Pablo Gamallo Otero, Gabriel Pereira Lopes, Alexandre Agustini

Automatic Acquisition of Formal Concepts from Text

This paper describes an unsupervised method for extracting concepts from *Part-Of-Speech* annotated corpora. The method consists in building bi-dimensional clusters of both words and their lexico-syntactic contexts. The method is based on *Formal Concept Analysis* (FCA). Each generated cluster is defined as a *formal concept* with a set of words describing the extension of the concept and a set of contexts perceived as the intensional attributes (or properties) valid for all the words in the extension. The clustering process relies on two concept operations: *abstraction* and *specification.* The former allows us to build a more generic concept by intersecting the intensions of the merged concepts and making the union of their extensions. By contrast, specification makes the union of the intensions and intersects the extensions. The result is a concept lattice that describes the domain-specific ontology underlying the training corpus.

1 Introduction

The pervasive and explosive proliferation of information systems requires a better understanding, control, and management of the conceptual structure underlying information. Solutions to represent conceptual structures are emerging in the form of *ontologies*, i.e., computer-based repositories of formal concepts about application domains [15]. It is broadly assumed that, not only database schemas or semi-structured data, but also textual sources play an important role to extract concepts and learn ontologies. Recent work in ontology learning has started to develop methods for the automatic construction of conceptual structures [12]. This is typically done in an unsupervised manner on the basis of text corpora relevant for the domain of interest. We have opted for extraction techniques based on unsupervised learning methods since these do not require specific external domain knowledge such as thesauri, and the portability of these techniques to new domains is much better.

This paper describes an unsupervised method for extracting concepts from *Part-Of-Speech* annotated corpora. The method consists in building bi-dimensional clusters of both words and their lexico-syntactic contexts. Each cluster, which represents a concept such as "entities in danger" is the result of either merging or unifying their constituents (i.e., words and contexts). In the last step of the method, we will identify prototypical constituents from the generated clusters. These prototypes will be used as concept centroids in the last step of our method: word classification.

The basic intuition underlying our corpus-based approach is that similar concepts can be aggregated to generate either more specific or more generic concepts, without inducing odd associations between contexts and words. A new concept is generated by specification if we make the union of the constituent contexts (intension expansion) while the words are intersected (extension reduction). A new concept is generated by abstraction if the lexico-syntactic contexts are intersected (intension reduction), while we make the union of the constituent words (extension expansion). Intersecting words and contexts in an accurate way allows us to generate tight clusters with prototypical constituents. The theoretical background of our work is based on is *Formal Concept Analysis* (FCA). The clusters we acquired have all the features of "formal concepts" in FCA. Figure 1 shows a cluster of words and lexico-syntactic contexts learnt by our system. The cluster represents a formal concept with a word extension and a descriptive intension. The clustering algorithm only selects those contexts that can co-occur with all words in the extensional set.

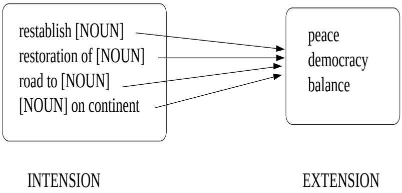

Abbildung 1: *A bi-dimensional cluster generated by our method*

2 Related Work

Local syntactic contexts have been largely used to extract classes (or concepts) of semantically similar words. Yet, approaches differ in the way they define word similarity. Some of them assume that two words are similar if they co-occur with a number of identical local contexts [6, 8]. Semantic similarity is then computed by using the whole set of local contexts associated with each word. Unfortunately, the contexts of a word are usually very heterogeneous and multidimensional. They impose different selection restrictions and then select for different semantic facets or senses of a word. For instance, the noun *organisation* appears, at least, in two different types of contexts: those selecting for temporal events (*organisation* of the party, to finish the *organisation*, etc.) and those requiring institutions (hired by the *organisation*, the president of the *organisation*, etc.). Given such a contextual diversity, this word can be semantically associated to a list of very heterogeneous nouns: *procedure*, *action*, *company*, *ministry*,.... This "absolute" view of semantic similarity leads to collapsing heterogeneous contextual information onto a single axis.

In order to induce semantically homogeneous lists of words, other approaches do not compare the semantic similarity between words, but between $< context, word >$ pairs and sets of those pairs. These sets are perceived as lexico-semantic concepts (also called "classes" or "selection types") [11, 16]. Given two vocabularies, W and $CNTX$, which represent respectively the set of words and the set of local contexts, a class ou concept is defined as a pair $< CNTX', W' >$, where $CNTX' \subseteq CNTX$ and $W' \subseteq W$. In this model, the same word or context can in principle belong to more than one concept. So, the positive side of these approaches is that they try to take into account linguistic polysemy. Some difficulties arise, however, in the process of class generation. Those approaches propose a clustering algorithm in which each concept is represented by the centroid distributions of all of its members. This is in conflict with the fact that many words and local contexts can significatively involve more than one semantic dimension. As a result, the clustering method turns out to be too greedy since it overgenerates many wrong associations between words and local contexts. For instance, the work by Roth induced a particular concept containing the association between verbs (viewed as local contexts) such as *cost, play, spend, be, ...* and nouns like *money, role, fund, part*, etc. See Figure 2. This concept contains several wrong association pairs: for instance, $< cost[N], role >$, $< play[N], fund >$, etc. Besides that, there also are too broad-sense words (*be, use, part, time, ...*), which may belong to almost any concept.

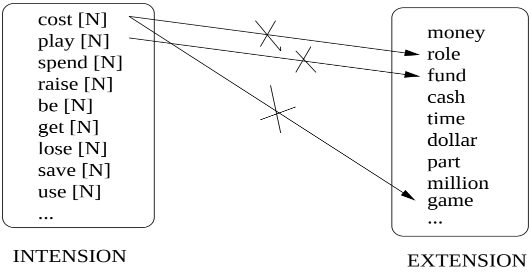

Abbildung 2: *An excerpt of a bi-dimensional cluster appearing in [16]*

To avoid these drawbacks, a more recent approach tried to limit the information contained in the centroids by introducing a process of "clustering by committee" [10]. The centroid of a cluster is constructed by taking into account only a subset of the cluster members. This subset, called "committee", contains the more representative members (prototypes) of a concept. So, the main and more difficult task of such an approach is to first identify a list of committees, i.e., a list of semantically homogeneous clusters. Committees represent basic linguistic concepts of similar words and are useful for word classification.

Other approaches also try to identify homogenous clusters representing basic semantic concepts. The main difference with regard to the former method is that each basic cluster is constituted, not by similar words, but by a set of similar local contexts [3, 9, 1, 14, 5]. The method is focused on computing the semantic similarity between lexico-syntactic contexts. Words are no more seen as entities to be clustered but as features of contexts. These are taken as the main objects in the clustering process. As lexico-syntactic contexts turn out to be less polysemic than words, these approaches assume that searching for concepts of homogeneous contexts is easier and more efficient than to find tight concepts of semantically related words. The main problem, however, is that the basic clusters of contexts identified in the first step tends to be very small and specific. The average size of a basic cluster is only two members. In order to generate larger concepts, most of these approaches require a second step with a greedy clustering process. Unfortunately, this greedy clustering step tends to overgenerate many context-word associations.

The method proposed in this paper is close to the last type of approaches described in the previous paragraph. Our main contribution is the use of very restrictive operations (specification and abstraction) in the process of building tight clusters. Thanks to these constraints on clustering, we try to solve the overgeneration problem. A tight cluster will be defined as a bi-dimensional entity consisting on both a set of words and a set of contexts, if only if each word is semantically associated to all contexts of the cluster.

3 Theoretical Background: Formal Concept Analysis

Formal Concept Analysis (FCA) [7, 13] is a particular method of data analysis and knowledge representation based on *Galois lattice* (also called *concept lattice*). In this framework, a concept is defined as a dual unit consisting of two parts: a set of objects (the extension of the concept) and a set of attributes or properties valid for all the objects in the concept (its intension). The family of these concepts obeys the mathematical axioms defining a lattice.

The main idea underlying FCA is to argue that a concept lattice is an efficient tool for several applications, such as lexical database design, ontology learning [12], knowledge acquisition, or conceptual clustering. In this paper, our contribution is to use a concept lattice to design a particular strategy of conceptual clustering.

3.1 Formal Concepts

To define formal concepts, FCA starts with the notion of formal context. A *formal context* is a triple $\Bbbk\colon = (O, A, R)$, where O is a set of objects, A, a set of attributes, and $\Re$ a binary relation between O and A, i.e. $\Re \subseteq O \times A$. A concept lattice of $\Bbbk$ is a partial order over all pairs of the form (E, I), where $E \subseteq O$, $I \subseteq A$ s.t.:

$$\begin{aligned} E &= \{o \in O \,|\forall a \in I, oRa\} \\ I &= \{a \in A \,|\forall o \in E, oRa\} \end{aligned} \tag{1}$$

The relationship oRa (which belongs to $\Re$) is read “the object o has the attribute a”. The pair (E, I) is called a *formal concept*, where E is the extension of the concept (i.e., the set of objects it comprises), and I is its intension, i.e., the set of attributes shared by all members of the concept’s extension. Partial order is defined as follows: if (E_1, I_1) and (E_2, I_2) are formal concepts, we define a partial order $\leq$ by saying that $(E_1, I_1) \leq (E_2, I_2)$ whenever $E_1 \subseteq E_2$. Equivalently, $(E_1, I_1) \leq (E_2, I_2)$ whenever

Tabelle 1: A formal context of “states”

	president (pr)	prime-minister (pm)	european union (eu)	kingdom (k)	islamic rules (ir)
Belgium (B)		X	X	X	
Portugal (P)	X	X	X		
Pakistan (PK)	X	X			X
Iran (I)	X				X
Saudi Arabia (A)				X	X

$I_1 \subseteq I_2$. Every pair of concepts in this partial order has a unique greatest lower bound (meet) and a unique least upper bound (join), so it satisfies the axioms defining a lattice. The greatest lower bound of (E_1, I_1) and (E_2, I_2) is the concept with objects $E_1 \cap E_2$ and attributes $I_1 \cup I_2$. The least upper bound of (E_1, I_1) and (E_2, I_2) is the concept with attributes $I_1 \cap I_2$ and objects $E_1 \cup E_2$.

3.2 A Toy Example

The cross table 1 depicts a small formal context. The elements on the left side are objects while the elements at the top are attributes (or properties) of the objects. The relationship between them is represented by crosses. In this toy example, the objects are some states and the attributes describe whether they have a president, prime-minister, or a king, whether they belong to the European Union or whether they are ruled by Islamic principles.

Figure 3 represents the concept lattice of the formal context in Table1. In the diagram, each node represents a formal concept, consisting of a set of objects noted below (the extension) and a set of attributes appearing above (the intension). A concept c_1 is a subconcept of a concept c_2 if only if there is a path of descending edges from the node representing c_2 to the node representing c_1. The label of an object o is always attached to the node representing the smallest concept with o in its extension. In Figure 3, the label “Iran” is in the concept with extension {’I’, ’PK’} and intension {’ir’, ’pr’}. There is no smaller concept with ’I’ in the extension. Conversely, the label of attribute a is always attached to the node representing the largest concept with a in its intension. For instance, the label “kingdom” is in the concept with extension {’B’, ’A’} and intension {’k’}. There is no larger concept with ’k’ in the intension set. The top and bottom concepts in a concept lattice are special. The top concept is the largest one since it has all objects in its extension. Its intension is often empty but does not need to be empty. The bottom concept is the smallest one and has all attributes in its intension. Its extension is empty when there are at least two attributes that are mutually incompatible. For instance, “being a kingdom” (’k’) and ”to have a President” (’pr’). The top concept can be considered as the “universal” concept and the bottom one the “null” concept of a formal context.

A central notion of a concept lattice is the duality of concepts. This duality implies that if one makes the sets of extensions larger, they correspond to smaller sets of intensions, and vice versa. In Figure 3 those nodes with larger extensions (at the top) tend to have only one attribute. On the bottom, nodes with larger intensions have only one object. However, in the middle of the diagram, we find more balanced nodes, i.e., concepts with a similar number of elements in both the extension and the intension. In our toy example, these balanced concepts represent useful notions to describe some political systems of states. For instance, the concept characterised by the properties "to have a President" and "to have a Prime-Minister" represents those states that are standard republics. Islamic republics, on the other hand, can be represented by the concept containing the properties "islamic rules" and "to have a President". We claim that balanced concepts tend to be significant and meaningful nodes in any ontology, terminology, or lexical database.

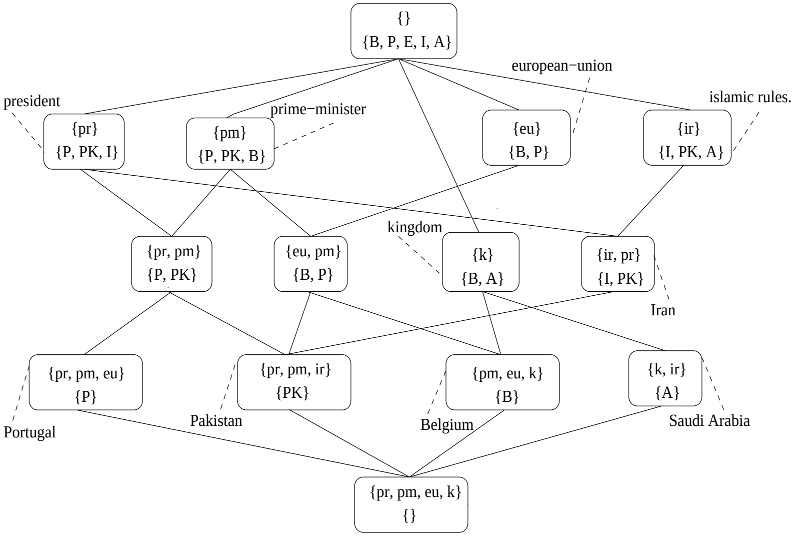

Abbildung 3: *A concept lattice from the formal context depicted in Table 1*

3.3 Building a Concept Lattice by Clustering of Words and Contexts

Following the ideas introduced above, we can build a lattice of formal concepts consisting of two linguistic dimensions. One dimension is the intension definition, i.e., a set of similar lexico-syntactic contexts with the same selection restrictions. The other one is its extension, i.e., the set of words appearing in such contexts and satisfying their semantic requirements. When the intension is very specific because it contains a large set of contexts, then the extension tends to be small. A lexico-syntactic context can be defined as a linguistic pattern constituted by a lexical word, a syntactic relation, and a morpho-syntactic position. For instance, "president of [NOUN]" is the lexico-syntactic context of nouns such as "Portugal", "Belgium", "Real Madrid", "republic", or "company", i.e. nouns denoting institutions with a president. In this particular application, co-occurrence in a corpus turns out to be the specific binary relationship between extensions and intensions. So, within a formal concept, each word in the extension "co-occurs" with each lexico-syntactic context in the intension.

New formal concepts are generated by means of a clustering process endowed with two complementary operations: *specification* and *abstraction*. If two similar formal concepts, FC_1 and FC_2, defined respectively as the pairs $< CNTX_1, W_1 >$ and $< CNTX_2, W_2 >$, are aggregated into a new concept, we can opt for two different operations:

specification: $FC_1 \, \Theta \, FC_2$, which represents a more specific concept whose intension is the set of contexts $CNTX_1 \cup CNTX_2$, and the extension the word set $W_1 \cap W_2$.

abstraction: $FC_1 \, \Phi \, FC_2$, which represents a more generic concept whose intension is the intersection $CNTX_1 \cap CNTX_2$, and the extension the union $W_1 \cup W_2$.

The clustering method we will describe in the following section makes use of these two operations. The resulting concepts generated by such operations give rise to a concept lattice. The more balanced concepts in that lattice will be the startpoint (i.e., centroids) of a further process: word classification.

4 The Method

Our method consists of 4 steps. In Step I, we describe the linguistic process allowing us to create context vectors. Step II introduces a clustering algorithm relying on the specification operation. The aim is to identify a list of balanced concepts. In Step III, these concepts are merged by a hierarchical clustering and the abstraction operation. As a result, we build a concept lattice with several unrelated abstract formal concepts at the top level. The specific information involved in the definition of each top abstract concept will be used in the following classification step. Finally, in Step IV, further words are classified and assigned to the appropriate formal concepts.

4.1 Step I: Building Context Vectors

In this step, lexico-syntactic contexts will be represented as vectors of word lemmas. The basic value of each vector position is the co-occurrence frequency between the context and the corresponding lemma. The whole vector space can be perceived as the *Formal Context* from which we will extract formal concepts.

To create the vector space, we first need to identify lexico-syntactic contexts from texts. We start by POS tagging the input corpus. Then, we use basic pattern matching techniques to identify potential binary dependencies. From each binary dependency, two

Binary Dependencies	Contexts
to (threat, health)	< threat to [NOUN] > < [NOUN] to health >
of (import, sugar)	< import of [NOUN] > < [NOUN] of sugar >
robj (approve, law)	< approve [NOUN] > < [VERB] law >
lobj (approve, president)	< president [VERB] > < [NOUN] approve >
modAdj(legal, document)	< legal [NOUN] > < [ADJ] document >
modN(area, protection)	< protection [NOUN] > < [NOUN] area >

Tabelle 2: Some binary dependencies and their corresponding lexico-syntactic contexts.

complementary lexico-syntactic contexts are selected. Table 2 shows some representative examples. A lexico-syntactic context defines a set of semantically related words. Given a binary dependency:

to (threat, health) ,

two templates are selected: < danger to [NOUN] >, which represents the set of nouns that can appear after "danger to", for instance, "health", "peace", "stability", etc. On the other hand, < [NOUN] to health > represents the set of nouns appearing before "to health": "danger", "access", "threat", etc. We follow the notion of *co-requirement* introduced in [5].

Note that *lobj* represents the relationship between a verb and the noun immediately appearing at its left; *robj* is the relationship between a verb and the noun appearing at its right. On the other hand, *modAdj* is the relationship between a noun and its adjective modifier and *modN* is the relation between two nouns: the head and its modifier.

Finally, each lexico-syntactic context is associated to its co-occurring words to build the vector space.

4.2 Step II: Extracting Balanced Concepts

4.2.1 Filtering Concepts

We start by filtering out lexico-syntactic contexts that are sparse in the training corpus. A context is sparse if it has high word dispersion. Dispersion is defined as the number of different word lemmas occurring with a lexico-syntactic context divided by the total number of different word lemmas in the training corpus. So, the vector space is only constituted by those lexico-syntactic contexts whose word dispersion is lower than an empirically set threshold.

Tabelle 3: The 5 most similar contexts to “threat to [N]”

{threat to [N]}	{risk to [N]}	0.213
{threat to [N]}	{endanger [N]}	0.191
{threat to [N]}	{[N] aspect}	0.172
{threat to [N]}	{damage [N]}	0.171
{threat to [N]}	{guarantee of [N]}	0.155

4.2.2 Context Similarity

Then, for each context with low dispersion, we compute its top-k similar ones, where $k = 5$, using a Dice coefficient as similarity measure [4].

Similarity between two lexico-syntactic contexts $cntx_1$ and $cntx_2$ is computed as follows:

$$Dice(cntx_1, cntx_2) = \frac{2 * \sum_i min(f(cntx_1, w_i), f(cntx_2, w_i))}{F(cntx_1) + F(cntx_2)} \tag{2}$$

where $f(cntx_1, w_i)$ represents the number of times $cntx_1$ co-occurs with the word lemma w_i. $F(cntx_i$ stands for the absolute frequency of $cntx_1$. This is the similarity score used to build the top-5 lists of similar contexts. For instance, Table 3 shows a list with the 5 most similar contexts to “threat to [N]”, according to the information extracted from the corpus *Europarl*.

4.2.3 Basic Concepts (input of clustering)

The basic concepts used as input of the clustering process are extracted from these ranked lists. Given the top-5 list associated to a lexico-syntactic context (and the set of word lemmas it classifies), we build 5 basic concepts by aggregating that context to each one in the list. The words in the extension are those co-occurring with both contexts. Table 4 shows the five basic concepts associated to the context “threat to [N]” that were extracted from the ranked list in 3. These basic concepts are quite generic since their intension has only two attributes (2 contexts). They will be the input of the process of clustering by specification.

4.2.4 Clustering by Specification

The first basic concept, 00231, is taken as the centroid since it is constituted by the top-1 similar context to “threat to [N]” (see again Table 4). The clustering process consists in aggregating the remaining concepts together around the identified centroid if only if they share more than 50% of the word lemmas. All aggregations are made using the operator of “specification” since each generated concept is obtained by intersecting the two word sets of each aggregated concept. As a result, we obtain:

Tabelle 4: The top-5 concepts built around the context “threat to [N]”

00231	{threat to [N], risk to [N]}	{health, environment, security, price, peace, stability}
00232	{threat to [N], endanger [N]}	{whole, democracy, peace, life, health, environment, security, stability}
00233	{threat to [N], [N] aspect}	{welfare, safety, employment, health, security}
00234	{threat to [N], damage [N]}	{employment, integrity, peace, life, health, environment, fishing, stability}
00235	{threat to [N], guarantee of [N]}	{safety, democracy, peace, job, freedom, security, stability}

FC_{37}	{endanger [N], damage [N], threat to [N], risk to [N]}	{health, environment, peace, stability}

which is the result of two specification operations:

$$FC_{37} = 00231 \ \Theta \ 00232 \ \Theta \ 00234$$

Here, clustering involves the centroid, 00231, and two concepts, 00232 and 00234, which satisfy the similarity condition (share at least 50% of words). Note that the specification operation allows us to build concepts with a more balanced relationship between the extension and the intension. This process is repeated for the other top-5 lists of similar contexts extracted from the corpus. The set of balanced concepts generated at the end of the process is the input of the following clustering step.

4.3 Step III: Generating Abstract Concepts by Hierarchical Clustering

A standard hierarchical clustering takes as input the specific and balanced concepts built in the previous step to generate more generic ones. For this purpose, we make use of an open source software: Cluster 3.0[1]. In this step, we use the operation of abstraction to build the successive aggregations. So, each generated concept is constituted by both the union of word sets and the intersection of contexts. Table 5 illustrates the concept lattice organising the information around $NODE_{77}$. This top-level concept is obtained from two successive abstractions:

$$NODE_{77} = FC_{37} \, \Phi \, NODE_{30}$$
$$NODE_{30} = FC_{420} \, \Phi \, FC_{202}$$

[1] *http://bonsai.ims.u-tokyo.ac.jp/˜mdehoon/software/cluster/software.htm*

Tabelle 5: Hierarchical construction of the generic formal concept $NODE_{77}$

$NODE_{77} : NODE_{30} \Phi FC_{37}$	{endanger [N]}	{health, life, patient, environment, peace, stability, quality}
$NODE_{30} : FC_{202} \Phi FC_{420}$	{endanger [N], risk to [N]}	{health, life, patient, environment, quality}
FC_{202}	{*endanger [N]*, risk to [N], expense of [N]}	{*health*, life, patient, environment}
FC_{420}	{*endanger [N]*, risk to [N], plant [N]}	{*health*, life, quality}
FC_{37}	{damage [N], *endanger [N]*, risk to [N], threat to [N]}	{*health*, environment, peace, stability}

Words and contexts organised around $NODE_{77}$ seem to characterise the abstract concept of "entities in danger". Note that the concepts we are able to learn (e.g., entities in danger) do not try to represent word senses as the synsets do in WordNet. Rather, they characterise top-level concepts of an upper-level ontology. In our notation, concepts labeled as $NODE_i$ stands for those generated by abstraction, whereas those labeled with FC_i represent concepts generated by specification. Figure 4 depicts the diagram representation of Table 5. This is another visualisation of the same lattice sample.

In our framework, the same word lemma can belong to the extension of different top-level concepts. For instance, *environment*, which is a member of $NODE_{77}$, is also a member of another concept aggregating nouns such as *agriculture*, *interior*, *justice*, *culture*, and *finance*, by their association with contexts like "minister of [N]", "ministry of [N]", or "minister for [N]".

Finally, if we observe more carefully Table 5 and Figure 4, we find out that *health* and "*endanger [N]*" are the only elements appearing in the three specific bottom-level concepts. They be considered as the prototypical or more representative constituents of these concepts with regard to the training corpus (they are in italic in the table). Prototypical elements will play an important role in the following step: word classification.

4.4 Step IV: Word Classification

So far, the generated clusters have been loosing relevant information step by step, since they were aggregated using intersecting operations. Besides that, the intersecting aggregations did not allow us to infer context-word associations that were not attested in the training corpus. As has been mentioned above, our objective was to design a very restrictive clustering strategy so as to avoid overgeneralisations.

In order to both reintroduce lost information and learn new context-word associations, the last step aims at assigning more word lemmas to the balanced concepts generated in the first clustering process. A word is assigned to one or more concepts in the following way:

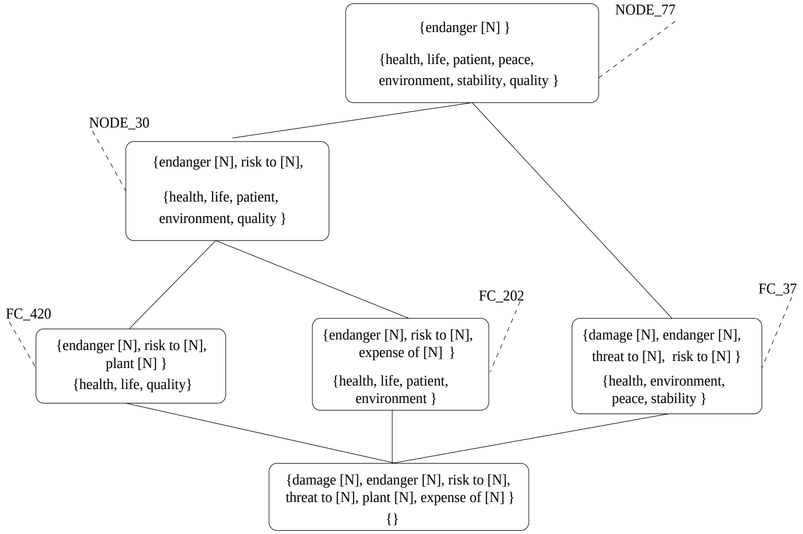
NODE_77
{endanger [N] }
{health, life, patient, peace, environment, stability, quality }
NODE_30
{endanger [N], risk to [N],
{health, life, patient, environment, quality }
FC_420
{endanger [N], risk to [N], plant [N] }
{health, life, quality}
FC_202
{endanger [N], risk to [N], expense of [N] }
{health, life, patient, environment }
FC_37
{damage [N], endanger [N], threat to [N], risk to [N] }
{health, environment, peace, stability }
{damage [N], endanger [N], risk to [N], threat to [N], plant [N], expense of [N] }
{}

Abbildung 4: *A diagram representation of the concept lattice depicted in Table 5*

We start by identifying the centroids used for classification. Given a concept, the representative centroid is constituted by the word lemmas and contexts that were considered as prototypes in the abstraction process (Step 2). For instance, the centroid extracted from concepts FC_{420}, FC_{202}, and FC_{37}, during the construction of $NODE_{77}$ is: $< \{endanger[N]\}, \{health\} >$. If a lemma fills the *classification conditions* imposed by this centroid, then it is assigned to the three balanced concepts in the example.

The classification conditions that a candidate lemma must fill are two: First, it must be *similar* to those word lemmas appearing in the centroid. Second, it must co-occur in the training corpus with the contexts of the centroid.

To measure similarity between word lemmas, we used the same coefficient as for context similarity: Dice score. In addition, each lemma was provided with a list containing its top-5 most similar ones. So, two word lemmas, w_i an w_j, are considered to be similar if only if w_i is in the top-5 list of w_j, or conversely, if w_j is in the top-5 list of w_i.

At the end of the classification step, our system was able to assign “security”, “democracy” , “growth”, and “energy” to the concepts organised around the top-level concept of *entities in danger*. Note that the acquired formal concepts refer to domain-dependent classes.

Tabelle 6: Corpus Data

	Balanced Concepts	Abstract Concepts	Classif.	Accuracy of Classif.
Público	264	91	492	92%
EuroParl	227	68	226	94%

5 Experiments and Evaluation

Experiments have been carried out using two different text corpora. A Portuguese corpus with 10 million tokens extracted from the general-purpose journal *O Público*, and an English excerpt (3 million tokens) of the European Parliament Proceedings (*EuroParl*). The Portuguese corpus was *POS* tagged with TreeTagger[2], using our own training corpus and lexicon[3]. The English corpus was tagged with an open source analizer: Freeling [2].

Table 6 depicts the number of balanced and abstract concepts extracted from each corpus, as well as the number of word classifications. Let's remember that balanced concepts were the output of Step II and abstract concepts the one of Step III. The extraction was only focused on nouns and nominal contexts. Note that not many abstract classes were learnt. This is in accordance with the basic ideas underlying formal ontology.

Measuring the correctness of the acquired lexico-semantic classes is not an easy task. We are not provided with a gold standard to which results can be compared. As the acquired concepts are corpus-dependent and do not represent word senses, there is no pre-existing ontology nor thesaurus containing the type of information our system is able to learn. Indeed, most concepts we learnt refer to domain-dependent knowledge. For instance, the class of world regions with internal conflicts and genocides: *Kosovo, Balcans, Serbia, Colombia, Chechnya, East Timor, Sierra Leona, region*. These word lemmas appear in contexts such as "conflict in [N]", "war in [N]", and "genocide in [N]". Another domain-dependent concept we learnt is the class of Portuguese towns with Bishop: *Viseu, Braga, Lisboa, Beja, Coimbra, Leiria, Guarda*. These names of towns co-occur with contexts such as "bispo de [N]", "diocese de [N]", "distrital de [N]", and "distrito de [N]"[4].

Other acquired concepts represent more heterogeneous classes and consist of open sets of words. For instance, we extracted an open set of entities in danger ($NODE_{77}$ above), a set of different forces that can be involved in a process (*threat, obstacle, access, impetus, contribution, ...*), a set of negative actions (*expulsion, terror, cleansing, genocide, massacre, destruction, atrocity, fighting, terrorism, ...*), different types of

[2] *http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html*

[3] Portuguese parameters can be downloaded in *http://gramatica.usc.es/∼gamallo/tagger.htm*

[4] These contexts can be translated as follows: "Bishop of [N], "diocese of [N]", "District of [N]", and "District of [N]", respectively.

statements (*remarque, comment, observation, word, point, statement, recommendation, suggestion, argument, request, ...*), and so on.

To evaluate the quality of the formal concepts we have acquired, we set a subjective evaluation protocol focused on the accuracy of word classification. Each word assignment to a concept was judged as correct or incorrect by a human evaluator. An assignment was considered as correct if the assigned word lemma is *semantically required* by all the lexico-syntactic contexts defining the concept. The 4th column of Table 6 shows the accuracy score. In fact, this evaluation measures the amount of overgeneration produced by the system. Overgeneration is about 8% in *O Público* and 6% in *EuroParl*.

In further research, we intend to develop a process of context classification. In this process, each formal concept will be assigned lexico-syntactic contexts that were not involved in the previous clustering steps. This way, we will be able to learn better intensional definitions of concepts.

Literatur

[1] P. Allegrini, S. Montemagni, and V. Pirrelli. Example-based automatic induction of semantic classes through entropic scores. *Linguistica Computazionale*, pages 1–45, 2003.

[2] X. Carreras, I. Chao, L. Padró, and M. Padró. An open-source suite of language analyzers. In *4th International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal, 2004.

[3] David Faure. *Conception de méthode d'aprentissage symbolique et automatique pour l'acquisition de cadres de sous-catégorisation de verbes et de connaissances sémantiques à partir de textes : le système ASIUM.* PhD thesis, Université Paris XI Orsay, Paris, France, 2000.

[4] W. Frakes. *Information Retrieval. Data Structures and Algorithms.* Prentice Hall, 1992.

[5] Pablo Gamallo, Alexandre Agustini, and Gabriel Lopes. Clustering syntactic positions with similar semantic requirements. *Computational Linguistics*, 31(1):107–146, 2005.

[6] Gregory Grefenstette. *Explorations in Automatic Thesaurus Discovery.* Kluwer Academic Publishers, USA, 1994.

[7] J. Hereth, G. Stumme, R. Wille, and U. Wille. Conceptual knowledge discovery - a human-centered approach. *Journal of Applied Artificial Intelligence*, 17(3):288–301, 2003.

[8] Dekang Lin. Automatic retrieval and clustering of similar words. In *COLING-ACL'98*, Montreal, 1998.

[9] Patrick Pantel and Dekan Lin. An unsupervised approach to prepositional phrase attachment using contextually similar words. In *ACL'00*, pages 101–108, Hong Kong, 2000.

[10] Patrick Pantel and Dekan Lin. Discovering word senses from text. In *Proceedings of ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 613–619, Edmonton, Canada, 2002.

[11] Fernando Pereira, Naftali Tishby, and Lillian Lee. Distributional clustering of english words. In *Proceedings of the 30th Annual Meeting of the Association of Comptutational Linguistics*, pages 183–190, Columbos, Ohio, 1993.

[12] Steffen Staab Philipp Cimiano, Andreas Hotho. Learning concept hierarchies from text corpora using formal concept anaylsis. *Journal of Artificial Intelligence Research (JAIR)*, 24:305–339, 2005.

[13] Uta Priss. Formal concept analysis in information science. *Information Science and Technology*, 40:521–543, 2006.

[14] M-L. Reinberger and W. Daelemans. Is shallow parsing useful for unsupervised learning of semantic clusters? In *4th Conference on Intelligent Text Processing and Computational Linguistics (CICLing-03)*, pages 304–312, Mexico City, 2003.

[15] M-L. Reinberger, P. Spyins, W. Daelemans, and R. Meersman. Mining for lexons: applying unsupervisded learning methods to create ontology bases. *Lecture Notes in Computer Science*, 2888:803–819, 2003.

[16] Mats Roth. Two-dimensional clusters in grammatical relations. In *Symposium on Representation and Acquisition of Lexical Knowledge: Polysemy, Ambiguity, and Generativity (AAAI 1995)*, 1995.