Dublin, pp.115–120, 2004.
16. Hensman, S. and Dunnion, J.. Constructing conceptual graphs using linguistic resources. In Proceedings of the 4th WSEAS international Conference on Telecommunications and informatics, Prague, Czech Republic, March 13–15, 2005. M. Husak and N. Mastorakis, Eds. World Scientific and Engineering Academy and Society (WSEAS), Stevens Point, Wisconsin, pp. 1–6, 2005.
17. Hensman, S. Construction of conceptual graph representation of texts. In Proceedings of the Student Research Workshop at HLT-NAACL 2004 (Boston, Massachusetts, May 02–07, 2004). Human Language Technology Conference. Association for Computational Linguistics, Morristown, NJ, pp. 49–54, 2004.
18. Hernández Cruz, M. Generador de los grafos conceptuales a partir del texto en español. MSc thesis. Instituto Politécnico Nacional, Mexico, 2007.

19. Kamaruddin, S.; Bakar, A.; Hamdan, A., and Nor, F. Conceptual graph formalism for financial text representation. Information Technology. International Symposium, 2008.
20. Kipper K., Korhonen A., Ryant N., and Palmer M. Extending VerbNet with Novel Verb Classes. 5th International Conf. on Language Resources and Evaluation, LREC 2006. Genoa, Italy. June, 2006; verbs.colorado.edu/~mpalmer/projects/verbnet. html.
21. Kovacs, L. and Baksa-Varga, E. Dependency-based mapping between symbolic language and Extended Conceptual Graph. Intelligent Systems and Informatics. 6th International Symposium, 2008.
22. Medical Image Retrieval Challenge Evaluation P., http://ir.ohsu.edu/image.
23. Manning, C. D., Raghavan, P., and Schütze, H. Introduction to Information Retrieval, Cambridge University Press. 2008; http://www-nlp.stanford.edu/IR-book.
24. National Library of Medicine, National of Institute of Health. United States Unified Medical Language System (UMLS), www.nlm.nih.gov/research/umls/about_umls.html, accessed April 2010.
25. Peltonen, J., Sinkkonen, J. and Kaski, S. Discriminative clustering of text documents, In: 9th International Conference on Neural Information Processing, ICONIP'02, pp. 1956–1960, 2002.
26. Pérez-Coutiño M., Montes-y-Gómez M, López-López A. Applying dependency trees and term density for answer selection reinforcement. CLEF 2006, 424–431, 2006.
27. Porter M. An algorithm for suffix stripping, Program, Vol. 14, no. 3, pp 130–137, 1980, Available on http://tartaus.org/~martin/PorterStemmer/.
28. Rassinoux, A. M.; Baud, R. H. & Scherrer, J. R. A Multilingual Analyser of Medical Texts Conceptual Structures. 2nd International Conference on Conceptual Structures, ICCS'94, College Park, Maryland, USA, August 16–20, 1994: Proceedings, 1994.
29. Rassinoux, A. M., Baud, R. H., Lovis, C., Wagner, J. C., and Scherrer, J. R. Tuning Up Conceptual Graph Representation for Multilingual Natural Language Processing in Medicine Conceptual Structures: Theory, Tools, and Applications. 6th International Conference on Conceptual Structures, ICCS'98, Montpellier, France, August 1998: Proceedings, 1998.
30. Reddy K. C., Reddy C. S. K., and Reddy P. G.. Implementation of conceptual graphs using frames in lead. In S. Ramani, R. Chandrasekar, K.S. R. Anjaneyulus (Eds.). International Conference KBCS'89 Bombay, India, December. LNCS Knowledge Based Computer Systems Volume 444, pp. 213–229, Springer, 1990.
31. Rege, M. Dong, M., and Fotouhi, F. Co-clustering Documents and Words Using Bipartite Isoperimetric Graph Partitioning. In: proceedings Sixth International Conference Data Mining ICDM'06, pp. 532–541, 2006.
32. Salton, G. Relevance assessments and Retrieval system evaluation, Information Storage and retrieval, 1969.
33. Schenker A., Bunke, M. L H. and Kandel, A. A Graph-Based Framework for Web Document Mining. In LNCS, vol. 3163, pp 401–412 Springer, Heidelberg, 2004.
34. Schenker A., Bunke, M. L. A. H. and Kandel A.A. Graph-Theoretic Techniques for Web Content Mining World Scientific Publishing, 2005.
35. Shafiei, M. and Milios, E. Latent Dirichlet Co-Clustering. In Sixth International Conference on, Data Mining (CDM'06), pp. 542–551, 2006.
36. Sleator, D. and Temperley, D. Parsing English with a link grammar. Third International Workshop on Parsing Technologies, 1993.
37. Sowa, J. F. Conceptual Graphs. Handbook of Knowledge Representation, 2008
38. Sowa, J. F. and Way, E. C. Implementing a semantic interpreter using conceptual graphs. IBM Journal of Research and Development, 30(1): pp 57–69, January 1986.
39. Tesnière L. Éléments de syntaxe structurale, Klincksieck, Paris, 1959.
40. Williams, R. A. Computational Effective Document Semantic Representation. DEST'07. Digital EcoSystems and Technologies Conference, IEEE-IES, 2007.