SVETLAN'
A System to Classify Nouns in Context

Gaël de Chalendar[1] and **Brigitte Grau**[1,2]

Abstract. Using semantic knowledge in NLP applications always improves their competence. Broad lexicons have been developed, but there are few resources made for non-specialized domains which contain semantic information available for words. In order to build such a base, we conceived a system, SVETLAN', able to learn categories of nouns from texts, whatever their domain. In order to avoid general classes mixing all the meanings of words, they are learned taking into account the contextual use of words.

1 INTRODUCTION

Using semantic knowledge in NLP applications always improves their competence as in Information Retrieval or Word Sense Disambiguation systems. Broad lexicons have been developed, but there are few existing resources which contain semantic information available for words that are not specialized to very specific domains apart from WordNet [1]. Moreover, manual or automatic processes that build semantic categories of nouns usually lead to define general categories. For example, words in WordNet are related to a Synset when they are synonymous, however Synsets correspond to large categories, and there are some shifts of meaning so that when two words belonging to a same Synset are considered within a specific context, they often no longer share a common meaning. Automatic processes that extract knowledge from texts by using statistical [2] or distributional [3], [4] approaches also lead to build broad classes, if they are not applied to specialized texts belonging to a very specific domain. On the other hand, we do not want to learn a general ontology, whatever the domain is. As most words are polysemous, we claim that a semantic base has to deal with all the meanings of a word, by associating them with their context of interpretation. Having such a semantic knowledge will allow information retrieval and Question/Answering systems for example to use deeper semantic analysis of texts, even if applied on database that contain texts on different domains which are non technical articles and uses a general and common vocabulary such as newspaper articles bases.

In order to build such a base, we conceived a system, SVETLAN', able to learn categories of nouns in context from texts, whatever their domain. It is based on a distributional approach: nouns playing the same syntactic role with a verb in sentences related to the same topic, i.e. the same domain, are aggregated in the same class. SVETLAN' relies on knowledge about semantic domains automatically learned by SEGAPSITH [5].

2 OVERVIEW OF THE SYSTEM

Input data of SVETLAN' (see Fig. 1) are semantic domains with the Thematic Units (TUs) that have given birth to them. Domains are sets of weighted words, relevant to represent a same specific topic. They are automatically learned by aggregating similar thematic units, made of sets of words. Each TU corresponds to a part of text that is homogeneous from a topic point of view and is delimited from a text by a topic segmentation process relying on lexical cohesion. Processed texts are newspaper articles that are pre-treated in order to retain only lemmatized content words.

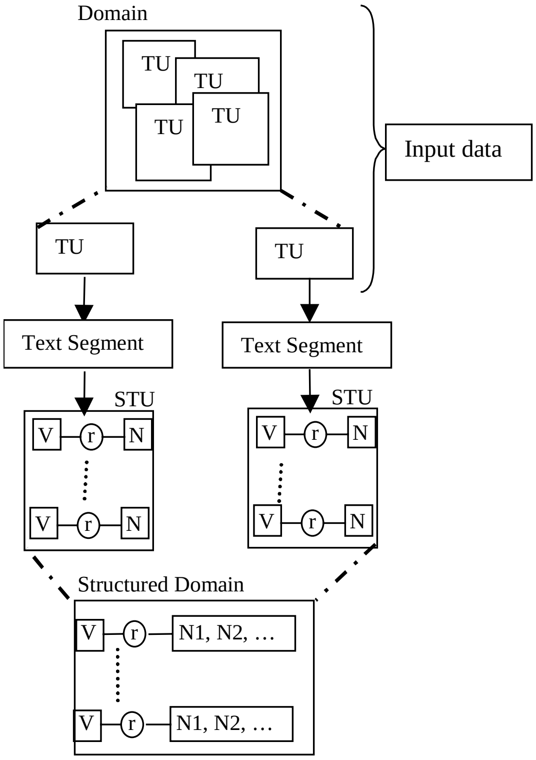

Figure 1. Schemata of Structured Domain learning

The first step of SVETLAN' consists of retrieving text segments of the original texts associated to the different TUs in order to parse their sentences. We extract then all the triplets constituted by

[1] LIMSI/CNRS, BP 133, 91 403 Orsay Cédex, France, email: {gael,grau}@limsi.fr
[2] IIE-CNAM, 18 allée J. Rostand, 91 000 Evry, France

a verb, the head noun of a phrase and its syntactic role from the parser results in order to produce the Syntactic Thematic Units (STUs). The STUs belonging to a same semantic domain are aggregated altogether to learn a Structured Domain. Aggregation leads to group nouns playing the same syntactic roles with a verb in order to form classes. As these aggregations are made within TUs belonging to a same domain, classes are context sensitive, which ensures a better homogeneity. A filtering step, based on the weights of the words in their domain allows the system to eliminate nouns from classes when they are not very relevant in this context.

3 SEMANTIC DOMAIN LEARNING

We only give here a brief overview of the semantic domain learning module. This one is described more precisely in [5]. This module incrementally builds topic representations, made of weighted words, from discourse segments delimited by SEGCOHLEX [6]. It works without any *a priori* classification or hand-coded pieces of knowledge. Processed texts are typically newspaper articles coming from *Le Monde* or *AFP (Agence France Presse)*. They are pre-processed to only keep their lemmatized content words (adjectives, single or compound nouns and verbs).

The topic segmentation implemented by SEGCOHLEX is based on a large collocation network, built from 24 months of *Le Monde* newspaper, where a link between two words aims at capturing semantic and pragmatic relations between them. The strength of such a link is evaluated by the mutual information between its two words. The segmentation process relies on these links for computing a cohesion value for each position of a text. It assumes that a discourse segment is a part of text whose words refer to the same topic, that is, words are strongly linked to each other in the collocation network and yield a high cohesion value. On the contrary, low cohesion values indicate topics shifts. After delimiting segments by an automatic analysis of the cohesion graph, only highly cohesive segments, named Thematic Units (TUs), are kept to learn topic representations. This segmentation method entails a text to be decomposed in small thematic units, whose size is equivalent to a paragraph. Discourse segments, even related to the same topic, often develop different points of view. To enrich the particular description given by a text, we add to TUs those words of the collocation network that are particularly linked to the words found in the corresponding segment.

words	occ.	weight
examining judge	58	0.501
police custody	50	0.442
public property	46	0.428
charging	49	0.421
to imprison	45	0.417
court of criminal appeal	47	0.412
receiving stolen goods	42	0.397
to presume	45	0.382
criminal investigation department	42	0.381
fraud	42	0.381

Figure 2. The most representative words of a domain about justice

Learning a complete description of a topic consists of merging all successive points of view, i.e. similar TUs, into a single memorized thematic unit, called a semantic domain. Each aggregation of a new TU increases the system's knowledge about one topic by reinforcing recurrent words and adding new ones. Weights on words represent the importance of each word relative to the topic and is computed from the number of occurrences of these words in the TUs. This method leads SEGAPSITH to learn specific topic representations as opposed to [7] for example whose method builds general topic descriptions as for economy, sport, etc.

We have applied the learning module of SEGAPSITH on one month (May 1994) of *AFP* newswires. Figure 2 shows an example of a domain about justice that gathers 69 TUs.

As some of these domains are close and refer to the same general topic, we have applied a hierarchical classification method based on their common words to organize them in separate general topics and to structure them. Figure 3 shows the hierarchies built about sport, police and stock exchange. Each leaf is a domain, named by its two more weighted words, while internal nodes are described by their name and their size, i.e. the number of common words found in their children.

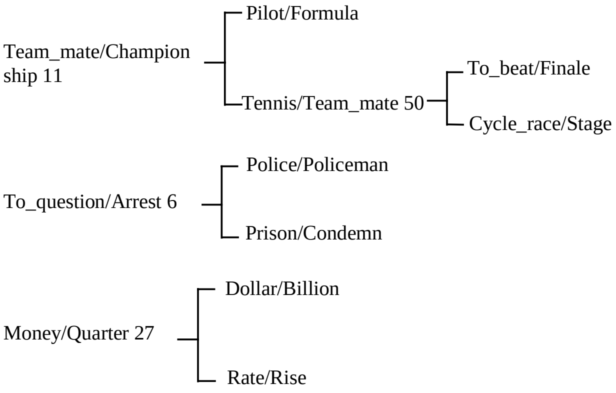

Figure 3. Three hierarchies of semantic domains

4 STRUCTURED DOMAIN LEARNING

As in [4], verbs allow us to categorize nouns. A class is defined by those nouns which play a same role relative to a same verb. In order to learn very homogeneous[3] classes, we only apply this principle on words belonging to a same context, i.e. a domain.

4.1. Syntactic analysis

In order to find the verbs and their arguments in the texts, we use the syntactic analyzer Sylex [8], [9]. Figure 4 shows a little part of the results of Sylex for a sentence. The first part exhibits lexico-syntactic information for the words and this for four different interpretations pointed out by the string "**taux 4**" meaning an ambiguity rate of 4. This rate is due to the fact that Sylex cannot solve two ambiguities: the ambiguity of "*laisse*" between the verb "*laisser*" (to let) and the noun "*laisse*" (leash) and the ambiguity of "*critique*" between the verb "*to criticize*" and the noun "*criticism*". Note that Sylex does not consider the adjectival form which is the right interpretation here. The second part shows syntactic links found by Sylex. Between parenthesis are references to the words in

[3] We call homogeneous a class that contains words that denote a same concept in the corresponding domain.

the preceding analysis. Here Sylex has found four times the same interpretation in each of its possible analyses. In this case, we count one occurrence of the link. However if it finds several times the same relation between a verb and different words, for example several possible subjects, then we keep all the different interpretations because we have no way to choose between them. We make the reasonable expectation that the false interpretations will have much less occurrences in the corpus and so, will be filtered out during the rest of the processing.

```
******************** Phrase   193-466   ************************************
"L'état de santé critique du pilote autrichien Karl Wendlinger (Sauber-Mercedes),
victime d'un grave accident jeudi matin lors des premiers essais du Grand Prix de
Monaco de Formule Un, laisse planer une menace sur le déroulement de la course,
dimanche en Principauté."
******************** Partie 1   193-466   taux 4 ***************************
"L'état de santé critique du pilote autrichien Karl Wendlinger (Sauber-Mercedes),
victime d'un grave accident jeudi matin lors des premiers essais du Grand Prix de
Monaco de Formule Un, laisse planer une menace sur le déroulement de la course,
dimanche en Principauté."
                        <Lexico-Syntactic information>
  193-195   (164) "L'" "le" [gs.1,avn,pdet.1] pdet : singulier elision dmaj
  195-208   (165) "état de santé" "état de santé" [gs.1,nom.1] nom : masculin singulier
mot_compose locsw
...<snip>....
  382-388   (203) "laisse" "laisse" [gs.12,nom.1] nom : feminin singulier
  389-395   (204) "planer" "planer" [gs.13,verbe] verbe : infinitif
...<snip>....
  193-195   (16) "L'" "le" [gs.1,avn,pdet.1] pdet : singulier elision dmaj
  195-208   (117) "état de santé" "état de santé" [gs.1,nom.1] nom : masculin singulier
mot_compose locsw
...<snip>....
  382-388    (211) "laisse" "laisser" [gs.13,verbe] verbe : singulier autoontif antiontif
anontif present indicatif subjonctif imperatif
  389-395   (212) "planer" "planer" [gs.14,verbe] verbe : infinitif
...<snip>....
                              <Syntactic Links>
`L'état de santé critique' (164) ->- cn head ->- `du pilote autrichien' (170)
`planer' (204) ->- a2 head ->- `une menace' (205)
...<snip>....
`planer' (153) ->- a2 head ->- `une menace' (154)
...<snip>....
`planer' (161) ->- a2 head ->- `une menace' (162)
...<snip>....
`planer' (212) ->- a2 head ->- `une menace' (213)
`sur le déroulement' (66) ->- cn head ->- `de la course' (235)
```

Figure 4. An extract of a sentence analysis by Sylex.

The results of Sylex are very detailed and not easy to parse directly with, say, Perl. Furthermore, we do not need all the information it extracts. In fact, we only need to find the verb with its links and the head nouns arguments of these links. So, we have developed a formal grammar that extracts from these raw analyzes the associations between a verb and its arguments. This grammar extracts links from the results of Sylex in the following format:

i # j   verb # **token1** # *lemma1* # k <u>rel</u> # **token2** # *lemma2* # l

where i and j are the boundaries of the sentence that contains the link in the corpus; **token1** and *lemma1* are the token and the lemma of the verb respectively ; <u>rel</u> is the syntactic relation which can be "subject", "direct object" or a preposition ("to", "from", etc.) ; **token2** and *lemma2* are the token and the lemma of the head noun of the noun phrase pointed by the relation; lastly, *k* and *l* are the indexes in the corpus of **token1** and **token2** respectively. Figure 5 shows some links that we have extracted from the results of Sylex.

token1	*lemma1*		<u>rel</u>	**token2**	*lemma2*
hang over	*hang over*		<u>subject</u>	**threat**	*threat*
play	*play*		<u>object</u>	**cup**	*cup*
hear	*hear*		<u>of</u>	**sources**	*source*

Figure 5. Examples of extracted links

Sylex, as other syntactic analyzers, has difficulties with some constructions and as a consequence introduces errors that can cause problems to the remaining of the system. Some common errors are the bad interpretation of the passive form that causes a subject to be analyzed as a direct object and conversely, a direct object to be viewed as a subject. Another common error is that it often happens that Sylex does not find any link in a phrase. That’s what we will call *silence*. We will see in Section 5 that we can obtain good results despite these problems thanks to the redundancy needed to validate the links in the next steps of processing. But another consequence of this redundancy needs is that the system must use great quantities of texts in order to create classes with a satisfactory size.

Having gotten the syntactic links in the texts, we want to group them relatively to the belonging of their text segment to a Thematic Unit. So, we define a Syntactic Thematic Unit (STU) as a set of <*Verb*→*syntactic relation*→*Noun*> structures, i.e. a syntactic relation instantiated with a verb and a noun. We will refer to these structures as Instantiated Syntactic Relations or ISR. We are able to put in relation the links extracted from the results of Sylex and the words contained in the domains because each domain in the thematic memory remembers which thematic units have been used to create it. In the same way, each thematic unit remembers the part of text it comes from.

4.2. Aggregation

In order to construct group of words with very similar meanings, we want to group the nouns appearing with the same syntactic role in relation to a verb inside a Domain. Then, a Structured Domain (SD) is a set of <*Verb*→*syntactic relation*→$Noun_1$, …, $Noun_n$> structures, i.e. an aggregated ISR.

STUs related to a same domain are aggregated altogether to form a Structured Domain. Aggregating a STU within a SD consists of:

- aggregating their ISR that contain a same verb ;
- adding new ISR, i.e. adding new verbs with their arguments made of a syntactic relation and the lemmatized form of a noun.

Figure 6 shows the aggregation of a SD and three ISR. This example shows all the possible effects of the aggregation. In the figure, bold elements represent new or updating data. Aggregating an ISR in a SD that already contains the verb of the ISR leads to increment the occurrence number of the verb, as for *play* in the example. Similarly, the occurrence number of same nouns related to the verb by the same relation are updated (as for *match*), and new relations with their associated nouns are added to the verb. In the example, the subject *champion* is added. An ISR with a new verb is simply added with an occurrence of 1, as for <to *lose*→ *object*→ *championship*>.

Syntactic Domain source		
to play [4]	object	cup [3], match [1]
	with	ball [1]
to win [2]	subject	player [1]
	object	match [1]
3 Instantiated Syntactic Relations sources		
to play	subject object	champion match
to lose	object	championship

Syntactic Domain result		
to play [**5**]	object	cup [3], match [**2**]
	with	ball [1]
	subject	**champion [1]**
to win [2]	subject	player [1]
	object	match [1]
to lose [1]	**object**	**championship [1]**

Figure 6. An example of the aggregation of three ISR in a SD

Classes of nouns in the produced SDs contain a lot of words that disturb their homogeneity. These words often belong to parts of the different TU at the origin of the SD that are not very related to the described topic. Either they result from an error of the topic segmentation process or they correspond to a meaning of a verb scarcely used in the current context. Another possibility is that the ISR results from an error of Sylex. As these cases do not often recur for the same words in the same context, their nouns are weekly weighted in the corresponding domains. This characteristic gives us a mean to filter the class content: each noun that possesses a weight lower than a threshold is removed from the class. By this selection, we reinforce learning classes of words according to their contextual use. Figure 7 shows two aggregated links first obtained without filtering in its upper part and the filtered counterparts in its lower part. The class associated to the verb '*to establish*' has been completely removed as the weights of both '*base*' and '*zone*' are lower than the threshold, while the class related to the verb '*to answer*' with the '*object'* link has been reduced by removing '*list*'. We can see on this example that this filtering is efficient: the verb '*to establish*' as the words '*base'* and '*zone'* are not very related to the domain of '*nuclear weapons*' from which this example is taken and the usage of '*to answer a list*' has a very low probability. More details on the effects of the filtering process will be given in section 5.

to establish	*object*	base, zone
to answer	*object*	document, question, list
~~**to establish**~~	~~*object*~~	~~base, zone~~
to answer	*object*	document, question, ~~list~~

Figure 7. Filtered aggregated links in a domain about nuclear weapons

In the principle, the described operations are not very complicated. The difficulties comes from the necessity to work with data coming from various tools. Furthermore, for performance and practical reasons, we do not apply the chain of tools text by text. The natural way to see the process would be to :

- read a text,
- extract the TUs from it,
- extract the corresponding STUs,
- add each TU to its domain,
- add each STU to its corresponding domain,

and after the processing of all the texts, to filter the classes.

In fact, each computing step is done on the entire corpus and the results are next aligned. This allows us to save computation time as we do not have to run each tool multiple times. However we have to deal with dictionaries and indexes for various files and tools.

5 RESULTS

The experiments[4] we have conducted had as a goal to show that SVETLAN' lead to learn classes of words which obviously belong to the same concept in the domain. To obtain such results we have chosen to run our system on one month of *AFP (Agence France Presse*) wires, that forms a corpus stylistically coherent but that covers varied subjects with very polysemous and non specific verbs.

These wires are made of 4,500,000 words and 48,000 sentences in 6,000 texts. The thematic analysis gives 8,000 TUs aggregated in 2,000 domains. More details on these domains can be found in [5]. From these 48,000 sentences, 117,000 different Instantiated Syntactic Links are extracted by Sylex. 24,000 of these links concern subject, direct object, or circumstantial complements introduced by a preposition and are integrated in 1,531 Structured Domains.

After aggregating, but before filtering, the system obtains 431 aggregated links with two or more arguments, equivalent to 431 word classes. Some of them, such as <*to manufacture* → *direct object* → *bomb, weapon*> are good. Nevertheless other classes are heterogeneous as <*to return* → *direct object* → *territory, strip, context, synagogue*> (here strip comes from the Gaza Strip), or clearly mix different meanings of a verb, like <*to quit* → *direct object* → *base, government*> which mix together the meanings "to leave a place" and "to retire from an institution". For the two latter cases, one can see the interest to take into account the fact that the domains contain words with different weights representing their relevance to this domain. The higher the weight, the higher the relevance of this word in this domain. So we apply the aforesaid filter to our classes and retain only those with weights higher than a threshold. The class <*territory, strip, context, synagogue*> is corrected to <*territory, strip*> and < *base, government*> is removed.

Among the wrong classes, some are due to errors of Sylex, as <*to confer*→ *direct object* → *price, actor*> where *actor* should be linked to *to confer* by the preposition *to*. The remaining others are due to the extensive use of two different meanings of the verb in the same domain, as for*: <to conduct/to manage* → *direct object* → *delegation, negotiation*> (in French: "conduire une négociation/une délégation"). This kind of error is inherent to the method we use and should be removed by other means. Note that the correctness of the links have been manually judged by ourselves. The precision measure used below is the ratio between the number of good classes and the total number of classes. We cannot define a recall measure because we have no way to know which classes we

[4] Please note that all the tests have been made in French. So the English examples that appear here are translations from French.

miss. To our knowledge, there is no existing resources with associated classes that would allow us to formally judge the results.

We have tried two thresholds: 0.05 and 0.1. Figure 8 details the results for both.

Threshold	**Total**	**Good**	**Sylex errors**	**Remaining errors**
0.05	73	46 63%	13 18%	14 19%
0.1	38	27 71%	7 18%	4 11%

Figure 8. Results of the filtering for two thresholds

After filtering, a lot of classes are removed but the remaining classes are well funded in most cases. An example of a retained class for both thresholds is :

<to injure→ subject → colonist, soldier>

With a threshold set to 0.1 rather than to 0.05, we retain only 38 links, but we gain 8% in precision. If we ignore the errors due to Sylex, the real precision of SVETLAN' is in the first case 78% and in the second case 87%. It is very good and shows the interest there is to choose a good threshold.

Our experiments lead to homogeneous classes containing words denoting a same concept, though these classes contain few words. In order to directly view the interest there is to construct and cluster classes of words in being guided by their belonging to a domain, it is interesting to see what kind of classes would be obtained by the merging of all domains, that is to say : creating context-free classes. So, we have applied the same aggregation principle to the same corpus but without taking into account the domains. Just below, we show two classes for the verb "*to replace*". The top one is made context-free and the bottom one is made inside a domain. This verb is very general. Virtually everything can be replaced !

to replace	*object*	text, constitution, trousers, combustible, law, dinar, rod, film, circulation, judge, season, device, parliament, battalion, police, president, treaty
to replace	*object*	combustible, rod

The first group of words merges very different senses while the second class, much more little, is better because it contains words referring to very similar concepts: a rod of enriched uranium is nuclear combustible, thus the words "*rod*" and "*combustible*" actually denote the same concept in the nuclear domain. Another example is the following, for the verb "*to attribute*":

to attribute	*object*	talk, prize, decoration, pope, responsibility, television, attempt, letter, contract, ministry, jury, funds, authority, note, bonus, band, bombing
to attribute	*object*	prize, decoration

Obtaining meaningful classes with a corpus such as *AFP* shows the efficiency of our method. Moreover, it is very good to obtain cohesive classes for verbs very general and polysemous.

At this time, the class sizes are little. They do not contain a lot of words. A way to enlarge them could be to regroup classes that are related to the same verb, by the same syntactic relation in two domains belonging to the same hierarchy, i. e. a same more general context, assuming the words always have the same meaning. However this method has to be tested on more results in order to prove its reliability. With our results, we would build for example (law, constitution, article, disposition) in the domain of "*Law*" and (rebel, force, northerner, leader) in the domain of "*conflict*".

SVETLAN', in collaboration with SEGAPSITH, allows an automatic learning of structured semantic domains. Instead of just having sets of weighted words for describing semantic domains, domains are described by a set of verbs related to classes of words by a syntactic link. Besides, we can also view this base as semantic classes, each one being related to its context of interpretation.

As SVETLAN' works with very specific domains, it builds small classes. In order to generalize them, we could apply a process analogous to ASIUM, that merges classes independently of the related verbs according to a similarity measure, even if, in our case, this generalization process would operate in a same general domain. Afterwards, ASIUM asks an expert to validate its results.

Words are often polysemous or ambiguous. However, when used in context, they only denote one meaning, and moreover this meaning is generally the same in different occurrences of a same context. When building classes of nouns according to their contextual use, we avoid mixing all the meanings of a word, either for the verbs or for the nouns. Such a result can be exhibited in the classes (law, constitution) and (law, article, disposition) in the juridical context, where the words "*article*", "*constitution*" and "*disposition*" do not attract synonymous of their other meanings as "*section*", "*composition*" and "*aptitude*" for example.

6 RELATED WORKS

There is a lot of works dedicated to the formation of classes of words. These classes have very various status. They can contain words belonging to the same semantic field or near synonymous.

WordNet [1] is a lexical database made by lexicographers. It aims at representing the sense of the bigger part of the lexicon. It is composed of Synsets. A Synset is a set of words that are synonymous. These Synsets are linked by IS A relations. Its coverage is large but this is, in a sense, a shortcoming as its classes are too large and do not refer to precise meanings. Indeed, the generality of its contents makes it difficult to use in real sized applications that are often centered on a domain. It rarely can be used without a lot of manual adaptation.

IMToolset, by Uri Zernik [2], extracts, for a word, several clusters of words from text. Each of these clusters reflects a different meaning of the studied word. This extraction is done by scanning the local contexts of the word, the 10 words surrounding it in the texts. These signatures are statistically analyzed and clustered. The result is groups of words that are similar to our domains but more focused on the sense of a word alone.

We have already stressed out some characteristics of ASIUM by D. Faure and C. Nedellec [4], and we give here some more details. ASIUM learns subcategorization frames of verbs and ontologies from text using syntactic analysis and a conceptual clustering algorithm. It analyses texts with Sylex and creates basic clusters of words appearing with a same verb and a same syntactic role or preposition, as do SVETLAN'. These basic classes are then clustered to create an ontology by the mean of a cooperative learning algorithm. The main difference with SVETLAN' is this cooperative generalization part: ASIUM depends on the expert who has to valid, and possibly to split, the clusters made by the algorithm.

This approach is justified for specialized technical texts, but ASIUM, applied on texts such as *AFP* wires would certainly not be able to extract good basic classes. Furthermore, as each word does not occur a lot in these texts, the distance is not appropriate to the grouping of our classes. On the contrary, on technical texts and with the cooperation of an expert, ASIUM will certainly obtain better results than ours from the point of view of the domain coverage.

7 CONCLUSION

The system SVETLAN' we propose, in conjunction with SEGAPSITH and the syntactic parser Sylex, extracts classes of words from raw text. These classes are created by the gathering of nouns appearing with the same syntactic role after the same verb inside a context. This context is made by the aggregation of text about similar subjects. The first experiments carried out give good results. But they also confirm that a great volume of data is necessary in order to extract a large quantity of lexical knowledge by the analysis of syntactic distributions. Moreover the very low recall of the syntactic parser and its systematic errors on some constructions, for example the passive form, which is very common in the journalistic style of our corpus, reduce the number and size of the classes. To solve this problem, we envisage trying another analyzer or adding a post-processing step to Sylex that detects the passive form by using data already in its output. These adaptations and the study of more larger corpora will allow us to obtain a good coverage of numerous semantic domains. So, we will be able to give valuable semantic data useful in a lot of applications as information retrieval systems or word sense disambiguation systems.

REFERENCES

[1] Christiane Fellbaum, WordNet: an electronic lexical database, The MIT Press, 1998

[2] Uri Zernik, TRAIN1 vs. TRAIN2: Tagging Word Senses in Corpus, RIAO'91, 1991

[3] Gregory Greffenstette, Explorations in automatic thesaurus discovery, Kluwer Academic Pub., Boston, 1994

[4] David Faure and Claire Nedellec, ASIUM, Learning subcategorization frames and restrictions of selection. In Y. Kodratoff ed., proceedings of 10th ECML – Workshop on text mining, 1998

[5] Olivier Ferret and Brigitte Grau, A Thematic Segmentation Procedure for Extracting Semantic Domains from Texts, Proceedings of ECAI'98, Brighton, 1998.

[6] Olivier Ferret, How to thematically segment texts by using lexical cohesion? Proceedings of ACL-COLING'98 (student session), pp. 1481-1483, Montreal, Canada, 1998.

[7] C.-Y. Lin. Robust Automated Topic Identification, Doctoral Dissertation, University of Southern California, (1997).

[8] Patrick Constant, Analyse Syntaxique Par Couches. Ph.D thesis, École Nationale Supérieure des Télécommunications, April, 1991.

[9] Patrick Constant, L'analyseur linguistique SYLEX. 5ème ecole d'été du CENT, 1995.