Learning Semantic Classes for improving Email Classification

Nicolas Turenne
UMR INRA-INAPG – Biométrie et Intelligence Artificielle (BIA)
16 rue Claude Bernard 75005 Paris cedex 5
turenne@inapg.inra.fr

Abstract

In this paper we present a new term clustering method evaluated by a text classification task involving user profiles. The clustering algorithm is based on the extraction of graph patterns of terms in a training corpus. User profiles are then described by their areas of interest and the related term clusters. We demonstrate the utility of our approach in electronic mail classification. The e-mail folders represents the user areas of interest. They are described by the relevant term sets learned by clustering. A given message is routed to the relevant folder according to the distance between the terms its contains and the term clusters associated to the folder.

1 Introduction

A system that could automatically organize messages would be useful for financial and world news applications where decision making processes are based on new events and the evolution of ongoing events. Some work has been achieved in this way to process email streams [Sahami *et al.*, 1998][Kiritchenko *et al.*, 2002][Boone, 1998]. The problem is clearly related to a general classification problem [Kowalski , 1997].

Classically people use supervised learning algorithms to learn a classifier such as a decision tree or support vector machines [Joachims, 1998][Quinlan , 1993] from texts represented by keywords vectors. Such a representation does not reflect the richness of the content and word are ambiguous. Unsupervised learning can play a key role for identifying relevant keywords in the form of terms and relevant sets of semantically close words. In this paper we focuse on two kinds of knowledge capture: terms because they are more typical to a domain and semantic classes (such as nearest neighbours) because of usage similarity in context [Lewis , 1992]. There are two main ways of acquiring semantic class : general handmade resources such as WordNet and automatic classifier useful for domain analysis using windows-based or syntactical-based co-occurrence analysis [Resnik, 1992]. These approaches do not yield very convincing world classes. Lots of documents and large homogeneous corpora are required to automatically point out interesting associations with high confidence and relations in handmade dictionaries are not well adapted to mail analysis.

We present a new method for the automated construction of filters to classify messages in a user's mail stream. The core of the clustering algorithm extracts terms co-occurring with a same set of other terms and matching a relational pattern (i.e. a graphical model). The graphical model is given *a priori* such as a network of associations. Term classes are considered as discriminant according to the size of their intersection. We assume that co-occurrences of terms occurring in fragments of texts (and repeated several times) can be significant of lexical semantic and conceptual associations [Harris, 1968]. We used a window-based approach to detect co-occurrences after a syntactic tagging phase that identifies terms and "relators". A phrase (i.e. a term) meaning is not the sum of its word ones. Syntactic tagging ensures phrase identification and the identification of relational words (i.e. a relator) such as verbs. Then term co-occurrences are not just analyzed linearly ; for instance a co-occurrence between a Term A and a relator R makes a relation between A and B if the term B co-occurs with the relator R in the same sentence or a different one, with respect to the relational pattern. This could lead firstly to a natural extraction of most distinctive features and secondly to reveal thematic trends. Term sets are associated to document classes viewed as areas of interest (i.e. mail folders) by supervised learning from training data. Finally, decision making for classification is carried out by minimal overlapping of previously unseen message terms with classes of each area of interest. The method has been evaluated on various experimental sets.

Section 2 presents the related work, section 3 describes the system. Section 4 presents the evaluation framework, the experimental results and a comparison to other systems.

2 Related Work

During the past decades term classification has become unpopular in information retrieval due to lack of absolute confidence in results and because the process is a time-consuming [Grefenstette, 1996][Sparck-Jones, 1987]. A renewal of clustering techniques showed interesting results for document categorization.

Scatter/Gather [Cutting *et al.*, 1992] is an approach which regained interest for document clustering. For a docu-

ment class an averaged word vector is estimated with the closest documents.
A term clustering algorithm proposed by [Slonim *et al*., 2001] mixes notions of distribution, probability and clustering. The method called agglomerative information bottleneck is a hierarchical and better based on a probabilistic distance definition than on a general probabilistic classification strategy. The method assumes a distribution of objects from a set X (nouns for instance) according to a y variable from a set Y (verbs for instance). A mutual information measure is used to merge 2 objects according to their distributions. Minimization with a variational analysis leads to a membership probability of x knowing a class considering a Kullback-Leibler distance between the distribution of y knowing x and the distribution of y knowing the class.

Large scale data distribution has been studied to some extent. In specific tasks the user provides a set of keywords to describe his/her interests for data dissemination, for instance filtering [Yan *et al*., 1995]. Text classification, which is a well established field, but the first papers about email classification have been given after 1998. Some are based on rules others on documents description.
[Sahami *et al*., 1998] proposed training a naive Bayesian classifier on emails manually classified into spam and non-spam. Besides, they also considered domain-specific features (e.g. .edu, .com), which was shown to help to build a more accurate filter. They applied a Bayesian approach and achieved precision of 97.1% on junk and 87.7% on legitimate mail and recall of 94.3% on junk and 93.4% on legitimate mail.
[Boone, 1998] implemented an agent (Re:agent) to filter messages in 2 stages: the first one extracts features or keywords to build a vector representation (TFIDF). A rule is acquired with 2 kinds of approximation: k-nearest neighbors and neural networks with backpropagation. A test compares this approach with a vector model one (feature extraction, weighted vector creation, averaged vector estimation by folder – i.e. Rocchio approach). This method makes a kind of medium set to define a category without assumption on relation between terms. Though simple the method as the Bayesian one is quite powerful.

3 System Description

3.1 General architecture

The main components of the classification system, called Enaïm, in which a user holds the central role. The system can process English as well as French. A user can interact with Enaïm through a general graphical user interface. Three actions are available to a user:
- fetching new messages (1),
- reading messages stored in folders (2),
- profiling thematic folders (3).

The user profile is scanned by the classification module that stores a new message in a suitable folder.
The user bootstraps the profile extraction by adding or importing a quantity of messages into existing folders (an import function is included to get a formatted archive of a market email client such as Netscape, Microsoft or Eudora).
A corpus of messages (titles and bodies) is associated to each folder. An area of interest related to a folder is generated as a set of term classes extracted by automatic clustering of terms from the corpus. A user profile is a set of areas of interest.
The result of such a clustering analysis is a set, i.e. term classes, considered to characterize thematic areas of interest of the user.
Then a new message is compared to each area of interest (using classes) in turn to decide and accept the classification to the corresponding folder.

3.2 A distributional semantics approach

Several approaches of cluster extraction from texts are essentially inspired from pattern recognition. In this family of methods graph clustering fits well enough the assumption of a distributional approach since relations are processed as discrete states [Minker *et al*., 1972] [Turenne, 2000]. Our clustering algorithm regroups terms into classes representing same contextual usages of a domain. Within this corpus processing technique, we aim to extract groups such as: "layer, law, tribunal, victim", which can be subsumed clearly by a category, in this case "justice".
The algorithm takes as input pairs of positions in textual contexts in order to build classes of terms. We assume that, as in the distributional lexical theory, one could infer semantic associations from the repetition of some co-occurrence structures [Harris, 1968].
Basically a text is a sequence of words having a unique *position* (i.e. occurrence or rank of a word or occurrence). We call a *monoterm* a single word and a *multiterm* a sequence of words. A *co-occurrence* is a pair of positions. We do not use syntactic roles between terms. Our clustering is window-based or co-occurrence-based [Grefenstette, 1996]. A *window* is the sequence of words at the left and at the right of a target term. A *context* is defined by a limit: either a maximum window size set to 50 words or by the end of the current paragraph defined by a carriage return. This textual chunk reflects a topical unit.
In our algorithm the relator plays a key role. A relator (R) is a term of a given grammatical category and occurring in the same window as a target term (A). The category depends on the type of text and task. We decided to use a verb as relator category for a general applicability of any kind of email document. A schema is couple of terms (A-R). We assume that A is linked to B if they share a relator R. A contextual co-occurrence is less constrained than a syntactic association. That means for instance that most of pairs subject-verb occur only once in a corpus though we can find several times an expression (not like subject) and a verb in the same contexts. In this last case the verb will be considered as a relator. For instance in the sentence “At the hospital, the ventriculography showed an infarctus.”, it is not absurd to associate “to show” and “hospital”., even if they lack a syntac-

tic dependency. In particular we base our search of co-occurrence of relational structures on the noun phrase / verb schema (NP-V). Other kinds of schemes could be used such as names or emails.

3.3 Clustering technique

Our clustering algorithm consists in four stages [Turenne, 1999] around the building of a contingency table. This matrix gives values of binary relations between two sets of items. In our case this relation is the number of co-occurrences between a term and a relator and finally a matrix transformation states if two terms share relators. The stages are :
- Tagging terms (terms are also truncated to root form using a small sets of suffixes).
- Co-occurrence counting (ex: presence of a noun and a verb in a window).
- Aggregating terms according to a measure of similarity
- Discriminating groups according to an overlapping factor of dissimilarity.

There is a corpus example. It may describe the domain of TextMining in mailing lists and talks about natural language technologies and meetings:
P1: an overview of Human Language Technology by gathering and classifying information.
P2: an overview of Human Language Technology by gathering and classifying language resource.
P3: I am working on Human Language Technology and I have gathered a huge language resource.
P4: Euromap Event ; Details are as follows: Workshop on language resource evaluation.
P5: Euromap Event ; Details are as follows: Workshop on language engineering software distribution.

The core of the clustering algorithm extracts terms co-occurring with a same set of other terms and matching a relational pattern (i.e. a graphical model). The graphical model is given a priori such as a network of associations.
We do not cluster according to a numerical measure of similarity but with respect to relational constraints (i.e. a motif) around a centroid in a similar way as the k-means method. We have empirically observed that terms having a good rate of occurrences without being highly frequent are sufficiently representative of all significant corpus categories (example of categories for a corpus about IT: OS Platforms, Nanotechnologies, Knowledge Systems, Algorithms, Networks…). We call these terms "pole terms". We apply the definition below to choose our pole terms, hence as Figure 2 shows, we can extract triplets then and make a 4-clique using triplets centered around a same pole term (see Table 3 and 4). All triplets for each pole term are scanned. For a given triplet (P, A, B) a scan is done to search another triplet containing P and A if it is as (P, A, C) we check if the triplet (P, B, C) exists, if it is a 4-clique is stored.
Definition: let freq_max be the maximum frequency of the term set we aim to cluster. A term is a *pole term* if its frequency belongs to the interval [min*freq_max, max*freq_max] (empirically we defined min=0.1 and max=0.3).
For this corpus Table 2 associates terms (Table 1) with clusters and relators (Table 1). From paragraph P1 we deduce that "Human Language Technology" and "information" share relators "to gather" and "to classify". A link is created between them, and so on… From paragraph P5, we deduce that "Workshop" is linked to "distribution" and both P4 and P5 yield a link between "evaluation" and "distribution" sharing "to *follow*". Then "Workshop", "evaluation" and "distribution" form an association triplet.

Table 1. Sample for each clustering stages.

Terms	Pole Terms	Relators
overview	*overview*	*gather*
human language technology	*workshop*	*classify*
language resource	*information*	*be*
euromap event	*language evaluation*	*work*
workshop	*software distribution*	*have*
information		*follow*
language technology		*engineer*
language resource evaluation		
language		
software distribution		

Table 2. Number of itemsets for each stage.

#Clique3	#Clique4	#Temporary classes
65	*80*	*30*

Table 3. 3-cliques for stenosis (i.e. pole term).

[**workshop,** language resource, language technology]
[**workshop,** language evaluation, language]
[**workshop,** human technology, language]
[**workshop,** language resource, language evaluation]
[**workshop,** human technology, language resource]
[**workshop,** language resource, language]
[**workshop,** language, software distribution]
[**workshop,** language evaluation, software distribution]
[**workshop,** human technology, language technology]
[**workshop,** human technology, language evaluation]
[**workshop,** language resource, software distribution]
[**workshop,** human technology, software distribution]
[**workshop,** language technology, software distribution]
[**workshop,** language technology, language]
[**workshop,** language technology, language evaluation]

Table 4. 4-cliques for stenosis (i.e. pole term).

[**workshop,** language resource, language technology, language evaluation]
[**workshop,** language resource, language technology, human technology]
[**workshop,** language resource, language technology, language]
[**workshop,** language resource, language technology, software distribution]
[**workshop,** language evaluation, language, human technology]
[**workshop,** language evaluation, language, language resource]
[**workshop,** language evaluation, language, software distribution]
[**workshop,** language evaluation, language, language technology]

[**workshop,** human technology, language, language resource]
[**workshop,** human technology, language, software distribution]
[**workshop,** human technology, language, language technology]
[**workshop,** language resource, language evaluation, human technology]
[**workshop,** language resource,, language evaluation, software distribution]
[**workshop,** human technology, language resource, software distribution]
[**workshop,** language resource, language, software distribution]
[**workshop,** language, software distribution, language technology]
[**workshop,** language evaluation, software distribution, human technology]
[**workshop,** language evaluation, software distribution, language technology]
[**workshop,** human technology, language technology, language evaluation]
[**workshop,** human technology, language technology, software distribution]

Figure 1. Cliques of terms.

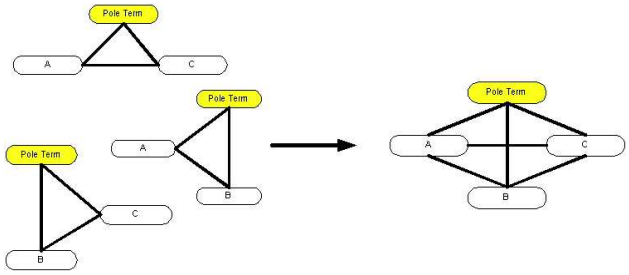
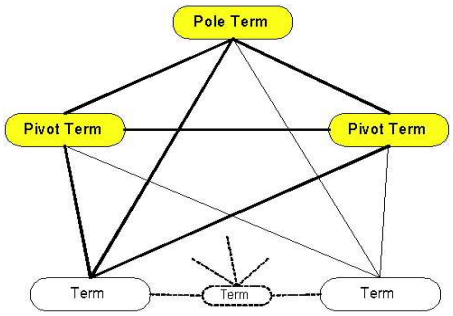


Figure 2. Cluster of terms.



So further on, we assume that a pair of 4-cliques has to share two terms, namely pivot terms, to be merged. These pivot terms behave as a classification heuristics to enrich classes without increasing the size of the class set. This merging stage does not stop till a 4-clique is found having terms matching with these pivot terms. If no pivot terms are found for the current 4-clique it is stored as temporary class. Once a 4-clique is successfully merged it does not participate anymore in another cluster (see Table 5). This operation can be viewed as the application of the pattern (i.e. a graphical model) (see Figure 2) on the contingency table. An overlapping heuristics is used in the last stage to keep only clusters having at most N terms in common with clusters that have been already built. In general N is set to 1 or 2 (see Figure 3).

Typical properties of clusters ensure that all terms can not be systematically clustered (i.e. incomplete classification) and a given term can belong to several clusters (i.e. multi-classification). Internal stability of a cluster is sensitive to deleted or new links : the deletion of a pivot term or a pole term destroys the cluster and the deletion of a normal term keeps the cluster without the term.

Once we have obtained a series of 4-cliques for each term pole, we try to cluster them further still having the same pole term. At this time links are already created and we can only make hypotheses about graph configuration.

Table 5. Aggregated 4-cliques for all pole terms.

[**workshop,** language resource, language technology, language resource evaluation, human language technology, language, software distribution]
[**workshop,** language resource evaluation, language, human language technology, language resource, software distribution, language technology]
[**workshop,** human language technology, language, language resource, software distribution, language technology]
[**workshop,** language resource, language resource evaluation, human language technology, software distribution]
[**workshop,** language resource, software distribution, human language technology, language]
[**workshop,** software distribution, language technology, language, language resource evaluation, human language technology]
[**workshop,** language resource evaluation, human language technology, software distribution,<6> language technology]

A last processing generalizes clusters and builds a hierarchical tree from overlapping clusters. A thesaurus manager structures the clusters by creating a hierarchy.

The thesaurus, defined as the Roget, contains about 120,000 forms spread over 873 categories that are themselves in 26 categories. We assign a code (an integer) to each category, so we can assign two categories to a term.

A Naive-Bayesian algorithm makes 2 semantic tagging. At the first stage we look for the codes common to 2 or more terms of a cluster [Turenne, 2002]. As terms can belong to more than one category in the thesaurus we use a Naive-Bayesian model to decide which category is more frequent among terms. In the case of a phrase we look for the code of the whole phrase or of its word components. If some codes have the same frequencies to qualify for a cluster we choose the lowest code. If such a common code exists it becomes the semantic tag of the cluster or else we look for the codes of a pole term and select the lowest code as a semantic tag. In the second stage we attach each cluster code to an upper-level node whose value is within the range of the node. We also select predominant categories by sorting all the codes representing terms of clusters in decreasing frequency order and by selecting the most frequent three codes.

These are three examples of cluster we obtained from two corpora about mailing lists, one about Windows NT technology (44 messages, 1.1 Mb) and the others about information technology research (1819 messages, 8,2 Mb):

Topic Windows NT
Hypernym: Inquiry
general purpose search engine , general public , engine strategy conference , engine optimization firm , engine strategy seminar

Topic information technology

Hypernym: News
multimodal information access, semantic web, natural language , wide information system, constraint logic, semantic web technology, web agent, web intelligence, information fusion, international workshop, genetic algorithm, basic task

Topic information technology
Hypernym: Identification
operational text classification, natural language, corpus linguistics, real application, practical application, intelligent system, web world, development activity, text classification, system knowledge, relation data, workshop proceedings

3.4 Classification strategies

We consider the classification task as a pattern matching process. We want to identify the overlap between a set of terms T provided by an unknown message and a set of term classes characterizing a user category. This means we seek the intersection between two sets of features.
Classification may be seen as a deduction process. Features of a rule are basically designed as follows :

IF "message contains ***computer***" **THEN** transfer

In Enaïm term clusters produce premise constraints for the rules. We select the best candidate folder using a score. We call a strategy a combination of a discrimination model to assign an unknown message and the configuration to compute a score. Generally it is possible to exploit the score by maximizing the value (Naïve-Bayesian model or exclusive model), or by using a threshold (Neuronal model or non-exclusive model).
If scores are equal then the message is assigned to the corresponding areas of interest. We computed the score according to 3 configurations : if terms of the unknown message belongs to a same cluster, to clusters having a same pole term or to the whole sets of clusters. Finally we tested developed 10 possible strategies (model/configuration) (see Table 5).

If we call C an area of interest. A message M is decomposed as an ordered list of terms and the following scores S_i is computed for each C (see Table 6):

- S_1: the maximum size of the intersection of the set of terms contained in M and a class of C;
- S_2: the maximum size of the intersection of the set of terms contained in M with the union of classes of C having a same pole term;
- S_3: the number of terms contained in M and in a class of C + the number of terms contained in M equal to a pole term of a class of C;

Table 6. Classification strategies.

Strategy	Definition
1	Non exclusive score by cluster
2	Non exclusive score by area of interest
3	Non exclusive score by pole term
4	Exclusive score by cluster
5	Exclusive score by pole term
6	Exclusive score by interest centre
7	Mixed score (exclusive cluster / exclusive area of interest)
8	Mixed score (non exclusive cluster/ non exclusive area of interest)
9	Mixed score (exclusive cluster/non exclusive area of interest)
10	Mixed score (non exclusive cluster/ exclusive area of interest)

A minimal threshold having been fixed for each of the three scores, 2 ways for classification are considered:
1. assign a message to all areas of interest centers for which the score is higher than the assumed threshold (Neuronal model);
2. assign a message to all areas of interest for which the score is maximal. (Naive-Bayesian model).

4. Evaluation

4.1 Measures

We used the classical evaluation measures, *precision* and *recall*. Precision measures the proportion of relevant documents retrieved compared to noise in the list of results. Recall measures the proportion of relevant documents retrieved [Yang , 1999].
This evaluation estimates the efficiency of classification. To this end, we consider the notion of *utility* [Hull *et al.*, 1999]. This notion takes into account a mixed effect: sampling (distribution of messages) and pooling (best classified selection).
Here is our utility function for the folder c_i :

$$F_1(c_i) = A \cdot R^+ - B \cdot N^+ \qquad (1)$$

where :
A is the weight for the number of relevant documents extracted;
B is the weight for the number of non-relevant documents extracted;
R+ is the number of relevant documents transferred;
N+ is the number of non-relevant documents transferred.

This function F1 takes into account the difference between the number of relevant documents and the number of non-relevant ones filtered in the c_i folder. The measurement can depend on a folder and on the system nature dealing with various sets of documents. To reduce this measurement to a usable one for comparison, one normalizes with the utility measure related to the maximal non-relevant number of documents a user could tolerate (i.e. one choose a non-relevant number of documents s and one deduces $F1(c_i) = -B.s$). We also take into account the maximal possible relevant number of documents $M_u(c_i)$. These 2 parameters define a function called "utility function" u normalized between –1 and +1. A system will perform better as the function u reaches +1.
Then, for the folder c_i and the system S:

$$M_u(c_i) = \max(A.R+) \qquad (2)$$

$$L_u(c_i) = \min(-B.N+) \qquad (3)$$

One fixes N+ arbitrarily, for example N+ = 2 non-relevant documents tolerated. We write the scaled function:

$$u(S,c_i) = \frac{\max(Fl(c_i), L_u(c_i)) - L_u(c_i)}{M_u(c_i) - L_u(c_i)} \quad (4)$$

The baseline is the non classification level (0 filtered document). The more the system diverges from the baseline in the positive direction, the more satisfying the results.
Reduced scaled F1:

$$u^*(S,c_i) = u(S,c_i) - u(Baseline,c_i) \quad (5)$$

The global result is the average on the overall set of p folders:

$$u^*_{tot} = \frac{1}{p}\sum_{i=1}^{p} u^*(S,c_i) \quad (6)$$

R+ et N+ ensure the estimation of F1 given (12), we then estimate u* from (16) expression. We reduce u* by subtracting u*(baseline, R+=0 and N+=0).
We try to estimate the rate of covering (*typicality* or τ) of each term belonging to the profile. A profile is composed with one or several interest centre(s) (category or theme). Each area of interest C_i is a set of term classes. We can calculate the membership rate τ of a given term T from the profile having n_p interest centers.

$$\tau = \frac{\sum_{i=0}^{n_p}\delta_i}{n_p} \text{ where } \delta_i = \begin{cases} 1 \text{ if } T \in C_i \\ 0 \text{ else} \end{cases} \quad (7)$$

In the ideal case each area of interest is discriminant and tends to capture exclusively its terms : so a term belongs to only one area of interest and τ is $(n_p)^{-1}$.

Hereafter, we define a status (classifiability) according the language (set of words) describing a document to enlighten the problem of recall.
Definition. We assume a language L checking a set of constraints C, and a document D described with a set $(w_i)_n$ of words. L describes a set of categories in which D is supposed to belong. D is called *non-classifiable* for the set of categories if $w_i \notin L$, $\forall i$.

4.2 Databases

Email classification is a kind of document categorization so we decided to test Enaïm on well known benchmark databases. One is Reuters and the other is the Newsgroups package. But we also made our hand-made corpus from personal email data.

Reuters
The Reuters package contains 21,578 documents [Hull *et al.*, 1999]. Training/test information and categorical information were included in each message as a tag/value format. 7,800 messages were dispatched into 115 categories. Among them, 10 were large enough for learning purpose and 2,548 test messages were parsed in the classification process since their category was among the available corpora.

20-Newsgroups
The 20-Newgroups package contains 19,020 documents. Training/test information and categorical is included in each document. Documents come from 20 newsgroups stated as categories among with 7,647 are labelled as test messages.

Mailing Lists
Our purpose is to filter emails and we built our own corpus made with 3 categories. This corpus contains 1804 messages among with 511 are labelled as test messages. Table 6 shows information about the data. They are composed with emails coming from an account subscribed to several mailing lists.

Table 7. characteristics of the mailing lists database.

Category	Averaged Message size (in words)	Test	Training
IT research	550	389	1143
Windows NT	9000	8	10
Religion	1300	114	140

4.3 Experiments

Preliminary results
Our experiments are based on patterns (clusters of terms) composed by words or groups of words. The method intrinsically can depends on the model on which the groups are extracted. It is not the purpose of the paper to analyze how methods of terms extraction can influence the results. We make only the hypothesis that terms composed with several words can help the classification process.
We firstly made an experiment with term clusters only composed with multiterms. The runs on the 20-newsgroups package gives no results hence P (precision) and R (recall) are null. In this way documents are not classifiable.We tried to take into account lots of monoterms but they are much more frequent than multiterms and responsible of a large amount of co-occurrence relations. So this configuration though interesting is not tractable and so has not been tested. Finally the input file of the clustering algorithm is made such that it contains mainly multiterms and monoterms that are not widely distributed (1 % of the number of paragraphs) in the given corpus.

Table 8. u* values for Enaim and Trec-top ranked systems.

	[Hull *et al.*, 1999]	Strategy 7
Enaim		***0.17***
Microsoft /Sheffield	0.005	
Irit	0.05	
Claritech	0.05	
NTT	0.1	
ATT	0.1	
Queen's College	0.15	

We ran within the previous configuration of term extraction and using the Reuters package. In classification strategy 1 the precision was about 30% for the two biggest categories (*acq* and *earn*) and recall about 50-70%. On average, its value is near values obtained during Trec-7 and Trec-8 filtering tracks. Table 8 summarizes the averaged values.
The typicality parameter is a good indicator to measure the restricted vocabulary of a category compared to others. According to the set of clusters it seems that many terms belong to more than 30% of interest centres. On the 3809 terms (monoterms and multiterms) contained in the profile, 1558 belongs to more than 2 interest centers and 770 belongs to more than 3 interest centers. It influences a lot assignment of a message to a category and tends to drastically decreases the precision.
The influence of multiterms on recall can be understood by working out non classifiable test messages. Among the acq and earn categories 240 documents are non classifiable using strategy 7. Some of them are composed only by a title with an averaged size of 11 words. The majority is composed with more than 1 multiterm. We decided to decompose multiterms into monoterms and to assign a document with a naïve bayes approach using a threshold to decide if a document can be assign or not to a category. Table 9 shows the number of rightly classified messages and the number of rightly classified message added to ones rightly classified using a consensus rate of 40% (i.e. at least 40% of words in the message belonging to the targeted category).

Table 9. Overwiew of Classification with Term Clusters

	Total # Test Files	# Rightly Classified Files Enaim	# Rightly Classified Files Enaim + Simple Words membership
earn	1088	537	923
acq	719	509	688
Total	1807	1046	1611

Figure 4. Consensus recall and precision.

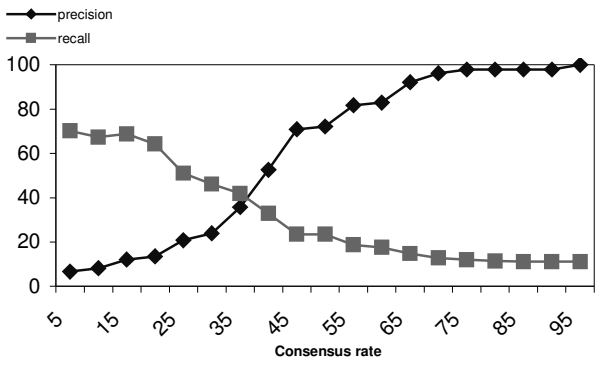


Figure 4 plots recall and precision related to the variation of the consensus rate to categorize a message. A consensus over 35% improves the recall but as the terminology of categories overlaps some messages for the same rate of consensus can be dispatched in another category it belongs.
The discussion on these preliminary results lead to the following empirical observations : only multiterms into clusters give no good recall considering short messages. So clusters have to contain monoterms. A naïve Bayesian assignment with the help of monoterms reveals an improvement of recall but a decrease of precision. Typicality has to be maximized to avoid an overlapping of simple words to reduce bias of precision. So interest areas do not have to share terms to achieve a good recall.

Final results
A final run considering conclusions from preliminary experiments and the strategy 7 gives very efficient results for our last experiment. We set a configuration with : decomposition of multiterms into monoterms in the incoming message, term clustering with a maximized typicality (no term clustered crossing interest areas). Output achieves a precision of 0.94 +/- 0.06 and a recall of 0.92 +/- 0.02. Table 10 summarizes confusion matrices about categories.
In the mailing list collection 9% of unknown have less than 100 terms though they represents 87% of the 20-Newsgroups collection. Table 10 shows results of the runs on the mailing collection. The windows Nt archive has been processed differently from the others. There are few documents in the collection but documents are large enough to gather lots of monoterms also spread out in other archives. For this archive the run gives good results if clusters contains only multiterms.

Table 10. confusion matrices for the mailing list package.

WindowsNT	Rightly class.	Badly class.	total
legitimate	8	0	8
non legitimate	1	502	503
total	9	502	511

Religion	Rightly class.	Badly class.	total
legitimate	103	11	114
non legitimate	15	382	397
total	118	393	511

Research	Rightly class.	Badly class.	total
legitimate	358	31	389
non legitimate	8	114	122
total	366	145	511

An experiment with 20-newsgroup does not show interesting results maximizing typicality with multiterms. Recall is 33 % and precision 58 %. [Slonim *et al*., 2001] had obtained an accuracy of 44% using words-clusters. Contrary to the preliminary experiments we had to match a monoterm contained into a multiterm to process a test message else a relevant test message could not be correctly classified in its topic. That means that typicality has to be considered in the same way. This may increase the recall significantly. In the current for instance on the 3200 test messages (40%) has been unclassified and 0

test messages (on 251) has been rightly classified in talk.religion.misc.

5. Conclusion

This paper presents a new algorithm for term clustering based on graph patterns and applied to a classification task. The clustering and classification algorithms have been implemented in a system called Enaïm. Term clustering characterizes areas of interest such as the topics of archives of small size messages. The clustering algorithm is based on graphs of relations between co-occurring terms. Hence a matching between term clusters and the terms of previously unseen messages is implemented in our classification module in order to test the strength and limits of the clustering algorithm. We have benchmarked Enaïm with the Reuters and 20-Newsgroups collections for tuning the configuration of clusters extraction. We have finally run the algorithm over a personal set of texts from mailing lists. Several classification strategies have been compared. The best one combines a maximization of cluster membership and whole cluster set membership of test message terms. Classification accuracy gives 94% of precision and 92% of recall. We pointed out from our experiments that some parts of the methodology can influence final results:

- a strong influence of the language modeling, i.e. how sequences of words are considered (monoterms only, multiterms only, mixed sets).
- the way terms are compared (i.e. between a monoterm and a multiterm).
- terms extraction methodology regards to the size of document (less or more than 100 terms)
- and regards to the kind of language (narrative or descriptive).

We demonstrated that co-occurrences extracted from documents taking into account shared relations (patterns) can be efficient in a specific task such as document classification. The evaluation shows that the use of such data structures can not be self-sufficient and depends strongly on preliminary language assumptions.

Acknowledgments

The author thanks Claire Nédellec (INRA-BIA) and Daniel Memmi (CNRS-IMAG) for their useful comments.

References

[Boone, 1998] Boone G., "Concept Features in Re :agent, an Intelligent Email Agent", Conference on Autonomous Agents, 1998.

[Cutting *et al*., 1992] Cutting D., Karger D., et al. "Scatter/Gather : A Cluster-based Approach to Browsing Large Document Collections", in the proceedings of the Special Interest Group on Information Retrieval Conference (ACM-SIGIR), 1992.

[Grefenstette, 1996] Grefenstette G., "Evaluation Techniques for Automatic Semantic Extraction : Comparing Syntactic and Window Based Approaches", in Corpus processing for Lexical Acquisition ed. B.Boguraev, J Pustejovsky MIT 1996

[Harris, 1968] Harris Z., Mathematical Structure of Language, ed. Wiley, 1968.

[Hull *et al*., 1999] Hull D. & Robertson S. "The TREC-8 Filtering Track Final Report" in the proceedings of the Text Retrieval Conference (TREC), 1999.

[Joachims, 1998] Joachims T., "Text Categorization with Support Vector Machines: Learning with Many Relevant Features", *Proceedings of the European Conference on Machine Learning*, (Chemnitz, Germany), 1998.

[Kiritchenko *et al*., 2002] Kiritchenko S. & Matwin S. "Email Classification with Co-training", *CASCON'02 (IBM Centre for Advanced Studies Conference)*, Toronto, 2002.

[Kowalski, 1997] Kowalski G., Information Retrieval Systems. Theory and Implementation, ed. Kluwer, 1997.

[Lewis, 1992] Lewis D., "An Evaluation of Phrasal and Clustered Representations on a Text Categorization Task", in the proceedings of the Special Interest Group on Information Retrieval Conference (ACM-SIGIR), Copenhagen (Denmark), 1992.

[Minker *et al*., 1972] Minker J., Wilson G. and Zimmerman B., "An Evaluation of Query Expansion by Addition of clustered Terms for a Document Retrieval System", Information, Storage and Retrieval,Vol(8), 1972.

[Slonim *et al*., 2001] Slonim N, Tishby N. "The power of word clusters for text classification". *23rd European Colloquium on Information Retrieval Research (ECIR)*, (Darmstadt, Germany), 2001.

[Quinlan, 1993] Quinlan, J. R., C4.5: Programs for Machine Learning, ed. Morgan Kaufmann, San Mateo, California, 1993.

[Resnik, 1992] Resnik P., "WordNet and Distriubtional Analysis. A class-based Approach to lexical Discovery." , AAAI Workshop "statistically-based NLP techniques, 1992

[Sahami *et al*., 1998] Sahami M., Dumais S., Heckerman D., Horvitz E. "A Bayesian Approach to Filtering Junk E-Mail", in proceedings of the AAAI Symposium 1998.

[Sparck-Jones, 1987] Sparck-Jones K. "Synonymy and semantic classification", ed. Edinburgh University Press, 1987 [1967]

[Turenne, 1999] Turenne N., "Learning of a pre-structured set of concepts : the GALEX tool", *Mathématiques, Informatique et Sciences Humaines*, Vol(148), 41-71, ISSN 0995-2314 1999.

[Turenne, 2000] Turenne N., "Statistical Learning for Concept Extraction from Texts. Application to textual Information Filtering", PhD thesis, Louis-Pasteur University, Strasbourg, 2000.

[Turenne, 2002] Turenne N. "Bayesian Discriminant Analysis for Lexical Semantic Tagging", *16th Internation European Meeting on Cybernetics and Systems Research (EMCSR)*, Vienna (Austria), 2002

[Yan *et al*., 1995] Yan T. , Garcia-Molina H. "SIFT - a tool for wide-area information dissemination". *USENIX Technical Conference*, (New Orleans, Louisiana), 1995.

[Yang , 1999] Yang Y., "An Evaluation of Statistical Approaches to Text Categorization", *Journal of Information Retrieval*, Vol 1, No. 1/2, pp 67--88, 1999