Information Retrieval with Conceptual Graph Matching

Manuel Montes-y-Gómez [1], Aurelio López-López [2], Alexander Gelbukh [1]

[1] Center for Computing Research (CIC),
National Polytechnic Institute (IPN),
Av. Juan Dios Bátiz s/n esq. Mendicabal, col. Zacatenco, CP. 07738, DF, Mexico.
mmontesg@susu.inaoep.mx, gelbukh@cic.ipn.mx

[2] INAOE.
Luis Enrique Erro No. 1, Tonantzintla, Puebla, 72840 México.
allopez@inaoep.mx

Abstract. The use of conceptual graphs for the representation of text contents in information retrieval is discussed. A method for measuring the similarity b etween two texts represented as conceptual graphs is presented. The method is based on well-known strategies of text comparison, such as Dice coefficient, with new elements introduced due to the bipartite nature of the conceptual graphs. Examples of the representation and comparison of the phrases are given. The structure of an information retrieval system using two-level document representation, trad itional keywords and conceptual graphs, is presented.

1. Introduction*

In many application areas of text analysis, for instance, in information retrieval and in text mining, shallow representations of texts have been recently widely used. In information retrieval, such shallow representations allow for a fast analysis of the information and a quick respond to the queries. In text mining, such representations are used because they are easily extracted from texts and easily analyzed.

Recently in all text-oriented applications, there is a tendency to begin using more complete representations of texts than just keywords, i.e., the representations with more types of textual elements. For instance, in information retrieval, these new representations increase the precision of the results; in text mining, they ext end the kinds of discovered knowledge.

A method for the comparison of texts in such a representation is one of the main prerequisites to begin using the new representation in various applications of text processing. In this paper, we discuss the use of the conceptual graphs to represent the contents of documents for information retrieval and text mining. This representation incorporates the information about both the concepts mentioned in the text and their relationships, e.g., *[binary]* ← *(attr)* ← *[search]*. We present a method for measuring the similarity between two phrases represented as conceptual graphs. This method does not depend on the kind of concepts and relations used in the graphs.

*The work was done under partial support of CONACyT, REDII, and SNI, Mexico.

First, we discuss the previous works concerning the comparison between two texts, introduce the notion of the conceptual graph, and describe the process of transformation of a text to a set of conceptual graphs. Then, we explain the main idea of the comparison of two conceptual graphs, and give the corresponding formulae. Finally, we discuss the possible applications of the method for information retrieval, and give some examples.

2. Related work

The comparison of text representations has been widely discussed in the literature. Important related work has been done in information retrieval, document clustering, conceptual clustering, and recently in text mining.

In information retrieval and document clustering, the weighted-keyword representation of documents is one of the most widely used [8, 9]. For this type of document representation, many different similarity measures are proposed, for instance, the Dice coefficient, the Jaccard coefficient, the Cosine coefficient [8], etc.

For the representation with binary term weights, the Dice coefficient is calculated as follows:

Dice coefficient:

$$S_{D_1, D_2} = \frac{2n(D_1 \cap D_2)}{n(D_1) + n(D_2)}$$

where $n(D_i)$ is the number of terms in D_i, and $n(D_i \cap D_j)$ is the number of terms that the two documents D_i and D_j have in common.

Because of its simplicity and normalization, we take it as the basis for the similarity measure we propose.

In information retrieval, some other kinds of representations different from the keyword representation have been used, for instance, conceptual graphs [2, 7]. For these representations, different similarity measures have been described for comparing the query graph and the document graphs. One of the main comparison criteria used for conceptual graphs is that if a query graph is completely contained in the document graph, then the given document is relevant for the given query. This criterion means that the contents of a document must be more particular than the query, for the document to be relevant for the query.

In text mining, text representations and similarity measures borrowed from information retrieval are widely used. Other representations and measures, for instance, probability distributions of topics and other statistical parameters, have been used too [1, 5]. Usually such representations and measures are tuned to improve the discovering of knowledge in texts.

3. Conceptual graphs

To compare two texts, e.g., a document and the user's query, first their representations in the form of conceptual graphs are built. A conceptual graph is a network of concept nodes and relation nodes [10, 11]. The concept nodes represent entities, attributes, or events (actions); they are denoted with brackets. The relation nodes identify the kind of relationship between two concept nodes; they are denoted with parentheses. At present, we consider relations from a few basic types, such as *attribute*, *subject*, *object*, etc. Thus, a phrase *John loves Mary* is represented with a graph like [John] ← (subj) ← [love] → (obj) → [Mary], and not like [John] ← (love) → [Mary].

In the system we developed, to build a conceptual graph representation of a phrase, a part-of-speech tagger, a syntactic parser, and a semantic analyzer are used. For example, given the phrase

Algebraic formulation of flow diagrams,

first, the part-of-speech tagger supplies each word with a syntactic-role tag, given after the bar sign:[1]

Algebraic|JJ formulation|NN of|IN flow|NN diagrams|NNS .|.

Then a syntactic parser generates its structured representation:[2]

[[np, [n, [*formulation*, sg]], [adj, [*algebraic*]], [*of*,
[np, [n, [*diagram*, pl]], [n_pos, [np, [n, [*flow*, sg]]]]]]], '.'].

The semantic analyzer generates one or more conceptual graphs out of such syntactic structure:[3]

[algebraic] ← (attr) ← [formulation] → (of) → [flow-diagram:*]

In this graph, the concept nodes represent the elements mentioned in the text, for example, nouns, verbs, adjectives, and adverbs, while the relation nodes represent some kind of relation between the concepts (prepositions are maintained to avoid the difficult problem of resolving the semantic relation they express). At the moment, we use only a limited set of relations but we plan to extend it to include some domain-specific semantic relations, and to start using more elements of the conceptual graph formalism, for instance, *n*-ary relations and contexts.

4. Comparison of conceptual graphs

For the purposes of information retrieval or text mining, it is important to be able to compare two phrases or texts represented with conceptual graphs. In particular, in

[1] The tagger we use is based on the Penn Treebank tagset.

[2] The parser we use was developed by Tomek Strzalkowski of the New York University basing on The Linguist String Proyect (LSP) grammar designed by Naomi Sager.

[3] We do not discuss here the structure of the semantic analyzer we use.

information retrieval one of the text is the document and the other one is the user's query. Each phrase or text may be represented as a set of conceptual graphs, for instance, a long phrase, or a whole text consisting of many phrases, are represented as a set of conceptual graphs.

In general terms, our algorithm for the comparison of two conceptual graph representations of two texts consists of two main parts:

1. Find the intersection of the two (set of) graphs,
2. Measure the similarity between the two (set of) graphs as the relative size of each one of their intersection graphs.

In general, we can find more than one subgraph as the intersection of the initial graphs, but the measurement of similarity is applied to each one of them separately, and only the highest value is kept. For the sake of explanation, we hereon deal with only one intersection subgraph.

In the first step, we build the intersection $G_1 \cap G_2 = G_c$ of the two original conceptual graphs G_1 and G_2. This intersection consists of the following elements:

- All concept nodes that appear in both original conceptual graphs G_1 and G_2;
- All relation nodes that appear in both G_1 and G_2 and relate the same concept nodes.

An example of such an intersection is shown on Figure 1. We show the concept nodes such as [*John*] or [*love*] as the points *A*, *B*, etc., and the relation nodes such as (subj) or (obj) as arcs. In the figure, of the concept nodes *A*, *B*, *C*, *D*, *E*, etc., only the concepts *A*, *B*, and C belong to both graphs G_1 and G_2. Though three arcs $A — B$, $A — C$, and $B — C$ are present between these concepts in G_1, only two of them are present in both graphs (with bold lines). Of course, for the arc between two common concepts to be included in the G_c, it should have the same label and orientation (not shown in Figure 1) in the two original graphs.

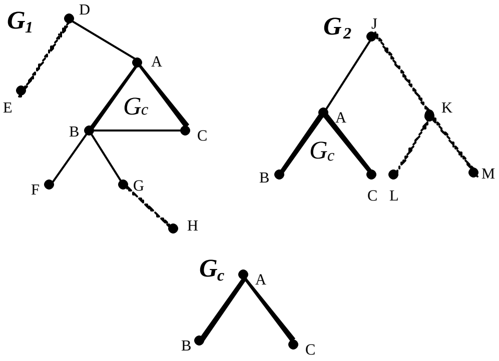


Fig.1. Intersection of two conceptual graphs

In the second step, we measure the similarity between the graphs G_1 and G_2 based on their intersection graph G_c. The similarity measure is a value between 0 and 1, where 0 indicates that there is no similarity between the two texts, and 1 indicates that the two texts are semantically equivalent.

Because of the bipartite (concepts and relations) nature of the conceptual graph representations, the similarity measure is defined as a combination of two types of similarity: the conceptual similarity and the relational similarity:

- The conceptual similarity measures how similar the concepts and actions mentioned in both texts are (like topical comparison).
- The relational similarity measures the degree of similarity of the information about these concepts (concept interrelations) communicated in the two texts.

5. Similarity measure

Given two texts represented by the conceptual graphs G_1 and G_2 respectively and one of their intersection graphs G_c, we define the similarity s between them as a combination of two values: their conceptual similarity s_c and their relational similarity s_r.

The conceptual similarity s_c expresses how many concepts the two graphs G_1 and G_2 have in common. We calculate it using an expression analogous to the well-known Dice coefficient [8]:

$$s_c = \frac{2n(G_c)}{n(G_1) + n(G_2)}$$

where $n(G)$ is the number of concept nodes of a graph G. This expression varies from 0 when the two graphs have no concepts in common to 1 when the two graphs consist of the same set of concepts.

The relational similarity s_r indicates how similar the relations between the same concepts in both graphs are, that is, how similar the information communicated in both texts about these concepts is. In a way, it shows how similar the contexts of the common concepts in both graphs are.

We define the relational similarity s_r to measure the proportion between the degree of connection of the concept nodes in G_c, on the one hand, and the degree of connection of the same concept nodes in the original graphs G_1 and G_2, on the other hand. With this idea, a relation between two concept nodes conveys less information about the context of these concepts if they are highly connected in the original graphs, and conveys more information when they are weakly connected in the original graphs. We formalize this using a modified formula for the Dice coefficient:

$$s_r = \frac{2m(G_c)}{m_{G_c}(G_1) + m_{G_c}(G_2)}$$

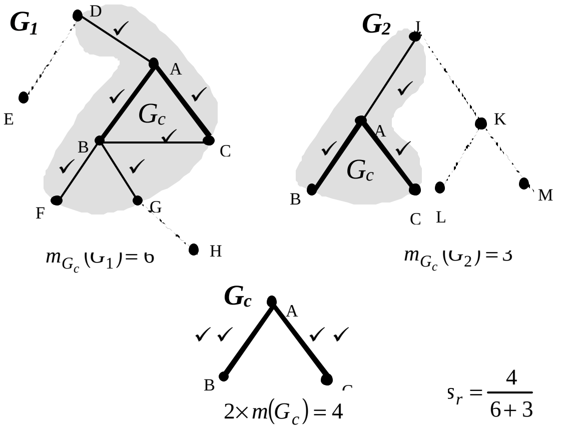


Fig. 2. Calculation of relational similarity

where $m(G_c)$ is the number of the arcs (the relation nodes in the case of conceptual graphs) in the graph G_c, and $m_{G_c}(G_i)$ is the number of the arcs in the immediate neighborhood of the graph G_c in the graph G_i. The immediate neighborhood of $G_c \subseteq G_i$ in G_i consists of the arcs of G_i with at least one end belonging to G_c.

Figure 2 illustrates these measures. In this figure, the nodes *A*, *B* and *C* are the conceptual nodes common for G_1 and G_2 and thus belonging to G_c. Bold lines represent the arcs (relation nodes) common to the two graphs. The arcs marked with the symbol ✓ constitute the immediate neighborhood of the graph G_c (highlighted areas), their number is expressed by the term $m_{G_c}(G_i)$ in the formula above.

The value of $m_H(G)$ for a subgraph $H \subseteq G$ in practice can be calculated as: $m_H(G) = \sum_{c \in H} deg_G c - m(H)$, where $deg_G c$ is the degree of concept node c in the graph G, i.e., the number of the relation nodes connected to the concept node c in the graph G, and $m(H)$ is the number of relation nodes in the graph H.

Now that we have defined the two components of the similarity measure, s_c and s_r, we will combine them into a cumulative measure s. First, the combination is to be roughly multiplicative, for the cumulative measure to be roughly proportional to each of the two components. This would give the formula $s = s_c \times s_r$. However, we can note that the relational similarity has a secondary importance, because it existence depends of the existence of some common concepts nodes, and because even if no common relations exist between the common concepts of the two graphs, some level of similarity exists between the two texts. Thus, while the cumulative similarity measure is proportional to s_c, it still should not be zero when $s_r = 0$. So we will smooth the effect of s_r:

$$s = s_c \times (a + b \times s_r),$$

With this definition, if no relational similarity exists between the graphs, that is, when $s_r = 0$, the general similarity only depends of the value of the conceptual similarity. In this situation, the general similarity is a fraction of the conceptual similarity, where the coefficient a indicates the value of this fraction.

The values of the coefficients a and b depend on the structure of the graphs G_1 and G_2 (i.e. their value depend on the degree of connection of the elements of G_c in the original graphs G_1 and G_2). We calculate the values of a and b as follows:

$$a = \frac{2n(G_c)}{2n(G_c) + m_{G_c}(G_1) + mG_c(G_2)}$$

where $n(G_c)$ is the number of concept nodes in G_c and $m_{G_c}(G_1) + m_{G_c}(G_2)$ is the number of relation nodes in G_1 and G_2 that are connected to the concept nodes appearing in G_c.

With this formula, when $s_r = 0$, then $s = a \times s_c$, that is, the general similarity is a fraction of the conceptual similarity, where the coefficient a indicates this portion.

Thus, the coefficient a expresses the percentage of information contained only in the concept nodes (according to their surrounding). It is calculated as the proportion between the number of common concept nodes (i.e. the concept nodes of G_c) and the total number of the elements in the context of G_c (i.e., all concept nodes of G_c and all relation nodes in G_1 and G_2 connected to the concept nodes that belong to G_c).

When $s_r = 1$, all information around the common concepts is identical and therefore they express the same things in both texts. In this situation, the general similarity takes it maximal similarity value $s = s_c$, and consequently $a + b \times s_r = 1$. Thus, the coefficient b is equal to $1 - a$.

6. Uses in information retrieval

Nowadays, with the electronic information explosion caused by Internet, increasingly diverse information is available. To handle and use such great amount of information, improved search engines are necessary. The more information about documents is preserved in their formal representation used for information retrieval, the better the documents can be evaluated and eventually retrieved.

Based on these ideas, we are developing a new information retrieval system. This system performs the document selection taking into account two different levels of document representation.

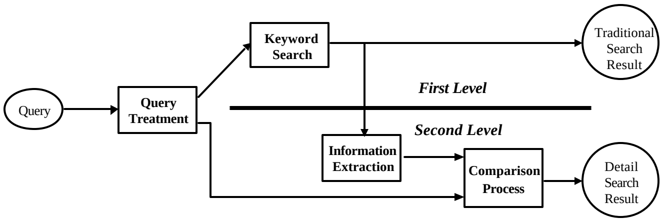


Fig. 3. Calculation of relational similarity.

The first level is the traditional keyword document representation. It serves to select all documents potentially related to the topic(s) mentioned in the user's query. The second level is formed with the conceptual graphs refl ecting some document details, for instance, the document intention. This second level complements the topical information about the documents and provides a new way to evaluate the rel e-vance of the document for the query.

Figure 3 shows the general architecture of our information retrieval system with two-level document selection. In this system, the query-processing module analyses the query and extracts from it a list of topics (keywords). The keyword search finds all relevant documents for such a keyword-only query. Then, the information extraction module constructs the conceptual graphs of the query and the retrieved documents, according to the process described in section 3. This information is currently extracted from titles [6] and abstracts [4] of the documents. These conceptual graphs describe mainly the intention of the document, but they can express other type of relations, such as cause-effect relations [3].

The following example is a conceptual graph extracted from a document abstract.

[demonstrate] → (obj) → [validity: #] → (of) → [technique: #]

This graph indicates that the document in question has the intention of *demonstra t-ing the validity of the technique*.

Then the query conceptual graph is comp ared – using the method described in this paper – with the graphs for the potentially relevant documents. The documents are then ordered by their value s of the similarity to the query.

After this process the documents retrieved at the beginning of the list will not only mention the key-topics expressed in the query, but also describe the intentions spec i-fied by the user.

This technique allows improving the retrieval of information in two main directions:

1. It permits to search the information using not only topical information, but also extratopical, for instance, the document intentions.
2. It produces a better raking of those documents closer to the user needs, not only in terms of subject.

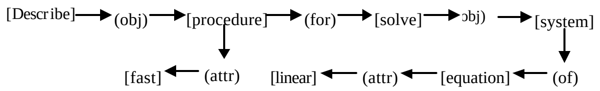
7. Preliminary Experiments

To test these ideas we assembled a small experiment in information retrieval, where both query and documents were represented as conceptual graphs. The document collection we used in this experiment was the CACM-3204, containing 3204 surrogates of articles in Computer Science.

The first step of the experiment was to construct conceptual graphs for the query and 512 of document titles. These conceptual graphs described the intention of the document (as explained above), and were constructed following the method detailed in section 3. They represent mainly the syntactic level of the texts. The syntactic relations used in the graphs are the following: obj (relates verbs with their objects), subj (relates verbs with their subjects), attr (relates nouns or verbs with their attributes, for instance, adjectives and adverbs) and prepositions (specific prepositions, such as *of*).

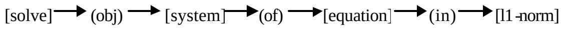
Then, as a second step, we compared the query graph with the conceptual graphs of the documents using the described comparison method, and finally returned to the user those document titles having the highest similarity values with the query.

For instance, for the query "*description of a fast procedure for solving a system of linear equations*", we built the following conceptual graph:

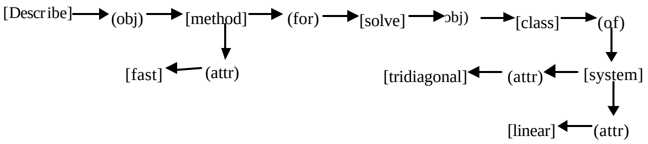


Then, we compared this query-graph with the document graphs and found as relevant documents, among other, the following:

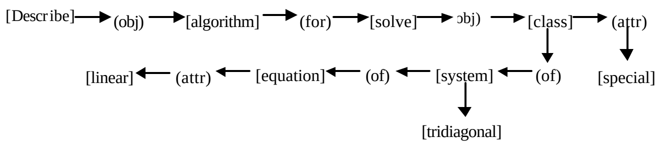
1) The document 2642:



2) The document 2697:



3) The document 1910:



The results for the comparison of these three conceptual graphs with the query are detailed in table 1.

Table 1. An IR experiment using CG.

Graph	G_c	s_c	a	s_r	S
2642	[solve]→(obj)→[system]→(of)→[equation]	0.5	0.42	**0.5**	**0.357**
2697	[describe] [fast] [solve] [system] [linear]	**0.53**	0.42	0	0.224
1910	[describe] [solve] [system]→(of)→[equation]→(attr)→[linear]	0.5	**0.44**	**0.5**	0.333

In spite of the simplicity of this experiment, we can observe the main properties of the measure and how the conceptual and relational similarities are combined to produce the final measure. For instance, the results showed that our measure indeed values more graphs with connected common elements than graphs with a larger number of common concepts but not connected. This means that our similarity measure focuses on what the text tells about the concepts (interconnection of concepts) and not only on the concepts it mentions per se.

8. Conclusions

We have described the structure of an information retrieval system that uses the comparison of the document and the query represented with conceptual graphs to improve the precision of the retrieval process by better ranking on the results. In particular, we have described a method for measuring the similarity between conceptual graph representations of two texts. This method incorporates some well-known characteristics, for instance, the idea of the Dice coefficient – a widely used measure of similarity for the keyword representations of texts. It also incorporates some new characteristics derived from the conceptual graph structure, for instance, the combination of two complementary sources of similarity: the conceptual similarity and the relational similarity.

This measure is appropriate for text comparison because it considers not only the topical aspects of the phrases (difficult to obtain from short texts) but also the relationships between the elements mentioned in the texts. This approach is especially good for short texts. Since in information retrieval, in any comparison operation at least one of the two elements, namely, the query, is short, our method is relevant for information retrieval.

Currently, we are adapting this measure to use a concept hierarchy given by the user, i.e. an *is-a* hierarchy, and to consider some language phenomena as, for example, synonymy.

However, the use of the method of comparison of the texts using their conceptual graph representations is not limited by information retrieval. Other uses of the method include text mining and document classification.

References

1. Feldman, R., and I. Dagan (1995). “Knowledge Discovery in Textual databases (KDT)”. Proc. Of the 1^{st} International conference on Knowledge discovery (KDD_95), pp.112-117, Montreal, 1995.
2. Genest D., and M. Chein (1997). “An Experiment in Document Retrieval Using Conceptual Graphs”. Conceptual structures: Fulfilling Peirce´s Dream. Lecture Notes in artificial Intelligence 1257. August 1997.
3. Khoo, Christopher Soo-Guan (1997). “The Use of Relation Matching in Information Retrieval”. Electronic Journal ISSN 1058-6768, September 1997.
4. López-López, Aurelio, and Sung H. Myaeng (1996). “Extending the capabilities of retrieval systems by a two level representation of content”. Proceedings of the 1st Australian Document Computing Symposium, 1996.
5. Montes-y-Gómez, M., A. López-López, A. Gelbukh (1999a). “Text Mining as a Social Thermometer”. In Procs. Workshop on Text Mining: Foundations, Techniques and Applications, Sixteenth International Joint Conference on Artificial Intelligence (IJCAI-99), Stockholm, Sweden, August 1999.
6. Montes-y-Gómez, M., A. Gelbukh, A. López-López (1999b). “Document Title Patterns in Information Retrieval”, Proc. of the Workshop on Text, Speech and Dialogue TDS’99, Plzen, Czech Republic, September 1999.
7. Myaeng, Sung H. (1990). “Conceptual Graph Matching as a Plausible Inference Technique for Text Retrieval”. Proc. of the 5th Conceptual Structures Workshop, held in conjunction with AAAI-90, Boston, Ma, 1990.
8. Rasmussen, Edie (1992). “Clustering Algorithms”. Information Retrieval: Data Structures & Algorithms. William B. Frakes and Ricardo Baeza-Yates (Eds.), Prentice Hall, 1992.
9. Salton, Gerald (1983). “Introduction to Modern Information Retrieval”. McGraw Hill, 1983.
10. Sowa, John F. (1983). “Conceptual Structures: Information Processing in Mind and Machine”. Ed. Addison-Wesley, 1983
11. Sowa, John F. (1999). “Knowledge Representation: Logical, Philosophical and Computational Foundations”. 1^{st} edition, Thomson Learning, 1999.