GROUPING DICTIONARY SYNONYMS IN SENSE COMPONENTS

[1]ALI AWADA, [2]MAY DEHAYNI

[1]Assoc Prof., Department of Applied Mathematics, Faculty of Sciences 1, Lebanese University, Hadath, Lebanon

[2]Asstt. Prof., Department of Applied Mathematics, Faculty of Sciences 1, Lebanese University, Hadath, Lebanon

E-mail: al_awada@ul.edu.lb, maydehayni@ul.edu.lb

ABSTRACT

In this paper, a study of the synonymy in a dictionary of verbs is presented and a new approach to solve the polysemy problem is proposed. More precisely, it is about dispatching verb synonyms in groups called "sense components", each corresponding to a verb meaning. The dictionary is represented as a graph and the sense component is obtained by studying circuits in this graph. This study is based on the following idea: verbs on a circuit may/must belong to the same component of sense. Our study has resulted in a graphical interface for automatic dictionary exploitation.

Keywords: *Graph, Circuit, Dictionary, Synonymy, Polysemy, Sense Components, Thresholding.*

1 INTRODUCTION

Dictionaries constitute the cornerstone of all automatic natural language processing. Indeed, they contain formal and comparable objects, they exist in almost all languages, and the most interesting is that they carry semantic information. We start from the following reflection: if dictionary definitions are effectively carriers of sense, it is necessarily due to the network that they establish between words (dictionary entries). We suggest the use of a simple structure, able to keep enough sense for our purpose: graphs. Indeed, all dictionaries can be represented as graphs in which the vertices and the edges can be defined of multiple ways. The most intuitive definition consists in taking as vertices the dictionary entries and to admit the existence of an edge from a vertex A to a vertex B if the entry B appears in the definition of the entry A. In these graphs, it seems obvious that there are different kinds of information, and therefore of edges, as the synonymy or the antonymy link between vertices, or the hyperonymy … Therefore, the study between dictionary entries lead to studying a graph and exploiting the networks established between words.

We think that it subsist sufficiently enough information in only the topological structure of these graphs. Dictionaries nature privileges synonymy link between words. This is why that most works on dictionaries are about the synonymy link. It is about, very often, to detect components having specific properties in graph terms as cliques [12] and gangs [13] thus leading to grouping synonyms; the set of elements belonging to a same component corresponds to an "elementary sense". Awada & Chebaro have introduced the "synonymetry" concept to quantify the synonymy strength between two words [2]. This study had for goal to detect the sense components in a dictionary of verbs being based on the N-connexity as regrouping criteria and synonyms classification. In another study, Awada defines the proximity of meaning between two words as the power of synonymy between them. He proposes an algorithm to measure this proximity that uses both the synonymy and the antonymy links [1]. However, the various proposed approaches suffer from the ambiguity of natural language. This ambiguity appears in dictionaries as the presence of polysemic entries merged together as a unique node in the graph. This problem usually results of synonyms metaphorical use, the metaphorymy is a concept proposed by Duvignau & al. [3] and Gaume & al. [5].

Vol6. No1. (pp 130 - 136)

In this paper, we study the synonymy through the examination of the dictionary graph of verbs and focus on the polysemy problem for which we propose some concrete solution. We define a new grouping criteria based on the circuit concept. Also we are working on the idea that all verbs of a graph circuit are likely parts of the same family for a very precise sense that we call "sense component". Finally, we have developed a Graphical User Interface that automatically explores the dictionary and provides different synonyms distribution of a given verb in different sense components.

2 THE POLYSEMY PROBLEM

The natural language richness is reflected by the complexity of the language dictionary graph. This complexity is illustrated in turn by an increased interconnection between graph vertices. Thus, graph manipulation suffers from qualitative weaknesses related to the ambiguity induced by the polysemy problem. Indeed, the entries (and hence vertices) can lead to polysemic associations versus-nature between different senses of verbs. Indeed, a verb with many senses can act as a link between verbs representing its different acceptations. We illustrate this problem through the following example (Figure 1):

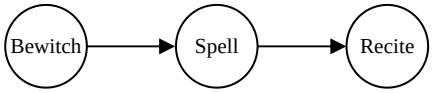

Figure 1. Association between two different sense verbs

The above example shows that *spell* is a synonym of *bewitch* and that *recite* belongs to *spell* synonym list. The path of length 2 between the verbs *bewitch* and *recite* should have meant that these two verbs have the same sense or nearly. However, it is not true since these two verbs have two completely different senses each corresponding to a meaning of the verb *spell*. This closeness between two verbs will result in the graph by the existence of two sets of verbs corresponding to *bewitch* and its direct synonyms from one side, and *recite* and its direct synonyms from the other side, *spell* is at the intersection of these two sets.

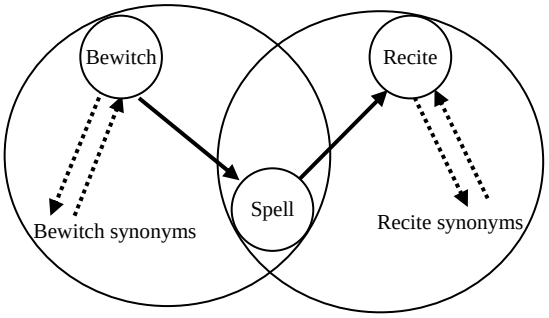

Figure 2. Polysemic verb at the intersection of two sense components

In order to solve the ambiguity problem, Victorri & Fuchs propose a dynamic construction model of sense [15]. In this model, they associate a semantic space to every polysemic unit, so that the sense of the unit in a given statement results from a dynamic interaction with the other linguistic units of the statement [4].

3 THE PROPOSED APPROACH

The dictionary presentation as a graph structure is characterized by a high number of relations (edges) between all verbs (vertices) having the same sense. This density of relations leads sometimes to the presence of circuits. Theoretically, two families of verbs having two different senses will result on the graph by two disjoined circuits. We deduce that a sense concept should correspond to a set of circuits in the graph. This assumption should be true even if there are polysemic verbs in the dictionary. Indeed, while searching circuits that start from a verb "V", a polysemous verb makes it is unlikely to return to the verb of departure "V". Therefore, the polysemous verb and all the verbs to which a path leads through the polysemous verb are obviously out of all the synonyms.

The basic idea of our work consists in grouping two verbs V_2 and V_3 synonymous of a given verb V_1 in a sense S_1 of this verb if there is at least a certain number of circuits starting from V_1, and passing at the same time through V_2 and V_3. It is well obvious that we must precisely define what we have called “a certain number of circuits”. We agree to call this number “the acceptance threshold”.

3.1 Choice of The Acceptance Threshold

Let's start by studying the effects of the variation of the acceptance threshold and its influence on the expected results:

As we have mentioned, the acceptance threshold acts as the filter that will prevent to add some synonyms of a given verb to the same sense component, and will therefore allow, in contrary, grouping others. A weak value of the acceptance threshold would enter in the same sense component verbs that have few or not enough relations between them as synonyms of the examined verb because of few circuits joining them.

On the other hand, a high value of the acceptance threshold would prohibit grouping verbs that have the same meaning. In some cases, it may lead to eliminate some verbs that would be considered as non acceptable synonyms of the examined verb.

Let's illustrate our purpose through the example of the figure 3:

Assume that the examined verb is « *to keep* » and the number of circuits containing the verbs *to keep*, *to preserve* and *to protect* is N_1. We have depicted the verbs joining *preserve* to *protect* by X. The number of circuits N_1 is obtained by adding the number of circuits N_2 passing through the verb *prevent*, and the number of circuits N_3 passing through any other verb Y (each of the two symbols X and Y represents several verbs).

Assume that the acceptance threshold is less than N1. This has the effect of classifying *preserve* and *protect* in the same sense component. Concerning *prevent*, two cases are to be taken into consideration:

- N_2 is greater than the acceptance threshold $\Rightarrow$ *prevent* is in the same sense component of *preserve* and *protect*.
- N_2 is less than the acceptance threshold $\Rightarrow$ *prevent* is not part of the previously mentioned component. Two cases are mentioned:
  - *Prevent* will belong to another component (that does not appear in the figure).
  - *Prevent* will not appear in any other component and therefore it will not be considered as a synonym of *keep.*

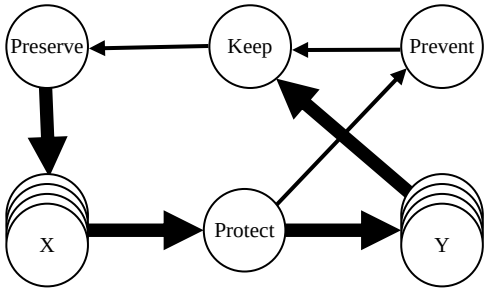

Figure 3. Graph containing multiple circuits (heavy arrows depict the set of paths between a verb and a group of verbs whereas a simple arrow depicts a set of paths between two verbs)

This simple case illustrates the difficulty of choosing the acceptance threshold value. For this reason we have minimized the role of the acceptance threshold by associating it to another factor: the circuit length.

3.2 Importance of The Circuit Length

One of the important factors that ensures the existence of significant synonymy between two words is the distance (in terms of graph) between them, and therefore of the connecting circuit length. Indeed, the richness of a language results in a complex dictionary. More senses are associated to a verb and more edges are connected to the corresponding vertex in the graph.

On the other hand, polysemic verbs in one or several circuits may leads to errors in the classification process. Indeed, as depicted below in figure 4, a circuit that starts in V1 can be constituted of two paths: one path starting in V_1 toward a polysemic verb V_2 having a sense S_1, and the other starting in V_2 toward V_1 and having another sense S_2.

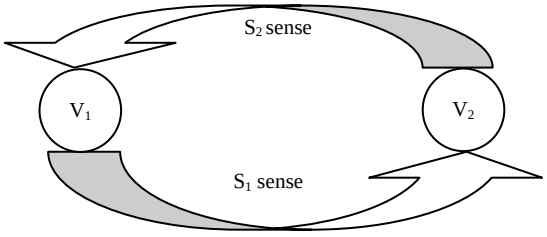

Figure 4. Circuit involving a polysemic verb

The polysemy is an inherent property to all languages, and thus it can not be eliminated. However, we will try to minimize its harmful effects by decreasing the length of the circuis to be processed, and therefore restricting the number of verbs that belong to. Indeed, in long circuits, there

Vol6. No1. (pp 130 - 136)

are more verbs, and therefore more chance to find polysemic verbs that may lead to mixing different sense components. However, taking into account only short circuits would lead to split a same sense component into several ones. Thus a compromise must be found to achieve an equitable solution. Also, we reformulate our grouping principle as follow:

Two verbs V_2 and V_3 synonyms of a given verb V_1 must be grouped in a sense component S_1 of V_1 if it exist at least a certain number of circuits of length less than or equal to a given length starting at V_1,and passing through V_2 and V_3 at the same time.

Thereafter, we will name the previously mentioned maximal length "the limit length".

4 THE SENSE COMPONENT CONSTRUCTION PROCESS

We have elaborated three methods of verb synonyms grouping. All these approaches use a matrix constructed from statistics on the dictionary's graph circuits.

4.1 The Common Circuit Matrix

In order to study a verb V, a preliminary step consists in building the common circuit matrix. When the list of circuits is generated, we construct a symmetrical square matrix (therefore a triangle) whose entries are the synonyms of V and where the cell content of coordinates (S_i, S_j) corresponds to the number of circuits starting from the verb and containing S_i and S_j. This matrix is of a primordial utility because it allows synthesizing the different relations between synonyms in pairs. It is obvious that it is necessary to study this matrix and to extract the pairs of synonyms having a significant relation. This is accomplished by comparing the content of each cell with the previously mentioned acceptance threshold.

Let's remind that the retained circuits are those having length less than or equal to the limit length fixed at the beginning. This limit length greatly influences on the results because a low value would eliminate indirect synonyms but would give a high number of sense components because a sense will be associated to small groups of verbs. Otherwise, a raised value has the advantage of decreasing the number of components but would include indirect synonyms away from the examined verb. Therefore it is necessary to take into account a compromise on circuits limit length. The software tests we have done show that the most appropriate limit length is about 5 edges.

This matrix constitutes the basis material on which we will process to group synonyms having the same sense in the same component.

4.2 Elaboration of The Potential Meaning Groups

In order to obtain the set of groups, we start by building groups each containing two elements. We try to extend these groups as follows:

A significant relation *R* exists between two synonyms S_i and S_j if their corresponding values in raw i and column j in the matix is greater than the acceptance threshold. These two synonyms form a pair as shown in figure 5.

	S_1	S_2	S_3	…	S_m
S_1	-	15	4		14
S_2	15	-	9		3
S_3	4	9	-		6
…					
S_m	14	3	6		-

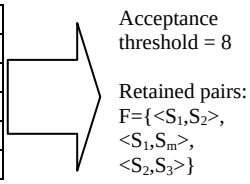

Figure 5. Example of synonyms pairs obtained from the matrix of commun circuits

Once the set of pairs E is developed, we transform E in a set of triplets while trying to include a significant synonym (corresponding to the same sense of the two verbs in the pair), then a set of quadruplets … Finally, when E becomes stable, it contains the potential sense components. We introduce later three grouping methods to extend E. These three methods use the following conventions:

E: the set of acceptable synonym pairs obtained from the matrix.

Ej: the j^{th} group of F.

n = the number of groups in E, then E = {E_1, E_2, … , E_n}.

S_k = the k^{th} synonym of the examined verb V.

m = number of verbs (synonyms) in the matrix.

4.2.1 Grouping by circuits extension

Principle

Consider a group of verbs {S_i,S_{i+1}, …, S_j}, we integrate the verb S_k into this group only if S_k maintains a meaningful relation simultaneously with all elements of the group. This means that the number of circuits that conatin S_k and all elements of this group is greater than the acceptance threshold.

Algorithm

```
do
{
  stability = True;
  For j = 1 to n
     For k = 1 to m
        If (Sk ∉ Ej)
           If the number of circuits
           containing Sk && all Ej elements
           is > acceptance threshold
           {
              Ej ← Ej ∪ Sk;
              stability = False;
           }
} while (stability == False);
```

Disadvantage

Recall that our goal is to obtain the sense components; we realized that the condition to include a given verb into a component is too tough to be checked by verbs that have the same meaning as the component one. Also we have abandoned this method to propose an improvement.

4.2.2 Grouping by separated associations

Principle

Consider a group of verbs $\{S_i,S_{i+1}, \ldots, S_j\}$, we integrate the verb S_k into this group only if S_k maintains a meaningful relation separately with every element of this group. This means that the matrix values of (S_i,S_k), (S_{i+1},S_k),…, (S_j,S_k) are all greater than the acceptance threshold.

Algorithm

```
do
{
  stability = True;
  For j = 1 to n
     For k = 1 to m
        If (Sk ∉ Ej)
           If there is a relation R
           between Sk && each Ej element
           {
              Ej ← Ej ∪ {Sk};
              stability = False ;
           }
} While (stability == False);
```

Evaluation

Although this second method is more flexible than the first one, it remains too constraining and still seems prone to amelioration.

4.2.3 Minimal constraint grouping

Principle

Consider a group of n verbs $\{S_i,S_{i+1}, \ldots, S_j\}$, we itegrate the verb S_k into this group only if there is a circuit of length n+1 containing the elements of the group and S_k.

Algorithm

```
Do
{
  stability = True;
  For j = 1 to n
     For k = 1 to m
        If (Sk ∉ Ej)
           If it exists a circuit that
           contains only Sk && all Ej
           elements
           {
              Ej ← Ej ∪ {Sk};
           stability = False;
           }
} While (stability == False);
```

Evaluation

Each element of E must correspond to a group of synonyms having a specific sense. However, this solution has a certain number of lacks. Indeed, the traversal direction of the synonymous verbs influences the result. Let's take the example of a group $E_i = \{S_1, S_2, \ldots, S_i\}$ and two candidates S_p and S_q and there is a circuit of length i+1 containing $S_1,S_2, \ldots, S_i$ and S_p and another circuit of length i+1 containing $S_1,S_2, \ldots, S_i$ and S_q but there is no circuit of length i+2 containing $S_1,S_2, \ldots, S_i, S_p$ and S_q. The verb to integrate the group E_i is the first examined, the other won't ever be added. This makes that the number of groups obtained remains greater than the number of meanings to retain. So, we envisage a merging step that consists in gathering the groups of same sense into the same sense component. We mention that we have adopted the third method that gave better results than the first two methods. We have developed our dictionary browser using this method.

4.3 Merging Potential Groups In Sense Components

The grouping yields a set E of groups E_i each containing verbs having the same sense. However, it may occur that two groups correspond to a same sense. A merging of such two groups seems necessary to get a sense component. We recall that there is equivalence between meaning and sense component.

4.3.1 Principle

Two groups E_i and E_j ($card(E_i) = n_i$; $card(E_j) = n_j$ with $n_j \leq n_i$) must be merged if:

- E_i contains $(n_j - 1)$ verbs of E_j.
- There is a relation (to specify) between the verb w of Ej not belonging to E_i and the verbs of E_i not belonging to E_j.

Indeed, E_i contains (n_i - n_j) verbs which are not in E_j. Let Q the set of these verbs and q = card(Q).
We have considered the study of different possibilities of relation between w and a certain number of elements of Q. We have established that forcing w to have a direct relation (edge) with each element of Q doesn't allow reducing the intermediate senses in a significant way and would let close senses groups not to be merged. After a thoroughly study of the problem, we reached the following conclusion: in order to include w in E_i, it is sufficient that there is an edge between w and one of the Q elements.

4.3.2 Algorithm

```
do
{
  stop = True;
  For i = 1 to n
     For j = 1 to n
        If (Ei ≠ Ej)
        {
           ni = card(Ei);
           nj = card(Ej);
           If (nj > ni) /* Ei smaller than
           Ej */
              Swap Ei and Ej ;
           If (card (Ei ∩ Ej) >= nj-1)
           {
              w = Ej \ (Ei ∩ Ej);
              Q = Ei \ (Ei ∩ Ej);
              If there is a relation R
              between w and an element of
              Q
              {
                 Ei ← Ei ∪ Ej;
                 stop = False ;
                 delete Ej; /*merge Ei and
                 Ej */
              }
           }
        }
} While (stop == False);
```

5 RESULTS

The study presented in this paper resulted in a software of dictionary browsing. This software allows the user to enter a verb to examine and returns its different sense components according to all previously mentioned steps. The obtained results allow us to affirm that a same sense component rarely contains verbs having different senses. However, a same sense can commonly be found in two different sense components. Thus every component corresponds to a degree of the examined verb meaning. As an example, the verb French verb garder (to keep) has the following four sens components:

- <préserver (to preserve), épargner (to spare), éviter (to avoid), sauver (to save), garantir (to secure), protéger (to protect), conserver (to conserve)>,
- <conserver (to conserve), maintenir (to maintain), préserver (to preserve)>,
- <conserver (to conserve), maintenir (to maintain), retenir (to hold) >,
- <retenir (to retain), éviter (to avoid), empêcher (to prevent) >.

We notice that the well provided component is the one corresponding to the most currently used meaning of the examined verb. Otherwise, a same verb can be found in two different components designating each a shade of meaning.
The example of the French verb peser (to weigh) illustrates clearly this property. The software associates the following sens components:

- <examiner (to examine), juger (to judge), considérer (to consider), apprécier (to appreciate), étudier (to study), réfléchir (to think), calculer (to calculate), approfondir (to deepen), estimer (to estimate)>,
- <importuner (to importune), presser (to press), harceler (to harass)>,
- <importuner (to importune), fatiguer (to tire), ennuyer (to bother)>.

6 CONCLUSION

In this paper, we attempt to realize a tool for dictionary automatic browsing that allows extracting the sense components associated to a given verb based on circuits in the graph of dictionary.

The developed interface offers the user the possibility to parameterize his research. Thus, he can choose the value of the acceptance threshold and the circuits limit length. Well obviously, it would require the user to be an expert as well as in computer science than in linguistics. This motivated us to envisage a solution in which the acceptance threshold is automatically calculated from the common circuit matrix. This solution doesn't require any expertise from the user. However, the computed value doesn't always produce the best results because the density of arcs in the graph is not uniform.

The obtained results seem encouraging and often correspond to the different meanings of the examined verb. However, the concept of sense is complex enough and ambiguous in linguistics and

some shades of sense seem very difficult to surround.

Otherwise, the construction of the dictionary that we have used lays a few problems since the point is put on the verbs while omitting the other grammatical categories. It seems obvious that verbs like *to do* or *to take* do not carry enough sense without the noun phrase that follows them. Therefore, the verb *to take* becomes polysemic and will be considered as a synonym to *to happen* (take place) as well as to *to snap* (take a picture). We assumed the radical solution that consists in eliminating this kind of verbs from our study in order to minimize the errors resulting from their use.

Finally, we estimate that the acceptance threshold choice is crucial and that a particular attention must be lent to it. Indeed, the results are closely bound to the acceptance threshold value and therefore depend greatly on it. It seems primordial to find a robust method that determines the optimal acceptance threshold value according to the examined verb and its connections. A statistical study of the acceptance threshold variation and its effects on the results are foreseeable.

REFERENCES:

[1] A. AWADA, « An approach to classify synonyms in a dictionary of verbs », - *International Conference on Applied Computing, IADIS-AC 2005,* Algrave, Portugal, pp. 317-322.

[2] A. AWADA, B. CHEBARO, « Etude de la synonymie par l'extraction de composantes N-connexes dans les graphes de dictionnaires », *Journées d'études linguistiques JEL 2004,* Nantes, France, pp. 245-249.

[3] K. DUVIGNAU, C. FABRE, F. FERRATY, O. GASQUET, B. GAUME, B. JOUVE, J. LANG, M.P. PERY-WOODLEY, « Les dictionnaires de langue: des graphes aux propriétés topologico-sémantiques », *Etats Généraux du Programme de REcherches en Sciences COgnitives de Toulouse (PRESCOT),* Toulouse, France, 2000.

[4] J. FRANÇOIS, J.L. MANGUIN, B. VICTORRI, « La réduction de la polysémie adjectivale en cotexte nominal: une méthode de sémantique calculatoire », *Cahier du Crisco no 14,* Université de Caen, France, septembre 2003.

[5] B. GAUME, K. DUVIGNAU, O. GASQUET, M.D. GINESTE, « Forms of Meaning, Meanings of Forms", *Journal of Experiment and Theoretical Artificial Intelligence,* vol. (14/1), 2002, pp. 61-74.

[6] L. GOSSELIN, « Le traitement de la polysémie contextuelle dans le calcul sémantique », *Intellectica 1996/1, 22,* pp. 93-117.

[7] D.Z. INKPEN, G. HIRST, "Automatic sense disambiguation of the near-synonyms in a dictionary entry", *Proceedings of the 4th Conference on Intelligent Text Processing and Computational Linguistics (CICLing-2003).*

[8] B. LE BLANC, D. DION, D. AUBER, G. MÉLANÇON, « Constitution et visualisation de deux réseaux d'associations verbales », *2nd Colloque sur Agents Logiciels, Coopération, Apprentissage et Activité humaine (ALCAA),* 2001, pp. 37-43.

[9] C.B. LE LOUPY, « Evaluation des taux de synonymie et de polysémie dans un texte », *Conférence TALN 2002,* Nancy, France.

[10] J. LYONS, « Sémantique linguistique », *Ed Larousse,* Paris, France, 1990.

[11] J.L. MANGUIN, B. VICTORRI, « Représentation géométrique d'un paradigme lexical », *Conférence TALN 1999,* Cargèse.

[12] S. PLOUX, B. VICTORRI, « Construction d'espaces sémantiques à l'aide de dictionnaires de synonymes », *TAL,* 39, n°1, 1998, pp. 161-182.

[13] F. VENANT, « Géométriser le sens », *Les Journées Graphes, Réseaux et Modélisation, ESPCI,* Paris, France, 2003.

[14] F. VENANT F., « Polysémie et calcul du sens », *7e Journées internationales d'Analyse statistique des Données Textuelles JADT 2004,* pp. 1145-1156.

[15] B. VICTORRI, C. FUCHS, « La polysémie, construction dynamique du sens ». *Ed. Hermès,* Paris, France, 1996.