From Term Variants to Research Topics

Fidelia Ibekwe-SanJuan * and Eric SanJuan **

* ERSICO – Université Lyon 3. 4 cours Albert Thomas, 69008 Lyon.
ibekwe@univ-lyon3.fr
** LAPCS – Université Claude Bernard, 50 av. Tony Garnier, 69366 Lyon Cedex 07.
eric.sanjuan@univ-lyon1.f

Fidelia Ibekwe-SanJuan obtained her BA at the University of Port-Harcourt (Nigeria) and her PhD at the Stendhal University in Grenoble (France). She is currently a lecturer in information science at the Jean Moulin University, Lyon (France). Her research interests revolve around term variant extraction, automatic text clustering, textmining, computational linguistics and knowledge organization.

Eric SanJuan obtained his PhD in mathematics and computer science at the Lyon I University (France) in December 2000. He is currently a lecturer in computer science at the Metz's University Institute of Technology (France). He is also a member of the Data Mining group of the theoretical and applied computer science laboratory (LITA) of Metz, and the organizer of the international conference JIM'2003 on Knowledge Discovery and Discrete Mathematics.

F. Ibekwe-SanJuan, E. SanJuan (2002). **From Term Variants to Research Topics.** *Knowledge Organization, 29(3/4).* 181-197. 21 refs.

ABSTRACT: In a scientific and technological watch (STW) task, an expert user needs to survey the evolution of research topics in his area of specialisation in order to detect interesting changes. The majority of methods proposing evaluation metrics (bibliometrics and scientometrics studies) for STW rely solely on statistical data analysis methods (co-citation analysis, co-word analysis). Such methods usually work on structured databases where the units of analysis (words, keywords) are already attributed to documents by human indexers. The advent of huge amounts of unstructured textual data has rendered necessary the integration of natural language processing (NLP) techniques to first extract meaningful units from texts. We propose a method for STW which is NLP-oriented. The method not only analyses texts linguistically in order to extract terms from them, but also uses linguistic relations (syntactic variations) as the basis for clustering. Terms and variation relations are formalised as weighted di-graphs which the clustering algorithm, CPCL (Classification by Preferential Clustered Link) will seek to reduce in order to produces classes. These classes ideally represent the research topics present in the corpus. The results of the classification are subjected to validation by an expert in STW.

KEYWORDS: Textmining, Hierarchical clustering, Scientific and technological watch, Term extraction, Terminological variation.

1. Introduction

Thematic trends survey is an aspect of scientific and technological watch (STW) that focuses on textual data analysis. As such, the goal pursued is identical to that of bibliometric and scientometric studies. However, the data analysis methods used in these studies base clustering solely on statistical criteria such as occurrence or co-occurrence of the units considered; for instance, co-word analysis (Callon , Courtial, & Turner, 1991) clusters of already existing keywords co-occurring in bibliographic index; and co-citation analysis (Small, Sweeney, Greenlee, 1985) and author co-citation analysis (White & Mccain 1989) cluster of respective references or authors cited together in a third document. These methods use factual informa-

tion (the presence of two authors or two books in the reference section of documents) to generate a co-citation map which reveals scientific collaborations amongst researchers, but they do not deal with textual data *per se*. Although some researchers have integrated linguistic processing of textual data in order to extract noun phrases (Warnesson,Coupet, Gouttas, & Huot, 1995) or term variants (Polanco, Grivel, & Royauté, 1995 ; Lelu, 1993), clustering is still based on statistical criteria (co-occurrence of count units). Statistical data analysis models favour very frequently occurring units whereas in STW, rarely occurring units can be of particular interest. The novelty of our system lies in the fact that clustering is based on meaningful linguistic relations; here syntactic variations can highlight information carried by low-frequency units in a corpus. Also, our method uses the expressive graph formalism to represent the objects clustered (terms and the relations between them).

With the advent of huge amounts of unstructured textual data, the focus has lately been on building textmining tools that enable expert users to gain useful knowledge rapidly without having to read the texts. Following the definition of datamining, textmining can be considered as the discovery of "*nontrivial, implicit, previously unknown and potentially useful information*"[1] from unstructured textual data. As such, the objectives pursued by textmining systems are somewhat akin to our STW in that it seeks to identify interesting trends in huge collections of texts that can be useful for various purposes.

Feldman et al. (1998) built textmining tools to identify important trends from text collections. In their systems, the user is required to encode some kind of domain knowledge before they perform textmining tasks. The user has to build a concept hierarchy, a sort of taxonomy of domain concepts. Although the user is assisted by a taxonomy editor, this task remains time consuming and the results, arbitrary. The *PatentMiner* system described in (Lent & Agrawal, 1995) also relies heavily on user participation. The latter has to define shape queries, that is, the trends he is looking for (*upward*, *slope* or *downward* distribution models), which are used to trigger the textmining tasks. Thus in these textmining systems, the user has to formulate a distribution model of co-occurring word/phrase sequences against which the textmining tool will compare the distributions observed in the text collection. It will then single out the deviant ones, deemed to be interesting. Thus, the textmining approach in these studies can be qualified as "supervised." The underlying hypothesis is that the user is knowledgeable enough about the co-occurrence distribution of individual words or phrases in his/her domain over a significant period of time (years, months) to be able to formulate meaningful distribution models. Also, it implies that the user knows what s/he is looking for before s/he finds it. Such a hypothesis would appear too strong in the face of reality.

Moreover, the text collections used in some textmining studies need not be coherent or thematically oriented. The sources can be varied: e-mails, newspaper articles, database texts, and so forth. In most textmining methods, authors were not particularly concerned with the linguistic value of the units extracted. Lent and Agrawal (1995) used mainly simple words as count units and relied on statistical indices like co-occurrence to indicate the association strength of two words. Although Feldman et al. (1998) extracted terms as count units, no clustering is performed. What the system basically does is to choose those co-occurring sequences whose distribution deviate from a user-specified model. Moreover, in these systems, users' queries have to be specified in a formal language such as the Boolean or the SQL languages. Results are presented as a ranking of highly co-occurring textual units. The notion of individual documents is quite strong here; the co-occurrence of units is with regard to their mutual appearance in individual documents.

In a STW task, the user's initial input into the system is quite minimal if not absent. Normally, the user's information need is defined as a natural language query (NLQ). The elaboration of a formal Boolean query needed for building the corpus is left to an information retrieval (IR) specialist. The corpus built is therefore thematically coherent and corresponds to technical and scientific publications published in the renowned journals of the field. Thus, the academic value of texts analysed in a STW endeavour has to be maximal considering the use to which the results are put.

In a STW endeavour, the information units extracted need to be clustered in order to reduce the data. This application relies on data analysis methods such as hierarchical clustering, principal component analysis, correspondence analysis or latent semantic analysis. What the expert user seeks is a graphic and global view of research topics in his/her field. The idea is to discover the structure of research topics such as contained in the texts. The results should enable the user to increase his/her competitive intelli-

gence. Thus STW is interested in capturing topic emergence, growth, shifts and obsolescence. STW is also to be differentiated from text categorisation or summarisation endeavours where the thrust is on automatically detecting the topics of individual texts (Salton, Singhal, Buckley, & Mitra, 1996).

We developed a STW methodology that meets the above requirements but which is particularly based on linguistic knowledge. A shallow linguistic processing of text corpora is using the INTEX linguistic toolbox (Silberztein, 1993) in order to identify relevant linguistic units. These units or terms, serve as input to the clustering algorithm. As opposed to Lent and Agrawal (1995) or to Feldman et al. (1998), our approach to clustering can be qualified as unsupervised since no user model is furnished to the system. The overall system can be described by the following figure.

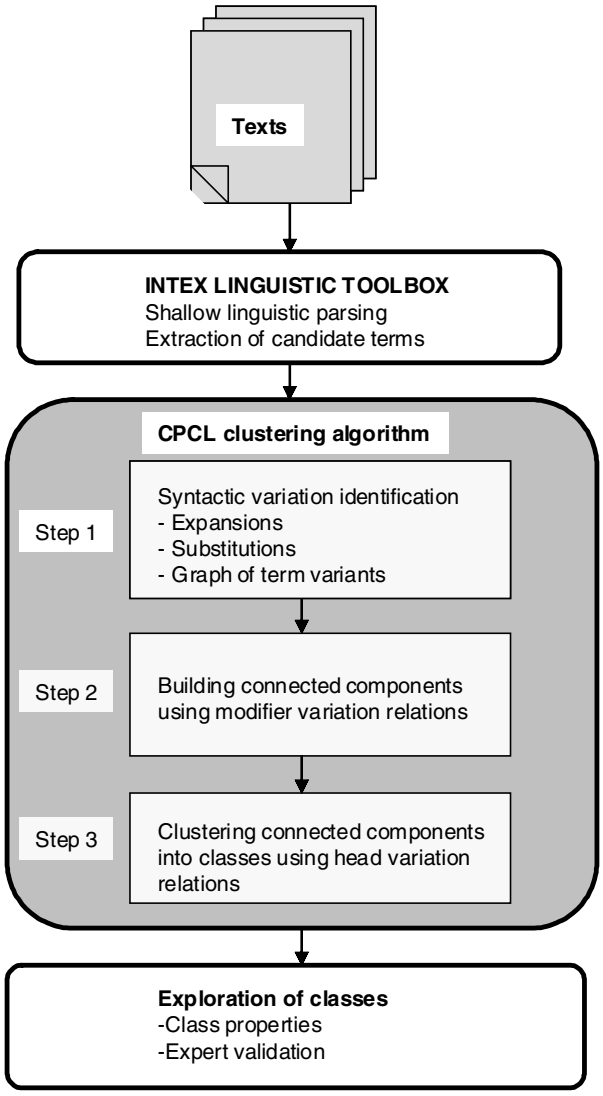

Figure 1. Overall system architecture

Different aspects of this methodology have been explored in Ibekwe-SanJuan (1998a and 1998b). First, we will focus on the automation of the term extraction module, which was previously not automated. Second, we will investigate in more detail the relevance of the different syntactic variation relations used for clustering with regard to a STW task.

To illustrate this, the method will be applied to a corpus of English scientific abstracts of 70,000 words collected on the basis of a STW request. The end user wished to know if there existed new and if possible natural additives to enhance his bread making process. The corpus was made available by the French Institute for Scientific and Technical Information (INIST).

The rest of the article is organised as follows: Section §2 describes the term extraction procedure; Section §3 presents the variation relations identified between terms; Section §4 describes the clustering algorithm (CPCL) and the properties of the classes it generates. Section (§5) is devoted to implementation issues and the validation of results by an expert.

2. Term extraction with INTEX

Although a certain number of linguistically-oriented tools exist for term extraction, we chose to define and implement term extraction rules according to the INTEX toolbox built by Silberztein (1993). While the morpho-syntactic approach adopted by Bourigault (1994) in LEXTER also tallies with our needs, this tool was first developed for the French language. ACABIT (Daille, 1994) and FASTR (Jacquemin, & Royauté 1994) are oriented to term-variant extraction in English, but need a reference terminology in order to bootstrap their systems. Such an approach can hinder the discovery of totally new terms, which are not in the reference terminology. Also, the definitions of term morpho-syntactic properties and syntactic variation did not correspond to our needs. In TERMS, Justeson & Katz, (1995) were solely concerned with term extraction in English using a rather restricted set of morpho-syntactic patterns (basically *N_N* patterns).

Our concern was to extract more or less complex terms as they appeared in the text corpus and to relate them through morpho-syntactic variations. We have not used any outside terminological resource so as not to alter the forms under which the terminological units appeared in the texts. This is more likely to portray terminological evolution and hence that of domain concepts. The implementation of the term

extraction rules was carried out in the INTEX linguistic toolbox. The choice of INTEX was supported also by its user-friendly interface to implement local grammars as graphs. INTEX provides relatively exhaustive linguistic resources in English and French, enabling the user to perform morphological, stemming and syntactic analyses. After morphological analysis on a text corpus, we defined several morpho-syntactic constraints, enabling us to identify sequences that contained potential terms. These constraints are implemented as finite state transducers with decreasing order of complexity, that is, some can be embedded in others[2]. These transducers are applied in an iterative fashion on the corpus and enable us to extract, first complex nominal sequences, which are in turn split into simpler noun phrases (NPs) until we reach the desired result. Splitting of noun phrases does not seek to extract atomic NPs (i.e. simple NPs without prepositional attachment). What we are seeking is a medium-grained splitting which allows for the extraction of complex terms that can reveal the associations between simple domain concepts. However, morpho-syntactic features alone cannot verify the terminological status of the units extracted since they can also select non- terms (Smadja, 1993). For instance, *bread characteristics* is a term in the food and agricultural sectors whereas, *book review,* also found in the corpus is not, although both have the same syntactic structure (N_N). Thus in the first stage, the terms extracted are only plausible candidates that need to be filtered in order to eliminate the most unlikely ones.

2.1 Definition of local grammars in INTEX

First we defined the structure of a minimal Noun Phrase (NP-min) as a multi-word nominal sequence without prepositional attachment. It is defined by the following regular expression:

NP-min :
*(<D> + <E>)(<NB> + <E>) <ADV> **
*<A> * <N> <N> ** [1]

where + = disjunction; E = empty string; D = determiner ; NB = number; ADV = adverb ; A = adjective ; N = noun;* = Kleene's operator indicating *n* or zero occurrences of the preceding category. This transducer will identify sequences such as :

a hydrophilic powdered lecithin [3a]
traditional sour dough starter cultures [3b]

Since one transducer can be embedded in another one in INTEX, the NP-min transducer will be embedded in other transducers describing more complex NP structures (NP-max):

NP-max :
NP-min (of +prep1+cc) (NP-min cc) NP-min*
(<E> +((of +prep1) NP-min (<E> +
(of+prep1+cc) (NP-min cc) NP-min (<E> +*
cc NP-min)))) [4]

where : *prep1* = a class of prepositions frequent in terms (from, to, by, for, with, in);
cc = coordinating elements (and, or, comma). NP-max identifies sequences such as :

a lengthy process, development of traditional bread flavour [5]
a guide to the formulation and processing of traditional yeast-containing doughnuts [6]

Other intermediary transducers will split the NP-max identified based on the presence of certain term-boundary markers (prep + determiner, coordination, presence of two prepositions in an NP-max). The examples above will be split as follows:

a lengthy process [5a]
development of traditional bread flavour [5b]
a guide [6a]
formulation [6b]
processing of traditional yeast-containing doughnuts [6c]

Because we cannot yet handle coordinating conjunctions properly, the splitting yields an erroneous sequence (6b). Splitting stops when no sequence contains either a coordination, two prepositional phrases (PP) or the sequence "prep + determiner". From our corpus, about 10, 000 candidate terms were extracted. These terms will be subjected to a filtering procedure in order to select the most likely ones.

2.2 Filtering candidate terms

Owing to the inherent structural ambiguity between an NP and a domain term, it is necessary to filter the candidate terms extracted in the preceding stage. Some previous studies (e.g., Daille, 1994) have suggested the use of various statistical filters for this task. After some empirical tests, it seems that statistical filters such as frequency or co-occurrence counts of pairs of words are inadequate to select good candidate

terms for our application, since statistical filters always favour frequently occurring items. In a STW task, the user will be looking for rare phenomena, that is, low frequency units that may point to emerging topics. For instance, half of our candidate terms appeared only once in the corpus, so they will simply be eliminated by most statistical tests. Ibekwe-SanJuan (1998a) suggested the combined use of lexical and statistical filters. The lexical filter eliminated terms beginning with a determiner other than "*the*" combined with term frequency. Indeed this determiner is the most frequent in the morpho-syntactic composition of terms. However this combined approach cannot filter out poor candidates such as "*book review*," which have the perfect term structure. We had observed in an earlier experiment that such poor candidates are naturally filtered out in later stages as they rarely possess variants and thus will not appear as research topics (see §3). Another solution is to first use a human filter (a domain specialist) and then to fine-tune the filtering based on preliminary results of the clustering. This was done for this corpus. The 10,000 candidates were first subjected to an indexer of the INIST''s PASCAL database, familiar with indexing articles in the food and agricultural sectors. The filtering was aimed at selecting only those terms that corresponded to domain concepts or objects. At this stage, one-word units were also eliminated since they do not allow for meaningful associations between terms; some 6,000 candidates were selected. After testing the clustering algorithm on these candidates, we were able to identify low content terms (too generic terms) that aggregated many other terms into the same classes, we then eliminated these terms based on this evidence. Finally, 3,651 possible terms were selected.

3. Syntactic variants identification

Tzoukermann, Klavans and Jacquemin (1997) extracted morpho-syntactic variants for applications such as automatic indexing or information retrieval. They studied a wide set of variation producing phenomena including derivational morphology and verb phrase variants, for example, *tree cutting* and *the trees have been cut down*. We focused for the moment on terms appearing as noun phrases (NPs). Although term variants can appear as verb phrases (VPs), we hypothesise that NP variants reflect more terminological stability, and more of a real shift in topic (*root hair → root hair deformation*) than their VP counterpart (*root hair → the root hair appears deformed*). Also, our application being quite sensitive, requires a careful selection of term variants depending on their interpretability. This is to avoid creating relations between terms which could mislead the end-user in his/her task. For instance how do we interpret the relation between *concept class* and *class concept*? Also, our aim is not to extract syntactic variants *per se* but to identify them in order to establish meaningful relations between them.

Given the two syntactic structures under which a term can appear - compound or syntagmatic - we first pre-processed the terms by producing the compound version of those terms that appear under a syntagmatic structure. This transformation is based on the following noun phrase formation rule for English:

$$D\ A\ M_1\ h\ p\ m\ M_2 \rightarrow D\ A\ m\ M_2\ M_1\ h \qquad [7]$$

where *D*, *A* and *M* are respectively strings of determiner, adjective and words whose place can be empty, *h* is a head noun, *m* is a word and *p* is a preposition. Thus, the compound version of "*addition of xanthan gum*" is "*xanthan gum addition*". This transformation does not modify the original structure under which a term occurred in the corpus. It only serves to furnish input data to the syntactic variation identification programs. Also, it enables us to detect variants in the two syntactic structures. We studied two families of syntactic variants: expansions and substitutions. Both types of variation can affect either the modifier words or the head word in a term.

3.1 Expansions (Exp)

Expansion is the generic name designating three elementary operations of word adjunction in an existing term. Word adjunction can occur in three positions: left, right or within. Thus we have left expansion, right expansion and insertion respectively. In the following explanations, t_1 and t_2 are terms:

- *Left expansion (L-Exp):*
  t_2 is a left-expansion of t_1 if and only if :
  t_1 = M h and t_2 = M' m' M h

- *Right expansion (R-Exp):*
  t_2 is a right-expansion of t_1 if and only if :
  t_1 = M h and t_2 = M h M' h'

- *Insertion (Ins):*
  t_2 is an insertion of t_1 if and only if :
  t_1 = M_1 m M_2 h
  t_2 = M_1 m m' M' M_2 h

Left-Expansion (L-Exp)	Right-Expansion (R-Exp)
(t1) bread manufacture →	*(t1) bakers' yeast* →
manufacture of dietetic bread	construction of bakers' yeast
manufacture of french bread	fermentation ability of bakers' yeast
manufacture of high quality bread	freezing behaviour of bakers' yeast
manufacture of moulded bread	culture of bakers' yeast growing
manufacture of plain bread	hybridization of bakers' yeast
manufacture of rye wheat bread	bakers' yeast preparation
manufacture of standard bread	sample of bakers' yeast
manufacture of whole bread	strain of bakers' yeast
manufacture of wholewheat bread	
Left-Right Expansion (LR-Exp)	**Insertion (Ins)**
(t1) glutenin subunit →	*(t1) bread improvement* →
anomalously slower migration of hmw glutenin subunit	improvement of bread flavour
apparent polymerisation of glutenin subunit	flavour improvement of bread
high mr glutenin subunit combination	improvement of bread quality
high mr glutenin subunit gene variability	improvement of bread shelf life
central repetitive domain of hmw glutenin subunit	
novel glutenin subunit composition	
wheat glutenin subunit dx	

Table 1. Sub-types of expansions

Some terms combine the two types of expansion - left and right expansions (noted as LR-Exp), for example *glutenin subunit→ apparent polymerisation of glutenin subunit*.

Examples of each sub-type of expansion are given in Table 1. The term at the beginning of each list *(t1)* has as expansion variants the terms listed below it.

3.2 Substitutions (Sub)

Substitutions mark the replacing of a component word in t_1 by another word in t_2 in terms of equal length. Only one word can be replaced at the same position to ensure the interpretability of the relation. We also distinguished between modifier and head substitution:

- *Modifier substitution (M-Sub):*
  t_2 is a substitution of t_1 if and only if:
  $t_1 = M_1\ m\ M_2\ h$ and $t_2 = M_1\ m'\ M_2\ h$
  with $m' \neq m$

- *Head substitution (H-Sub):*
  t_2 is a substitution of t_1 if and only if:
  $t_1 = M\ m\ h$ and $t_2 = M\ m\ h'$
  with $h' \neq h$

Examples of head and modifier substitutions are given in Table 2:

Modifier substitution (M-Sub)	Head substitution (H-Sub)
protein content of bread	frozen dough baking
protein content of bun	frozen dough characteristic
flour protein content	frozen dough ingredient
protein content of grain	frozen dough method
protein content of product	frozen dough product
	frozen dough treatment

Table 2. Sub-types of substitution.

Of the 3,651 terms selected, 83% were found to be involved in variation relations, thus showing the importance of the phenomena in a corpus of specialised texts.

3.3 Formal and conceptual properties of syntactic variations

The two families of syntactic variations studied yield interesting information on the formal and conceptual levels.

On the formal level, expansion engenders an ***anti-symmetrical*** or ***order relation*** between terms. If t_2 is an expansion variant of t_1, then t_1 is not an expansion of t_2. For instance *improvement of bread flavor* is an insertion variant of *bread improvement* but not vice versa. Expansions thus induce an order on the chain of variants formed. In later stages (§4.1), this results in directed arcs in the graph of variants used for clustering. For instance, *bread improvement* < *improvement of bread life* < *improvement of bread shelf life*.

Substitution on the other hand engenders a ***symmetrical relation*** between terms. If t_1 is a substitution variant of t_2, then t_2 is also a substitution variant of t_1, for example, *frozen dough baking* σ *frozen dough characteristic*. This results in non-directed arcs in the graph of variants (see §4.1).

On the conceptual level, syntactic variations yield relations which can reveal the association of concepts represented by the terms. Two major conceptual relations seem to be suggested: *class_of* and *generic/ specific*.

– *Class_of*

Substitutions (Sub) engender a relation between term variants which can be qualified as "class_of." Modifier substitution groups properties around the same concept class. In the list of modifier substitution variants above (see Table 1), the terms are gathered around the same concept class "*protein content"* which share different properties *(bun, grain, bread, flour...).* Head substitution groups concepts or objects around a class of property: *"frozen dough"* in the above example seems to be a property class shared by different concepts such as *baking, characteristics, products, treatment...* This intuition seems to be confirmed by the expert during the validation of the results (§5.1). Substitution relations do not imply a hierarchy amongst terms thus reflecting the symmetrical relation engendered on the formal level.

– *Generic / specific*

Expansions, all sub-types considered, engender generic/specific relations between terms, which echoes the anti-symmetrical relation observed on the formal level. It introduces a hierarchy amongst terms which allows us to construct quasi-paradigms. Jacquemin (1995) reported similar conceptual relations for insertion and coordination variants.

4. Automatic variants clustering

Clustering can be seen as a three step procedure. First, the CPCL algorithm builds di-graphs whose vertices are terms and are the edges of the six elementary variation relations defined above. In a second stage, it partitions the variation relations into two sub-sets, COMP and CLAS. COMP relations are usually variations that affect the modifiers in a term (L-Exp, Ins, Sub-M). These relations serve as a first stage of clustering for building connected components. The third and final stage clusters the connected components into classes using the CLAS set of relations (R-Exp, LR-Exp, Sub-C). The functioning of the algorithm is described in more details below.

4.1 CPCL: A three-tier clustering algorithm

– *Step 1. Building graphs of variants*

At this level, only one graph may be constructed if all the terminology is absolutely connected. In the present experiment, 103 sub-graphs were constructed of which one graph alone contained 88% of the term variants. A coefficient is computed for each type of variation relation. This coefficient is given as the inverse of the number of links for that variation type in all the graphs. In Figure 2 below, we have one L-Exp link, thus its coefficient is 1. We have two R-Exp links (0.50), six Ins links (0.16), nine Sub-C links (0.11) and seven Sub-M links (0.14). Because of the symmetrical nature of substitution links, they tend to be over-represented in the graphs. Our coefficient therefore introduces a normalising effect by penalising very prolific links and favouring rare links. In the example below, L-Exp link has the highest coefficient since it occurred only once in the graph. Although this coefficient is attributed to edges at the onset of the clustering process, it only comes into play during the final stage when connected components are clustered into classes (see step 3 below).

Directed arrows in the graph represent expansion links and non-directed arrows substitution links.

Viewed by portions, these graphs give forth interesting information on the terminological environment of a term. If we assume that the head word in a term represents an elementary concept in a domain and the modifiers its properties, we can put forward that this graph displays the properties shared by the same concept (see also §3.3), here *trial*, *characteristic*, *quality*, *leavening*.... This graph also enables us to follow changes in terminology. For instance, we can see where a shift in a property appears, between "*bread dough leavening*" and "*composite dough leavening*".

Notice that substitution relations are transitive on binary term variants (two-word terms) and generate complete sub-graphs as shown in Figure 3.

– *Step 2: From graphs of variants to connected components*

From the graph of all variants, the CPCL builds connected components using the modifier relations, (L-Exp, Ins, Sub-M). These connected components are sub-graphs in that there is always a link from one vertex to another. At the linguistic level, connected components represent sets of term variants sharing

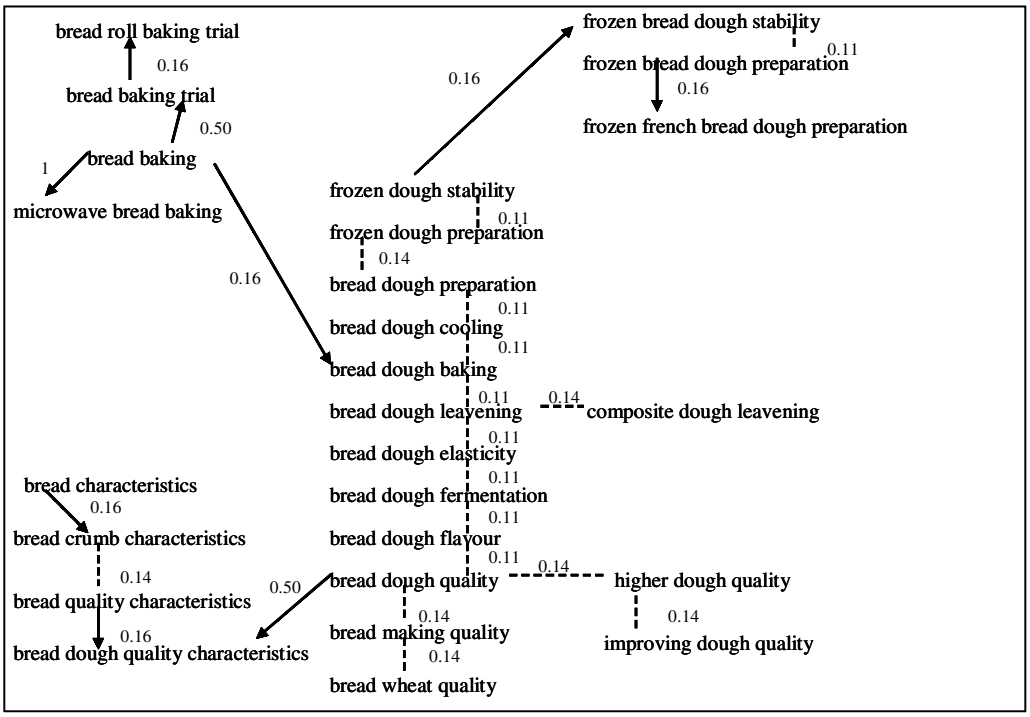

Figure 2. A portion of the weighted graph variants around "bread dough"

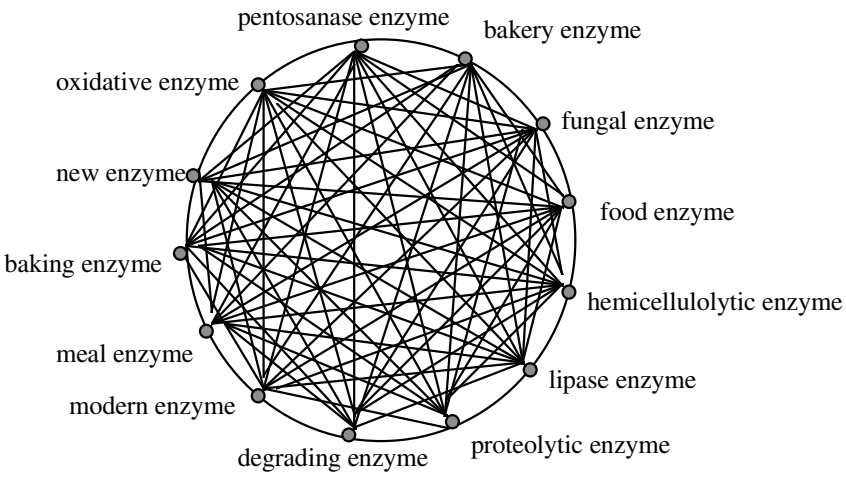

Figure 3. A complete graph formed by binary substitution variants

the same headword. The hypothesis behind using COMP relations as a first level of clustering is that they regroup term variants around the same concept families (or the same paradigm). We first obtain isolated "concept families" in the corpus. The second level of clustering will aim at highlighting the associations between these "concept families".

– *Step 3. From connected components to classes*

What the user is seeking is the identification of research topics and their interactions. Thus, connected components alone are not adequate to give this information since they depict lone concept families. A higher level clustering is necessary to identify the connected components that share preferential links. This is done by clustering them via the head variation relations (R-Exp, LR-Exp, Sub-C). The linguistic motivation being that since head variations indicate changes in head words, and thus perhaps shifts in concepts, they will be adequate to highlight research topic associations in the domain by linking together sets of terms that share the same modifiers but different heads.

Component 1	Component 2	Component 3
bakery enzyme	bakers' yeast	bakers' yeast strain
baking enzyme	Brewers' yeast	commercial baking strain
cell wall degrading enzyme	commercial bakers' yeast	commercial yeast strain
Degrading enzyme	commercial bakers' yeast	competent baking strain
export specific baking enzyme	commercial compressed yeast	danish strain
food enzyme	compressed bakers' yeast	diacetylactis strain
fungal enzyme	compressed yeast	diploid strain
grindamyl exel 16 bakery enzyme	cream yeast	fast acting yeast strain
hemicellulolytic enzyme	distiller yeast	fermenting strain
lipase enzyme	dried pellet yeast	freeze tolerant bakers' yeast strain
meal enzyme	dried yeast	freeze tolerant yeast strain
modern enzyme	dry yeast	industrial bakers' yeast strain
new enzyme	encapsulated yeast	lactis strain
oxidative enzyme	freeze dried bakers' yeast	lactobacillus brevis strain
pentosanase enzyme	frozen yeast	lactobacillus strain
proteolytic enzyme	granular yeast	mutant strain

Table 3. Example of connected components.

Two connected components are clustered when the following condition is satisfied: components c_1 and c_2 are clustered if the link between them is stronger than the link between any one of them and a third component c_3. The strength of this link is the sum of the coefficients for the head variation relations between the two components. Once clustering starts at this level, the algorithm can be iterated several times to suit the user's requirement or until it converges, that is, until classes are stable. At this final stage, the CPCL algorithm functions like any hierarchical clustering algorithm. Figure 4 illustrates this principle.

Simple lines at the term level represent COMP relations and bold lines represent CLAS relations. At level one, connected components are built using the COMP relations. At level 2, the connected components are clustered into classes based on the principle of the preferential link described above. At the first execution of the algorithm, components *c1* and *c2* are clustered since their link is stronger than the one relating them to components *c3* and *c4* (and vice versa). Thus we obtain three classes *CL1*, *CL2* and *CL3*. The first two classes are clustered again into a single class at the 1st iteration while CL3 remains alone since it shares no external link. The classes being stable at this stage, the algorithm converges.

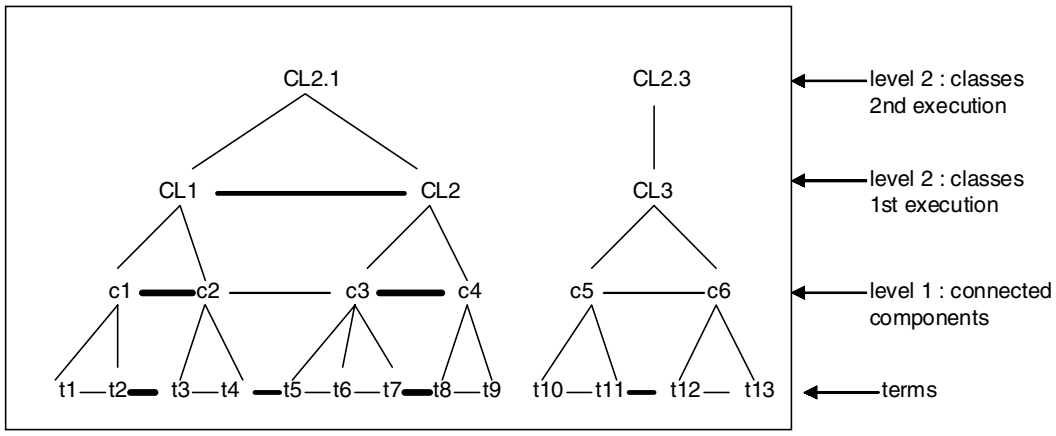

Figure 4. Principle of the CPCL algorithm

4.2 Class properties

We describe briefly here the properties that can be used to qualify classes obtained by the CPCL algorithm. They stem both from the hypothesis underlying our methodology and also the application needs. We hypothesised that variation relations would be relevant to depict research topics. In a STW task, the need to track trends is a crucial one so the properties defined have to integrate a dynamic dimension. Four

main properties were defined: class size, class centrality, variation indices and transformation index.

- *Class size*. One important feature of the CPCL method is that it does not impose a size on the classes obtained. Hence, classes vary in size and size may then indicate the proportion of a research topic in the corpus.
- *Centrality*. It characterises the number of external links outgoing from a class. It allows us to perceive the layout of classes with respect to one another. A class is deemed "central" if it has three or more external links; otherwise it is "peripheral."
- *Variation indices*. We already observed that terminological variation can be a sign that concepts represented are evolving. Hence characterising a class by its degree of terminological activity may indicate the stable or dynamic character of the underlying research topic. Two variation indices were elaborated: internal (Int_j) and external (Ext_j) variation indices. *Int_j* is calculated thus:

$$Int_j = \frac{R_j}{T_j}$$

where R_j is the sum of the coefficient of variation links in class $_j$; T_j is the total number of terms in class $_j$

Ext_j is calculated thus:

$$Ext_j = \frac{T^-_j}{T_j} \times \frac{T^+_j}{T}$$

where T^-_j is the number of terms in class j sharing variation relations with terms outside class j ; T_j is the total number of terms in class j ; T^+_j is the number of terms outside class j sharing variation relations with T^-_j ; T is the total number of terms.

- *Transformation index* relies on two parameters: the number of terms common to two classes in two different time periods[3] and the strength of variation links between the two classes. It is given by the following formula:

$$TRANS_{ij} = \frac{V_{ij}}{N^2_{ij} + 1}$$

- $TRANS_{ij}$ is the transformation index of class i towards class j; V_{ij} is the sum of the coefficient of variation relations between classes $_i$ and $_j$; N_{ij} is the number of terms common to classes i and j. Thus, the transformation index of class i for periods P_k and P_{k+1} is the average of the transformation indices $TRANS_{i.}$ of the classes in P_{k+1} with whom class i shares variation relations. $TRANS_{ij}$ will have a greater value when two classes i and j are at two different periods; P_k and P_{k+1} do not have any term in common but share a lot of variation relations.

5. Results evaluation and implementation issues

5.1 Results evaluation

By default, the classes produced by the CPCL algorithm are characterised using the properties defined above. In the present study, results evaluation focused on answering the end user's question precisely, that is, the existence of natural additives to enhance the bread making process. Let us recall that the textual data analysis task was triggered by a specific STW request. For the strict application of the class properties defined above to a corpus, we refer the reader to (Ibekwe-SanJuan, 1998b).

After iterating the CPCL algorithm on the graphs of term variants, the 2nd iteration was chosen as that which offers the best partitioning of domain classes. Thirty-three classes with variable sizes were obtained; the biggest class had 218 terms. The classes were subjected to a STW expert from the CVT[4] for validation. The choice of the expert was guided by his knowledge and experience on a similar task involving the same corpus but using a different data analysis method (François, Dubois, & Royauté, 2001). The STW expert was required to evaluate our results through the following questions:

1 – Did classes built by CPCL algorithm represent coherent domain topics? What are these topics?
2 – Did the overall structure of classes given by external links reflect the state-of-the-art of research in the field?
3 – Are there classes that can provide a more or less direct answer to the end user's STW need?

Seeking answers to these questions will also provide insight into another research concern we had: ascertain the relevancy of the variation relations studied for a STW task.

– *Topics represented by the classes*

After scrutinising the classes' contents, the expert was able to determine the topics represented by the ma-

jority of them. In most cases, the name chosen by the expert to name a class was taken directly from the class's content. Thus, terms extracted by our system were often adequate to name a class's topic. Table 5 gives the class's topic as well as the number of terms in each class.

Class	terms	Topic name
1	13	Measurement; thickness
2	8	Wheat bread**?**
3	4	Anti staling
4	5	Properties (firmness; softener)
5	4	Spring wheat or red spring wheat
6	33	Natural components or elements
7	9	Natural oil
8	6	Salt effect
9	5	"Vegetable" seed
10	5	Frozen dough
11	7	Quality of ingredients
12	10	Physical properties
13	53	Stability (fermentation; storage), dough handling
14	32	Effect of adding various substances, bread dough working**?**
15	4	Pump
16	10	Kind of flour (= meal)
17	4	Flour effect
18	16	Starter
19	42	Bread final aspect
20	90	Molecules
21	28	dough preparation methods /procedures**?**
22	218	Physical or chemical parameters influence; enzymes
23	4	Fermentation
24	16	Dough physical properties
25	12	Rheology; dough; mixture
26	5	Sulphur
27	41	Acid effect
28	17	Water influence
29	19	Bread quality
30	55	Yeast
31	11	Enzyme action
32	198	**??? (too vast, heterogeneous)**
33	12	**???**

Table 5: Topic(s) represented by each class.

Following this analysis, we can identify three categories of classes: (a) classes that represented known and relevant domain topics whose names are given; (b) classes whose topic could be partially or not at all identified (their name is followed by a **?** or by **???** respectively), and finally (c) classes whose topic though identifiable were uninteresting for scientific and technological watch. Among the relevant classes, one class (6) represented an emerging topic at the time of corpus constitution (1998).

– *The overall layout of research topics*

This is determined by representing the strength of external links between two classes. This strength is the sum of variation relations between the two classes. Classes at the centre of the network are those that have the most number of outgoing links. Figure 5 (see p. 192) shows almost all the classes to be in one network; four classes appear isolated in this figure. This may be because their external links were below the threshold we considered for clustering (0.001). Apart from class 32 whose content was deemed heterogeneous, the general layout of domain topics appeared relevant to the expert. The external links outgoing from classes 20 and 22, for example, are lexically and semantically justified. Class 22 is always linked to classes which have a meaning for the practitioner; *rheology* (25) is a physical parameter, just as *enzyme* (31) or *water* (28) are chemical parameters that influence the final product. Class 20, central in its links with classes 6, 29 and 14 seems to deal with the effect of different components/molecules on the bread quality.

In order to check the relevancy of the variation relations used for clustering, we will examine the internal and the external structures of some classes from each category.

– *Internal structure of some relevant topics*

By internal structure, we mean the image obtained by representing the variation activity within a class. This shows which terms are responsible for the class formation, that is, which types of variation links are involved in the class. Relevant and identified topics are represented in 78% of the classes produced (26 out of 33). We chose two classes (13 and 6) of variable sizes. Figures 6 and 7 (see p. 193) show the internal structure of these classes.

These figures exhibit rather connected structures. Many of the links in class 13 are initiated by anti-symmetrical relations (expansions). Class 6, which represents an emerging topic has a less interconnected structure although this could simply be due to its smaller size. A closer look at the term variants in class 6 shows its vocabulary to be relatively specific. The variants around "*wheat germ*, *wheat bran* and *bran incorporation*" appeared only in this class. This finding is supported by the external position of this class (see Figure 5). Class 6 is at the periphery of the network of classes and is linked to only two other classes (29 and 20). Note that class 6 contains term variants that can answer the STW request of the end user, that is, the existence of new natural additives in

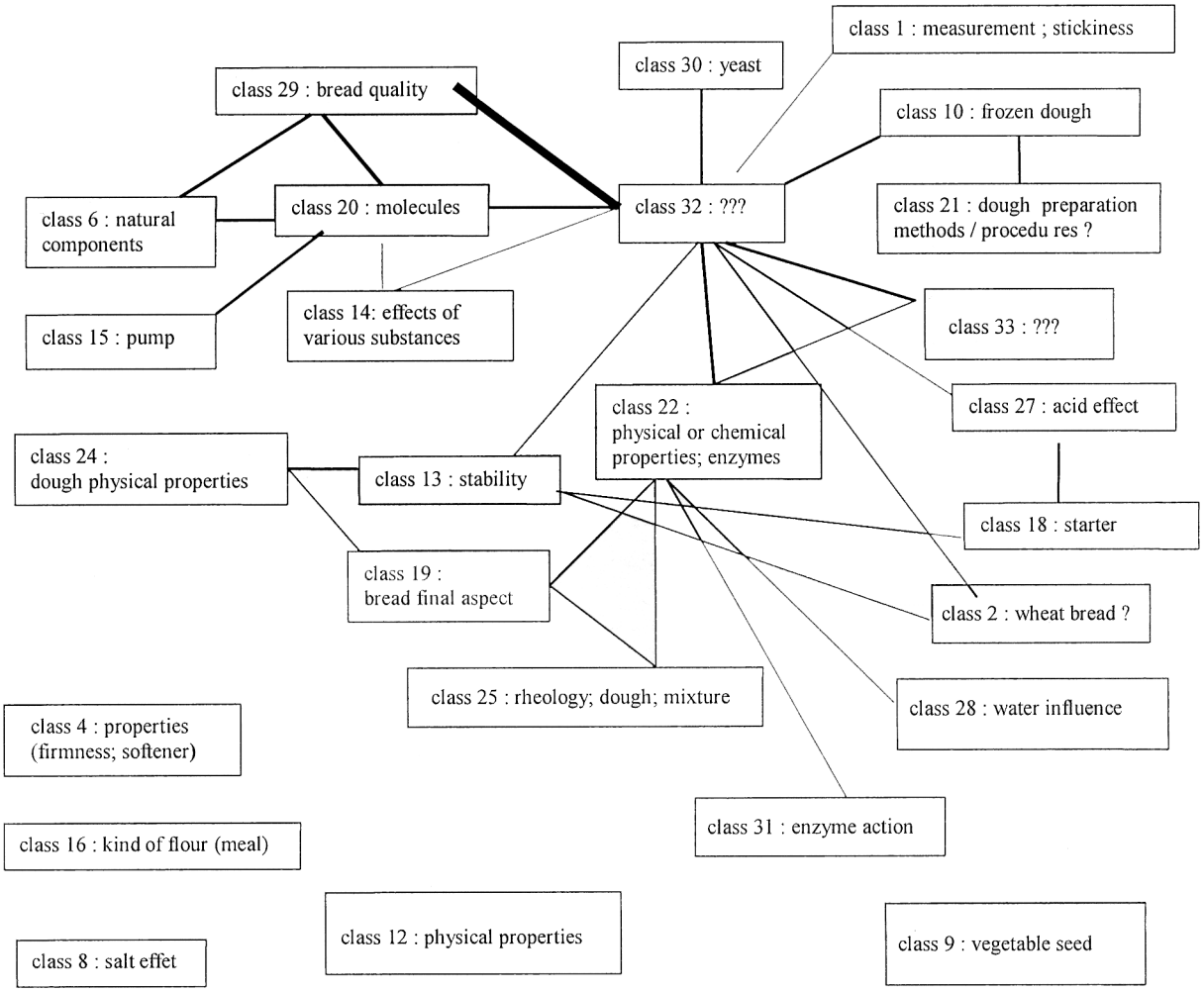

Figure 5. The external links between classes0

the bread making process. It is worth noting that the class representing an emerging topic is also the most interesting one from the point of view of the end user's STW need.

– *Structure of some partially-identified and unidentified topics*

Two of the classes observed in this category (classes 21 and 33) contained only symmetrical links (Figures 8 and 9 respectively, see pp. 193-194). It is interesting to note that all the variation relations in class 21 are made of ternary symmetrical links (three-word terms) while those in class 33 are made up of binary symmetrical links (two-word terms). Indeed, binary variants, due to their length, tend to form abundant chains of symmetrical relations whose significance is not always clear. This is the main reason why class 32 (see Table 5) is too vast. Most of its internal links were initiated by binary symmetrical relations. Moreover, these classes tend to have a wholly interconnected structure. Class 33 contained 12 terms most of which were interconnected. Finally, we note that although the topic depicted by class 21 was partially identifiable, this class is only linked to class 10 whose topic deals with "*frozen dough*" (see Figure 5). A closer look at the term variants in class 21 shows them to also revolve around this concept. Hence, the link between the two classes is lexically motivated.

– *Structure of an uninteresting topic*

Only one class is concerned in this category. Class 15 (Figure 10, see p. 194) is a very small class with only 4 terms, out of which 3 are in variation relations. These variations are all anti-symmetrical relations (expansions). We cannot however, conclude from this single example that such links indicate topic irrelevancy. Instead, it should rather be the reverse as the term variants involved here (multi-word terms and anti-symmetrical variations) normally underline shifts in either a property (modifier words) or the concept

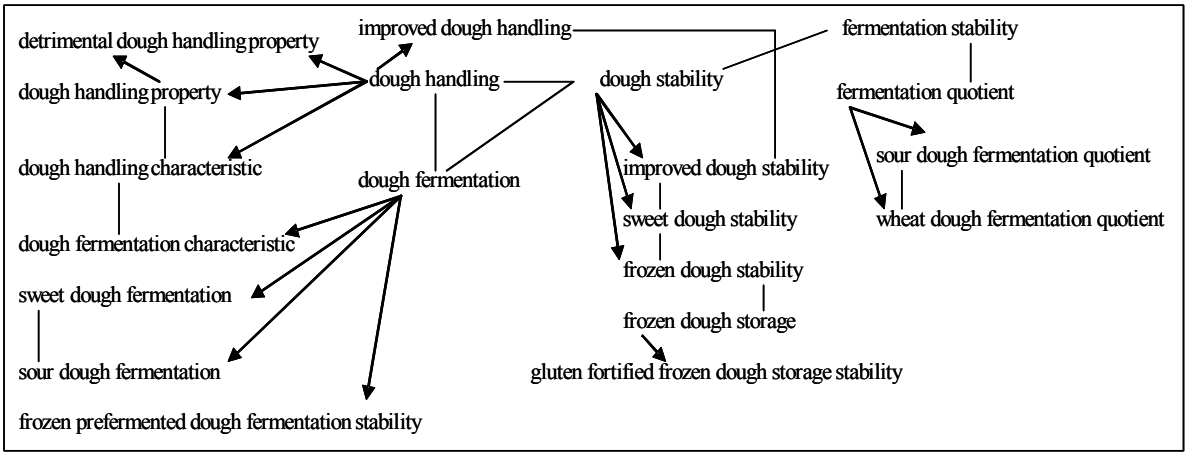

Figure 6. Internal structure of class 13 [fermentation, storage; dough handling].

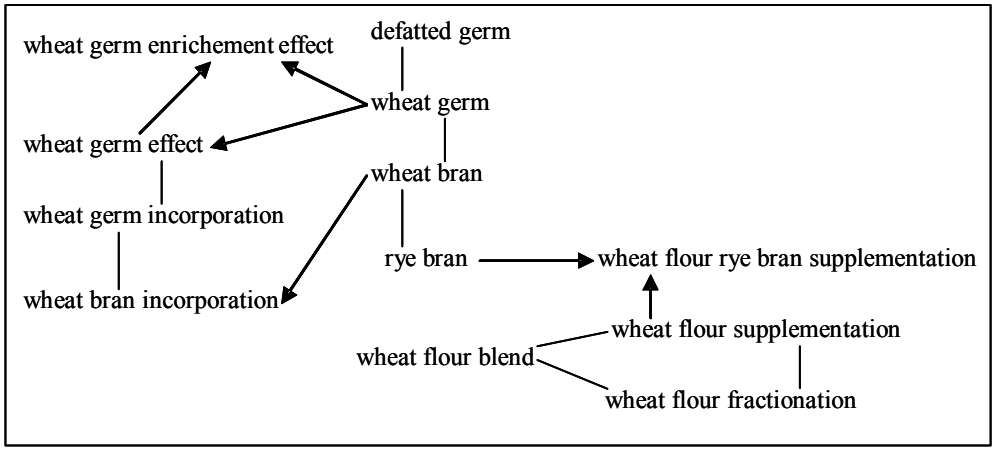

Figure 7. Internal structure of class 6 [Natural components or elements].

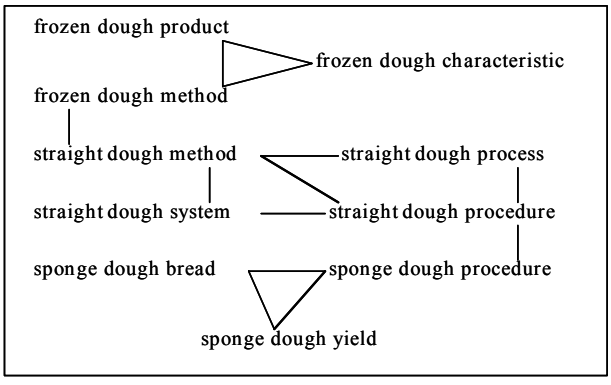

Figure 8. Internal structure of class 21 [dough preparation methods /procedures?].

(head word). Moreover, the external position of this class seems coherent with the interpretation in that it is linked to only one class (see Figure 5). We note also that the topic represented by this class was identifiable but deemed uninteresting for the task considered (STW). It may then correspond to a residuary research issue (*dough pump technology*).

– *On the relevancy of syntactic variation relations for STW*

In this paper we sought to explore whether a type of variation relation was more/less relevant for STW. Our findings, based on the few classes examined here cannot be conclusive. Nevertheless, they show quite logically that binary term variants in symmetrical re-

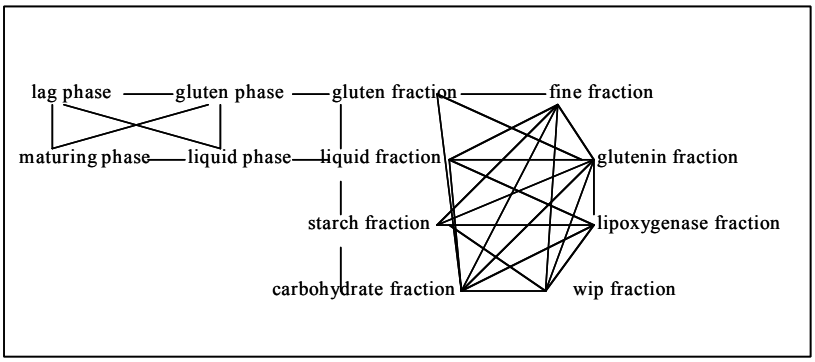

Figure 9. Internal structure of class 33 [???]

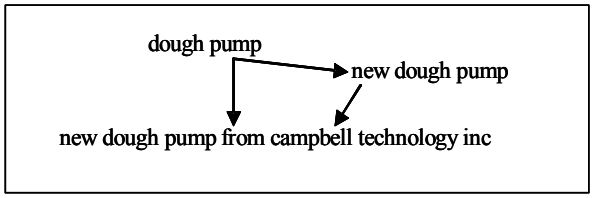

Figure 10. class 15 [Pump]

lations (head and modifier substitution) often yield large heterogeneous classes whose topics are difficult to identify (classes 32, 33 and to some extent class 22). This is because substitution as we defined it, occurs much more easily in binary than in longer terms. To deal with the problem of large heterogeneous classes, a solution may be to increase the threshold of external links at which connected components can be clustered, but we have to investigate the consequences carefully.

Despite this handicap, binary substitution variants are not proven to be irrelevant for the task at hand. On the contrary, they initiated the links between several classes as shown in the 1st column of Table 6. The number beside each term indicates the class from which it comes. We sought to know if the chain formed by these links had any meaning for the domain specialist. In the case of binary head substitution (H-Sub) variants, the expert concluded that the chain formed by binary variants highlighted the set of "concepts" that influenced a particular "object" or another domain concept, here "*dough*". Moreover, some conceptual interdependencies were observed amongst the chain formed by these binary term variants. For instance, "*improver*", "*leavening*", "*acidity*" and "*temperature*" are parameters that influence the final bread quality, that is, its "*stability*", "*weakening*", "*structure*" and "*tolerance*". Also, "*dough handling*" plays a role in bread preparation.

In the case of modifier substitution (M-Sub) variants (column two in table 6), the chain of relations highlighted the same "concept" family. This concept should be related to the *grains* (the flavour used), which will influence the *texture* of the bread. Binary substitution variants mostly account for the closely-knit network of topics on the external level (Figure 5) since they initiated most of the external links.

Binary H-Sub variants	Binary M-Sub variants
Dough handling (class 13)	bread texture (cl. 19)
Dough stability (cl. 13)	endosperm texture (cl. 19)
Dough weakening (cl. 24)	Crumb texture (cl. 19)
Dough structure (cl. 19)	good texture (cl. 19)
Dough improver (cl. 20)	grain texture (cl. 19)
Dough parameter (cl. 20)	harder texture (cl. 19)
Dough acidity (cl. 20)	loaf texture (cl. 19)
Dough temperature (cl. 20)	Softer texture (cl. 19)
Dough level (cl. 20)	
Dough leavening (cl. 20)	
Dough tolerance (cl. 20)	

Table 6. Binary substitution variants

The issue of substitution variation relevancy becomes more interesting as we consider longer terms (>2 words). For instance, given the three-word substitution variants (Table 7), it was obvious for the expert that they represented the same "property" family: "*frozen dough*" for the H-Sub variants and the same "concept" family, "*sour bread*" for the M-Sub variants (though it will be more prudent to observe more empirical results especially when the position of the substituted word changes).

Ternary H-Sub variants	Ternary M-Sub variants
frozen dough baking (cl. 10)	sour corn bread (cl. 21)
frozen dough characteristic (cl. 21)	sour dough bread (cl. 21)
frozen dough method (cl. 21)	sour maize bread (cl. 21)
frozen dough product (cl. 21)	

Table 7. Ternary substitution variants

This is consistent with our hypothesis in §3.3. From our brief survey, it would also appear that the number of internal links alone cannot determine topic relevancy. There were many interconnected classes amongst the relevant, unidentifiable and uninteresting classes.

5.2 Implementation issues

We distinguish two major components in the system we propose:

1 – Term extraction via the INTEX linguistic toolbox (Silbertzein, 1993)
2 – CPCL clustering programs

– *Term extraction with INTEX* (Silbertzein, 1993)
We implemented several automata in INTEX to extract terms from full texts described in section §2.1. We then wrote programs in the AWK language in order to normalise the INTEX outputs to suit the requirements of our clustering programs. The system has a batch queue facility to load and normalise the input file, using the lexical resources from INTEX.

– *CPCL clustering programs*
We developed a clustering system for terms based on our clustering algorithm CPCL. The entire system consists of approximately 300 commands programmed in AWK and inserted in a bash shell. This enables batch and pipeline processing and the system is used interactively. Below we now describe some of the key features of our system.

Input. The basic input to the AWK programs is a list of terms in text format. The clustering algorithm assumes that each term is associated with its compound structure (i.e., multi-word nominals). More formally, each term is associated with a string consisting of two or more words separated by special tokens, the last word being the head.

User's specifications. The system can be set up to do automatic unsupervised clustering over a corpus of terms. However, the user has the possibility of specifying the set of variation relations s/he wants to use to reduce the graph of term variants (i.e., to build connected components) and the relations s/he wants to use to apply a single-link clustering algorithm to the resulting graph (to build classes from connected components). The user can also specify the maximal level of the dendrogram s/he wants to view (i.e., the number of iterations desired).

Preprocessing. First of all, the system creates an index, which associates each word with the list of terms containing the word. Let's call this the 'word-term' index. This is done using the efficient hash tables available in standard AWK language. This preprocessing is essential to speed-up the variant identification stage.

Clustering process. Let N be the total number of terms, let l be the maximal length (number of words) of a term, and let n be the maximal number of terms containing a given word. Then using the 'word-terms' index, the number of comparisons needed to build the graph of variants (step 1, §4.1) is bounded by $O(N.l.n)$ where $l << n << N$. The graph of variants is represented using hash tables of incidence lists. There is a different table for each kind of variation but every table uses the same key. This representation allows for a fast generation of the connected components based on the relations selected by the user to reduce the graph (step 2, §4.1). We assign *1/f* points to each kind of variation, where f is the total number of pairs of terms linked in this way. To further reduce the graph, the points are summed and scaled to obtain a coefficient between components. A fast single link clustering algorithm is then carried out on this coefficient (step 3, §4.1).

Output. The output generated by the system is a relational database of terms, connected components and clusters. This database can also integrate other information like the source text, publication year or the authors of the documents from which terms were extracted. These are supplementary illustrative data which can be added to the classes to enhance their interpretation. The system automatically generates a collection of views that can be loaded into a standard spreadsheet program like Microsoft Excel or used as input to any graphics package.

Performance. We have tested our system in batch mode using lists consisting of approximately 10,000 terms. Experiments were performed on a 400 MHz Pentium PC with 256 MB of memory, running Red Hat LINUX 5.2 and on a 133 MHz Pentium Laptop with 32 MB of memory, running Microsoft Windows 98. The system took less than one minute on the PC and less than 6 minutes on the Laptop to complete the clustering over this data set.

Conclusion

The system we presented enables an expert user, in a STW context, to gain value-added information from huge textual data in order to augment his/her competitive knowledge. Our method automates some textmining tasks like linguistic analysis and extraction of relevant units (terms), structuring the units extracted by encoding some sort of relation (here the syntactic variations), and clustering these units into chunks that represent the information sought by the end user.

The interesting features of our method lie in many points. The units extracted from texts are often adequate to name a class's topic. In most data analysis methods, experts have to reformulate the class's topics as the items used for clustering (single words or keywords) were often inadequate for this task. Thus, terms are meaningful linguistic units for a STW task.

The CPCL clustering algorithm offers other advantages. It avoids the bias caused by fixing class size or number *a priori,* which ultimately leads to the artificial separation of related topics. This remains a critical aspect of most classical data analysis methods.

Also, clustering is based on an open set of linguistic relations, easy to represent with the expressive graph formalism. Thus, other higher level relations (like semantic relations) can be added. Although syntactic variations between terms give forth semantic interpretations which enhance results exploitation, the CPCL method cannot detect semantically related topics whose linguistic utterances are not in any syntactic or lexical relation, unless explicitly encoded.

Our brief survey shows that in regard to variation relations, it would seem that the internal structure of a class alone cannot indicate topic relevancy with regard to a STW task. Determining this lies within the scope of a domain expert. However, the clusters generated by our syntactic variations, often depicted coherent associations between domain topics identified as relevant. Indeed, the expert in a STW task needs formal backing to confirm his/her intuitions about the structure of research topics in his/hers field. This formal backing is usually given by results obtained from different data analysis methods.

Our future research will be channelled towards developing a user-friendly interface to access the whole system and to exploit its results. We are currently developing AWK commands that will provide a facility to generate different hypertext output formats. We also intend to investigate the means of dealing efficiently with the problem of large heterogeneous classes generated in the early iterations of the clustering algorithm. This is an inherent problem with hierarchical clustering algorithms and particularly occurs in the case of very connected graphs (reflecting very connected terminology). Another research direction will investigate the possibility of integrating IR models to enhance class content exploration, specifically in order to ascertain the research topics represented by certain classes. An IR model, such as this, tuned to exploring classes from a clustering algorithm, has been implemented by SanJuan (2002) for a similar application.

Notes

1 A derived version of the well known definition of data mining given by Frawley W.J., Piatetsky-Shapiro G, Matheus C.J., (1991). *Knowledge Discovery in Databases : An Overview.* In Piatetsky-Shapiro G, Frawley W.J. (eds). Knowledge Discovery in Databases. MIT Press : 1-27.

2 A transducer is a graph whose vertices are morpho-syntactic tags. In the simple case, this graph is equivalent to a regular expression. It becomes a "transducer" when, in INTEX, it not only recognises forms or patterns but also modifies the text.

3 A time period is obtained by partitioning the corpus such that there is approximately the same number of words in each partition. A partition spans an interval of years. This interval can be different from one partition to another. This is used for a surveying topic shifts in time.

4 Centre de Veille Technologique, CRP Henri Tudor (Center for Technological Watch) based in Luxembourg.

References

Bourigault, D. (1994). *LEXTER, un Logiciel d'Extraction Terminologique. Application à l'acquisition des Connaissances à partir de textes*. PhD. Dissertation, EHESS, Paris.

Callon, M., Courtial, J-P., & Turner, W. (1991). La méthode Leximappe : un outil pour l'analyse stratégique du développement scientifique et technique. In *Gestion de la recherche : nouveaux problèmes, nouveaux outils* (pp. 207-277). Bruxelles : VINCK, Boeck Editions.

Daille, B. (1994). Approche mixte pour l'extraction de terminologie : statistique lexicale et filters linguistiques. PhD Dissertation. Université de Paris VII.

Feldman, R., Fresko, M., Kinar, Y. *et al.* (1998). Text Mining at the term level. In J.M. Zytkow & M. Quafafou (Eds.), *Principles of Datamining and knowledge discovery. Proceedings of the 2nd European symposium PKDD'98*. (pp. 65-73). Nantes, France. Berlin-Springer.

François, C., Dubois, C., & Royauté, J. (2001). Utilisation d'un système d'analyse de l'information dans le processus de veille scientifique et technologique : pratiques collaboratives induites. *3rd Congress of the French chapter of ISKO*. Paris, 5-6 July 2001, 79-87.

Ibekwe-SanJuan, F. (1998a). Terminological variation, a means of identifying research topics from texts. *Proceedings of the Joint International Conference on Computational Linguistics (COLING-ACL'98)*, Montréal Québec, 10-14, August 1998, 564-570.

Ibekwe-SanJuan, F. (1998b). A linguistic and mathematical method for mapping thematic trends from texts. *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI'98)*, Brighton UK, 23-28 August 1998, 170-174.

Jacquemin, C. (1995). A symbolic and surgical acquisition of terms through variations. *Workshop on New Approaches to Learning for NLP, in 14th International Joint conference IJCAI'95*, Montréal,Quebec, Canada.

Jacquemin, C., & Royauté, J. (1994). Retrieving terms and their variants in a lexicalized unification-based framework. *Proceedings ACM-SIGIR 94*, Dublin, July 1994, 132-141.

Justeson, T.S., & Katz, S.M. (1995). Technical terminology : some linguistic properties and an algorithm for identification in text. *Journal of Natural Language Engineering, 1*(1), 9-27.

Lelu, A. (1993). *Modèles neuronaux pour l'analyse des données documentaires et textuelles*. PhD. Dissertation, Université Paris 6, Paris, France.

Lent, B., Agrawal, R., & Ramakrishan, S. (1997). Discovering trends in Databases. *Proceedings of the 3rd International conference on knowledge discovery in databases (KDD'97),* 227-230.

Polanco X., Grivel L., & Royauté J. (1995). How to do things with terms in informetrics : terminological variation and stabilization as science watch indicators. *Proceedings of the "5th International Conference of the International Society for Scientometrics and Informetrics", Illinois USA, 7-10 June 1995*, 435-444.

Salton, G., Singhal, A.,Buckley, C., & Mitra M. (1996). Automatic text decomposition using text segments and text themes. *Proceedings of Hypertext*, 53-65.

SanJuan, E. (2002). A Heyting algebra for modelling information retrieval based on thematic clustering. *Workshop on Discrete Mathematics and Data Mining (DM & DM'2002), in 2nd SIAM International Conference on Data Mining*, Arlington, VA, April 2002, 157-164.

Silberztein M. (1993) *Dictionnaire électronique et analyse automatique des textes. Le système INTEX*. Masson, Paris.

Smadja, F. (1993). Retrieving collocations from text : Xtract. *Computational Linguistics, 19*(1), 143-177.

Small, H., Sweeney, E., & Greenlee, E. (1985). Clustering the science citation index using co-citation. II. Mapping science. *Scientometrics*, *8* (5-6), 321-340.

Tzoukermann, E., Klavans, J., & Jacquemin C. (1997). Effective use of natural language processing techniques for automatic conflation of multiwords : The role of derivational morphology, part of speech tagging and shallow parsing. *Proceedings 20th Annual conference of ACM-SIGIR'97*, Philadelphia, PA, 148-155.

Warnesson, I., Coupet, P., Gouttas, C., & Huot C. (1995). L'analyse de dépêches de presses : Une application industrielle d'analyse de données textuelles. *Proceedings of the Congress "Les Systèmes d'Information Elaborée,"* Ile Rousse - France, 30 May - 2 June, 1995, 199-208.

White, H.D., & Mccain K.W. (1989). Bibliometrics, In M.E. Williams (Ed.), *Annual Review of Information Science and Technology* (pp. 119-186). New York: Elsevier Science Publishers.