Information retrieval by fuzzy relations and hierarchical co-occurrence[1]

Part II

László T. Kóczy[2]
Dept. of Telecommunication and Telematics
Technical University of Budapest
Budapest H-1521 Hungary
Fax: +36-1-463-3107 Phone: +36-1-463-4190 E-mail: koczy@boss.ttt.bme.hu

Tamás D. Gedeon
Dept. of Information Engineering
School of Computer Science and Engineering
University of New South Wales
Sydney 2052 Australia
Fax: +61-2-9385-5995 Phone: +61-2-9385-3965 E-mail: tom@cse.unsw.edu.au

Abstract

1. Introduction

An information retrieval system allows users to efficiently retrieve documents that are relevant to their current interests. The main problem is that the collection of documents from which the selected ones have to be retrieved might be extremely large, and often heterogeneous from various points of view: especially in the structure and the use of terminology. This is very obvious with areas where the language of the documents is close to natural language usage like in legal texts that form the main target of this study. In Part I of this study some mathematical tools were introduced for modeling the interdependence of individual words occurring in both the heading or some special parts of a documents, and the full texts. The method was called Hierarchical co-occurrence method. However, the simple models and search methods discussed there were not suitable for differentiating among various meanings of words apparent from their contexts. In this part we will attempt to refine the model based however still on the same fuzzy relational maps, in order to be able to identify those documents of a collection that are really significant from the point of view of the query.

A user typically specifies their interests via a set of individual words or expressions (phrases), that are fragments of natural language texts. The words occurring among those specified in the query might have two or more meanings, and the user is normally interested only in documents that use that particular word in a certain sense. While it is impossible to decide the preferred meaning of a particular word if the word stands alone, it might be quite possible if there is a set of words

[1] Supported by the Australian Research Council, Grant No. A49600961.
[2] Visiting Professor at the Dept. of Information Engineering, School of Computer Science and Engineering, University of New South Wales, E-mail: koczy@cse.unsw.edu.au.

given, and it is possible to compare the various meanings of each of the words in the set, leaving only those which might be connected. In this part we will propose a method that might be used for such purposes successfully.

2. Reduced hierarchical co-occurrence map

In Part I the concepts of fuzzy relation in general, fuzzy similarity and tolerance were discussed. Based on these, it was possible to establish fuzzy relational matrices or graphs to describe the degrees of co-occurence that we consider a good "measure" of connected meaning.

Let us assume now that we have a set of documents $D = \{D_1, D_2, ..., D_n\}$ and a set of keywords in the sense of Part I that is denoted by W, while the set of all significant words is denoted by *w*, and $W \subset w$. Let the sizes of these sets be $k = |W|$, and $m = |w|$. Consequently, the numbers of edges in the binary co-occurence relational graphs are

$$|G_W| = \binom{|W|}{2} = \frac{k(k-1)}{2} \text{ and } |G_w| = \binom{|w|}{2} = \frac{m(m-1)}{2}.$$

The hierarchical co-occurence graph, on the other hand has

$$|G_{Ww}| = km$$

edges.

If the topic of the documents under query is not very restricted, it is reasonable to assume that the number of significant words is rather high, at least in the order of 1000 (or several thousands). On the other hand, the keywords can be selected so that their total number do not exceed a few hundreds (or remain about 100). If we assume e.g. 100 keywords and 1000 significant words in total, the three sizes in question will be $|G_W| = \binom{100}{2} = \frac{100 \times 99}{2} = 4950$, $|G_w| = \binom{1000}{2} = \frac{1000 \times 999}{2} = 499500$, $|G_{Ww}| = 100 \times 1000 = 100000$. While the keyword co-occurrence graph is fairly small, the hierarchical co-occurrence one is very large and the significant word co-occurrence one is even larger than the latter, by almost one order of magnitude. This will remain so if the number of significant words is at least ten times bigger than the keyword set, which is however a reasonable assumption for every practical information retrieval system based on hierarchical co-occurence. With these sizes of the graphs it is justifiable to consider the restriction of the established co-occurrence map to the one shown in Fig. 1, rather than the full map shown in Fig. 8 in Part I. This type of relational map will be called a reduced hierarchical co-occurrence map.

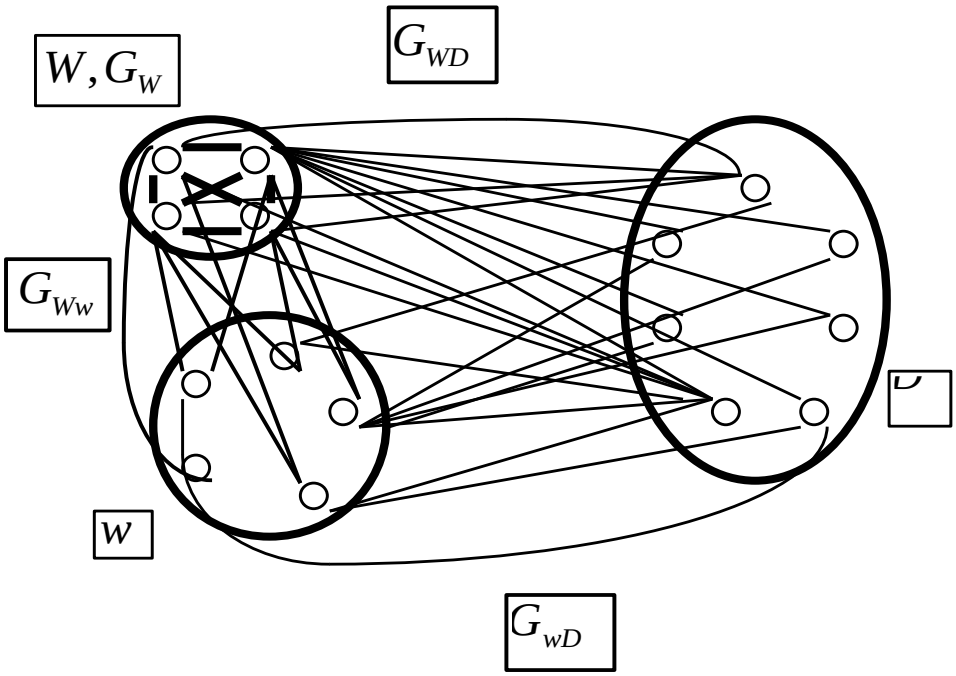

Figure 1.

3. Semantic connection maps

The main idea of semantic connection established by co-occurence degrees introduced in Part I was the following. If a certain word or phrase is frequently occurring together with another one in the same document, the two might have connected meaning or semantics. Further, if a word or phrase is frequently occurring in a document, or segment of a document of which the keywords (in the title, etc.) are certain other words, the former ones would belong to the class of semantically related concepts of the latter ones.

In Section 3 of Part I, a short overview of fuzzy relations was given with special stress on binary fuzzy relations similarity and tolerance (over $X \times X$), which play a significant part in featuring the degrees of being connected or related for *groups of nodes* (consisting usually of at least three elements). Such groups were referred to as α*-cliques* of the respective graph. Usually it cannot be guaranteed that these cliques are disjoint, i.e. the "partition " of the graph will be a cover in reality. This is due to the fact that some or all of the keywords/ significant words have more than one class of words which they are connected to in the sense of their meanings or contextual connectedness. This property will be used for establishing likely semantic connections and classes of the words. From the mathematical point of view it is insignificant whether the base set X of the relation is W or w. Because of the difficulty involved with the extremely large dimensions of G_w, usually we assume that the semantic connection map will be determined for G_W, however, if the size and speed of the available computer are sufficient, certainly the two together might contain more detailed information concerning co-occurrences than the latter one only.

While in Part I several methods were proposed, each of which was based on the idea of search initiated by the query of a single word, in this part we suggest a

family of methods which are always based on a certain group of assumedly coherent or connected words in the query. We assume that the user is often interested in documents that handle a certain *topic* rather than a concrete word or phrase (or a set of words, etc.). In order to do so they specify a set of words or phrases that they consider adequate for describing the topic. However, it must be taken into consideration that this list of words, etc. will usually not be complete as regarding all possible important keywords of the topic. On the other hand, it might be difficult to decide the real topic that the query tries to specify when one or several words have different meanings and contexts.

Let us explain it by a simple example: The user enters a query for "play". The word has several related but still different interpretations. If there is no other word added, a large number of documents will be retrieved where one or some of the subsequent conditions is fulfilled:

- the word "play" occurs in the heading or some important part (keyword search, Method 1 in Part I)
- the word occurs in the text of the document (at least once, or frequently, depending on what conditions are set)
- such words occur in the text that are frequent when "play" is in the heading, etc. (keyword and hierarchical co-occurrence based search, Method 2)
- "play" itself, or some other keywords occur in the headings, etc., which latter are known frequently occurring together with "play" (keyword compatibility based search, Method 3)
- words occur in the texts that often co-occur with headings containing "play" or frequently co-occurring keywords (keyword compatibility and hierarchical co-occurence based search, Method 4)
- etc.

In all these approaches the main problem is common: there is no way to differentiate among the various meanings of "play", and in the approaches where the tolerance (compatibility) classes of "play" have a role, keywords like "gamble", "toy", "sport", "music", etc. will appear mixed, as "play" itself matches with all these words to some extent.

Let us assume now that the query specifies "play, card". In this case "gamble", "toy" and "sport" will still remain in the set of possible associations, but "music" will certainly disappear as playing music has no connection with playing cards whatever.

If the query is more specific and says "play, card, bridge", certainly the word "toy" is falling away as playing bridge is not a game for children playing with toys (although some simple card games might be mainly interesting for small children who also play with toys), and "gamble" is also left out of consideration as bridge is a game of cards that has nothing to do with gambling, it is rather considered to be a kind of mental sports, similar to chess or go. On the other hand, while "bridge" has several meanings, first of all denoting a construction to lead a road over a river or valley, etc., the query words "play" and "bridge" guarantee that in this query not the primary meaning but the name of a certain card game was meant. The whole example is illustrated in Fig. 2, where visible graph edges mean "strong enough" connections, i.e. co-occurrences, and not visible graph edges denote connections below some reasonable threshold. Thick edges indicate the "strongest" connections to the first

query word “play”. Apparently they do not form any tolerance class. If “card” is added, the class (play, gamble, card, poker) emerges as one possible clique. The word “roulette” is in the clique (play, gamble, roulette), but as it is not connected (strongly enough) with “card”, this clique is discarded. On the other hand, the clique (play, card, bridge, chess, go, sport) is another possible tolerance class, where in reality the edges between “bridge” and “go”, and “go “ and “sport” are certainly weaker than e.g. between “bridge” and “sport” , etc. If there is a node “Hungarian Tarot”, its connections with the set (play, bridge, card) might be stronger than any other, and then the sub-clique formed by these four words might be an even better match for the original query. Some other “dead ends” are illustrated in the figure, like “toy” is connected to words like “teddy-bear” and the like, “music” leads to other cliques containing “piano”, “violin”, etc., and have only one common node with the others in “play”, “bridge” has connections like “road”, etc., however, these are clearly distinguishable from the topic of the query as they have no more significant edges to any of the other words in the query.

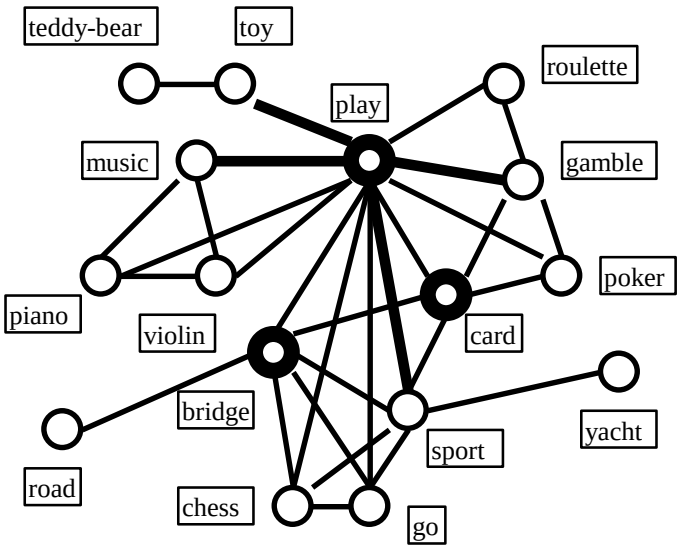

Figure 2.

4. Search by minimal cliques in the co-occurrence graph

By this example the main point in this study has been clarified. The *combination of words (keywords) that is contained in the query* has considerable information concerning the context of each individual word. The special meanings are restricted by the other words appearing in the same query. We suggest that always the *minimal subset of minimal tolerance/ similarity classes (reflecting the maximal possible α-cut of the graph) is selected which contains all the query words*. In the previous example, if only (play, card) are in the query, then both the class C_1 = (play, card, bridge, chess, go, sport), and the other one: C_2 = (play, card, gamble, poker) will be selected and all documents will be retrieved that have strong (or any) hierarchical co-occurrence with these classes. (In the sense of Part I). If however “bridge” is added, the situation suddenly changes as C_1 is a strong clique that contains all three words in the query, but the other one contains only two of them, so including this latter would not be minimal any more.

The proposed way of determining this class or these classes is the following:

1. Locate all query words in the fuzzy co-occurrence graph *W*.
2. Determine the minimal degree of co-occurrence among these words (α).
3. Find all α-cliques in the graph that contain at least one of the query words.
4. Determine all words in *w* that have a certain minimal level of co-occurrence with the keywords (determined independently, depending on the requested width of the search, β).
5. Find all documents that contain these latter words (eventually with an occurrence degree at least γ).

In the following the procedure described above will be presented by some pictures. In Figure 3 Step 1 is illustrated. The words in the query are identified in G_W.

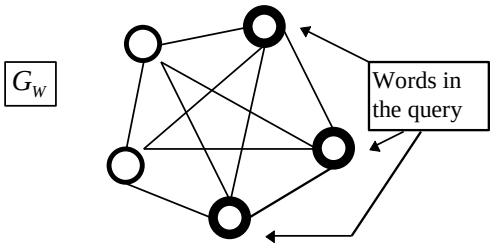

Figure 3.

The minimal co-occurrence degree among these is determined. (Fig. 4, Step 2).

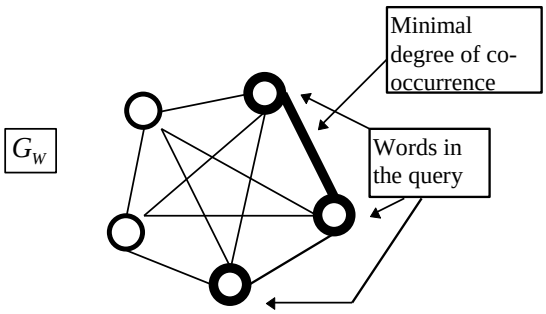

Figure 4.

All cliques with at least α strength are found. Fig. 5 shows all edges that are at least α strong, while Fig. 6 indicates all cliques (Step 3).

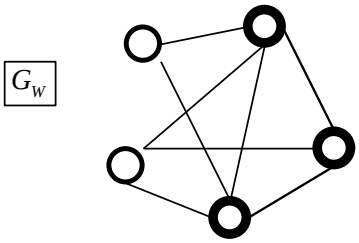

Figure 5.

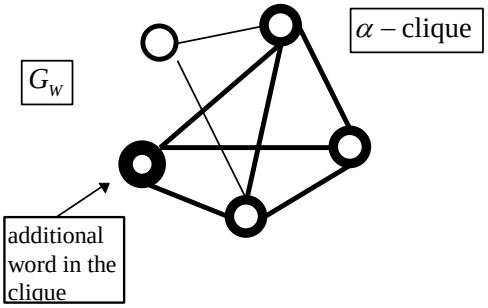

Figure 6.

From this step the search will go according to the methods described in Part I, e.g. as shown in Fig. 7.

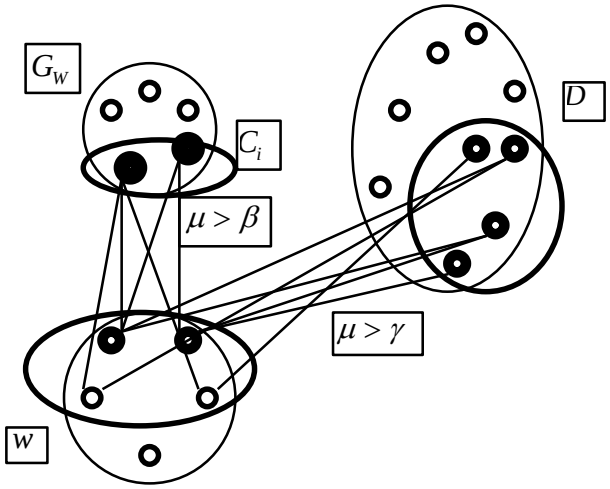

Figure 7.

5. Some concluding remarks

depending on the type of documents used it might be reasonable to break down the query into several tolerance classes. In this case instead of finding the minimal co-occurrence degree among the words in the query, this minimum must be specified independently from them, and then all cliques that have at least a degree of connectedness that is equal to this pre-specified degree, will take part in the further search.

If some of the words in the query are no keywords, first the connected (strongly co-occurring) keywords have to be found and then search will go along the above way.

If there is a co-occurrence graph known over the general words as well, then cliques must be found in both co-occurrence graphs and search has to be done for the union of these.

Further study and of related search methods and implementation of various co-occurrence and especially hierarchical co-occurrence based information retrieval techniques is currently carried out.

References

L. T. Kóczy and T. D. Gedeon: Information retrieval by fuzzy relations and hierarchical co-occurrence. Part I. Technical report IETR 97/01, Dept. of Information Engineering, School of Computer Science and Engineering, University of New South Wales, Kensington, Sydney, 1997. 18p.

Further references can be found in Part I.