Learning concepts in C-CLASSIC$_{\delta\epsilon}$

Véronique Ventos, Pierre Brézellec, Henry Soldano, Dominique Bouthinon
LIPN,
Université Paris-Nord,
93430 Villetaneuse, France
Tel: 33 1 49 40 38 94
e-mail: <firstname>.<lastname>@lipn.univ-paris13.fr

Abstract

C-CLASSIC is a variant of CLASSIC in which concept learning is theoretically tractable (following the PAC-learning settings). C-CLASSIC$_{\delta\epsilon}$ is an extension of C-CLASSIC with the connectives δ (default) and ϵ (exception) allowing to incorpore default knowledge within concept definitions. Previous work was concerned with both deductive aspects (semantics, subsumption) and inductive aspects (PAC-learnability) of C-CLASSIC$_{\delta\epsilon}$. Our purpose here is to discuss practical aspects of concept learning from instances within C-CLASSIC$_{\delta\epsilon}$. We present an algorithm learning a disjunctive definition of a target concept from positive and negative instances. Since the bottom-up search of the concept space relies on Least Common Subsumers computations, the way how instances are represented in C-CLASSIC$_{\delta\epsilon}$ is crucial. In this paper we propose the use of a domain theory divided into default rules and incoherence rules in order to extend the descriptions of instances with excepted properties such as $\mathtt{A}^{\epsilon}$. The presence of $\mathtt{A}^{\epsilon}$ in the description of an instance means "the instance should have the A propery but has not".

Key-Words: Learning from instances, Default knowledge.

1 Introduction

One of the central problems studied in Machine Learning is the task of inducing a definition of a concept from a set of positive and negative instances of this concept. The choice of an appropriate representation formalism is very important for learning since it should be expressive, useful, applicable and efficient. For these reasons, Description Logics have been receiving increased attention in the Machine learning community (e.g. [Kietz and Morik, 1994; Frazier and Pitt, 1996]). Thus, in [Cohen and Hirsh, 1994b; 1994a], many results about learnability of DLs are given. In [Cohen and Hirsh, 1994a], the authors describe *k-CoreClassic* which is *Mistake-Bounded-Identifiable* and consequently PAC-Learnable (i.e. learning is efficient in the Valiant's sense of PAC-learnablity [Valiant, 1984]). It has been shown in [Cohen and Hirsh, 1994b] that CLASSIC is not PAC-learnable but that C-CLASSIC is. In [Ventos, 1996], we presented C-CLASSIC$_{\delta\epsilon}$ which extends C-CLASSIC with two non classical connectives δ and ϵ introduced in the toy DL $\mathcal{AL}_{\delta\epsilon}$ [Coupey and Fouqueré, 1997]). C-CLASSIC$_{\delta\epsilon}$ makes it possible to express default and excepted properties in the definition of concepts while keeping concept classification monotonic and polynomial. In [Ventos *et al.*, 1997], we have proven that C-CLASSIC$_{\delta\epsilon}$ is PAC-learnable. The goal of this paper is twofold. On one hand, we describe how it is possible to learn in C-CLASSIC$_{\delta\epsilon}$. On the other hand, we highlight the advantages of the connectives δ and ϵ by comparing learning in C-CLASSIC$_{\delta\epsilon}$ and learning in C-CLASSIC.

Learning of concept definitions, from positive and negative instances, in C-CLASSIC$_{\delta\epsilon}$ requires four steps. In the first two steps, the descriptions of instances are extended by adding excepted properties, given a set of positive and negative instances of a target concept to learn together with a nor-

CEUR-WS.org/Vol-11/ventos.pdf

malization procedure and a domain theory. The third step consists in applying a bottom-up learning algorithm, Elena$_{DL}$, to the extended description of instances. This generally does not result in a unique C-CLASSIC$_{\delta\epsilon}$ term but rather in a disjunction of such terms. During the last step, each of these terms is inserted in the taxonomy of concepts using classification.

This paper is organized as follows: section 2 briefly presents C-CLASSIC$_{\delta\epsilon}$. Section 3 presents learning in C-CLASSIC$_{\delta\epsilon}$ together with a comparison with learning in C-CLASSIC.

2 C-CLASSIC$_{\delta\epsilon}$

The set of connectives of C-CLASSIC$_{\delta\epsilon}$ is the union of the set of connectives of C-CLASSIC [Cohen and Hirsh, 1994b] and the connectives δ and ϵ. Example: `Student` $\sqcap$ δ`(publications AT-LEAST 4)` $\sqcap$ $\forall$ `Age:MAX 27` $\sqcap$ `publications FILLS {JAIR,AI}` $\sqcap$ $\forall$ `publications:(`$\forall$ `year:ONE-OF {94,95,96})` describes students who generally have at least four publications, are less than 27 years old, have at least one publication in JAIR and AI, and whose publications have been published in the years 94, 95 or 96.

The connective δ intuitively represents the common notion of default. For instance, having δViviparous as a conjunct in the definition of the concept Mammal states that mammals are generally viviparous. The connective ϵ is used to represent a property that is not present in the description of the concept or of the instance but that should be. For instance, the definition of an ornithorynchus in C-CLASSIC$_{\delta\epsilon}$ is `Ornithorynchus` $\equiv$ `Animal` $\sqcap$ `Vertebrate` $\sqcap$ `Oviparous` $\sqcap$ `Has-teats` $\sqcap$ `With-beak` $\sqcap$ `Viviparous`$^{\epsilon}$. The `Viviparous`$^{\epsilon}$ property expresses the fact that `Viviparous` should be in the definition of ornithorynchus since it is an animal having teats and generally such animals are viviparous. The presence of `Viviparous`$^{\epsilon}$ in the definition of ornithorynchus makes it possible to classify `Ornithorynchus` under the concept `Mammal`. Indeed, in this framework, a concept is subsumed by a default property if its definition contains either the default property, the strict property or the excepted property (i.e. for all concept `A`, δ`A` subsumes both `A` and `A`$^{\epsilon}$, more precisely δ`A` is the most specific concept subsuming both `A` and `A`$^{\epsilon}$). Concepts whose definition does not mention anything (strict, default or exception) about a property `A` are not subsumed by δ`A`.

In [Ventos, 1996], we have provided C-CLASSIC$_{\delta\epsilon}$ with an *intensional* semantics based on an equational system *EQ* which determines the properties of the C-CLASSIC$_{\delta\epsilon}$ connectives. In this semantics, concepts are characterized by a *normal form* of their properties (this approach is similar to [Dionne *et al.*, 1993]) rather than by the set of their instances. In [Ventos, 1997], it has been shown that intensional semantics coincides with the classical model-theoretic semantics C-CLASSIC for the non-default part.

EQ makes it possible to define the subsumption in C-CLASSIC$_{\delta\epsilon}$ as follows. Let $=_{EQ}$ denote the equality (modulo EQ axioms) between two *terms* of C-CLASSIC$_{\delta\epsilon}$.

Definition 1 (Subsumption) *Let C and D be two elements of C-CLASSIC$_{\delta\epsilon}$, $C \sqsubseteq D$, i.e. D subsumes C, iff $C \sqcap D =_{EQ} C$.*

3 Learning in C-CLASSIC$_{\delta\epsilon}$

The goal of learning is to induce a definition of a concept from individuals of C-CLASSIC$_{\delta\epsilon}$ which are known to be positive or negative instances of the concept to learn. Such a definition should subsume every positive instances and no negative one. In order to achieve this task, we designed Eléna$_{DL}$ (see section 3.3) whose core operation is the Least Common Subsumer (LCS) computation consisting in finding the largest expressible set of common properties between two concepts. For the purpose of the LCS algorithm, the description of positive and negative instances needs to be normalized and extended (see section 3.2). Learning operations are illustrated using an example introduced in section 3.1. Finally, we compare in section 3.4 learning in C-CLASSIC$_{\delta\epsilon}$ and learning in C-CLASSIC.

3.1 An example

Let S+=$\{e_1, e_2, e_3, e_4\}$ be a set of positive instances of the concept to learn and S-=$\{ce_1\}$ a set of negative instances of this concept. Let T be a domain theory made of two sets R and I. R=$\{$*Animal* $\sqcap$ *Has-teats* $\rightarrow$ *Viviparous*, *Animal* $\sqcap$ *Wings* $\rightarrow$ *Flies*$\}$ is a set of default rules meaning that generally animals having teats are viviparous and that generally animals having wings fly. I =$\{$ *Viviparous* $\sqcap$ *Oviparous*, *Inapt-to-fly* $\sqcap$ *Flies* $\}$ is a set of incoherent concepts meaning that an instance can not be both oviparous and viviparous and that it is impossible to fly and to be inapt to fly. We explain in the next section the way to use these sets. We give now the description of the instances in C-CLASSIC$_{\delta\epsilon}$.

e_1 $\equiv$ `Animal` $\sqcap$ `Viviparous` $\sqcap$ `Vertebrate` $\sqcap$ `Barks`.

e_2 $\equiv$ `Animal` $\sqcap$ `Vertebrate` $\sqcap$ `With-beak` $\sqcap$

Oviparous ⊓ Has-teats.
e_3 ≡ Animal ⊓ Vertebrate ⊓ Flies ⊓ Quacks.
e_4 ≡ Animal ⊓ Vertebrate ⊓ Lives-in-Africa ⊓ Wings ⊓ Big-feet ⊓ Inapt-to-fly.
ce_1 ≡ Animal ⊓ Vertebrate ⊓ Lives-in-the-sea ⊓ Scales.

The extension of the description of instances is performed in two steps: a normalization step followed by an extension of the normal forms thereby obtained.

3.2 Normalization and extension of the instances

The first step consists in computing a normal form of each instance using a normalization algorithm described and shown to be polynomial in [Ventos, 1997]. Note that the normalization strategy chosen leads the addition of implicit information. This normalization strategy is a kind of *partial saturation*, a proceeding which is often used in machine learning to make easier LCS computation (see for instance [Bisson, 1992]). We illustrate this step on the example described section 3.1. The computation of the normal forms of e_1, e_2, e_3, e_4 and ce_1 is very simple as far as they are only described with primitive concepts. The normalization consists therefore in adding for each primitive concept P the property δP thanks to the fact that δP subsumes P. We give below the normal form of the instance e_2 (called NF-e_2): NF-e_2 ≡ Animal ⊓ δAnimal ⊓ Vertebrate ⊓ δVertebrate ⊓ With-beak ⊓ δWith-beak ⊓ Oviparous ⊓ δOviparous ⊓ Has-teats ⊓ δHas-teats.

The second step consists in extending normal forms of instances by adding excepted properties. To achieve this task, we use a domain theory made of two sets: a set R of default rules in the form C → D meaning that if a concept is subsumed by C, it generally has the D property, together with a set I of incoherent concepts.

Definition 2 (R) *R is composed of n rules called $R_1,\ldots,R_n$ such that $R_i = Premisse_i \rightarrow Conclusion_i$ where $Premisse_i$ is a term of C-CLASSIC and $Conclusion_i$ a term of C-CLASSIC where concept conjunctions are only allowed in the value restriction of roles.*

Definition 3 (I) *I is composed of m incoherences expressed as terms of C-CLASSIC ($C_1,\ldots,C_m$)*

The connective ϵ can be viewed as a marker of an incoherence between properties of the instance and a default knowledge of the domain. The main goal of the step is therefore to detect a potential incoherence between the description of an instance and the conclusion of an applicable rule[1]. The detection of an incoherence allows us to add an excepted property to the description of the instance as we explain in the following. First of all, we describe more precisely what are incoherences and how to detect them. We distinguish two kinds of incoherences: incoherences of type 1 and incoherences of type 2.

An incoherence of type 1 corresponds to an incoherence linked to one or more general axioms concerning the connectives of the language (here C-CLASSIC[2]). For instance, child AT-LEAST 2 ⊓ child AT-MOST 1 is incoherent and more generally for all role R, R AT-LEAST m ⊓ R AT-MOST n is incoherent if m > n. To detect this kind of incoherences, we apply the normalization procedure defined for C-CLASSIC in [Ventos, 1997][3].

An incoherence of type 2 corresponds to an incoherence linked to a background knowledge (e.g. *Inapt-to-fly* ⊓ *Flies* is incoherent). Such incoherences are listed in I. Consequently, I is scanned in order to detect them.

We give now a sketch of the algorithm[4] allowing to extend the normal forms of instances: let e be an instance described in C-CLASSIC$_{\delta\epsilon}$, NF-e its normal form and ENF-e the extended normal form searched for. For each default rule, if the instance verifies the premisse of the rule and if the conclusion of the default rule is incoherent with the description of the instance (i.e. if it leads to incoherences of type 1 or 2), the conclusion is excepted and added to ENF-e. Note that in this framework, default rules are only used to add excepted properties. If no incoherence is detected, the conclusion of the rule is not added to the description of the instance. Thus, we avoid problems caused by multiple extensions of the instances.

We illustrate now the algorithm on an instance of the example.

NF-e_2 satisfies (i.e. is subsumed by) the premisse of the rule *Animal* ⊓ *Has-teats* → *Viviparous*. The addition of Viviparous to NF-e_2 leads to an incoherence (*Viviparous* ⊓ *Oviparous* ∈ I). The property Viviparous$^\epsilon$ is added to NF-e_2. Thanks to the

[1]A rule is applicable if its premisse subsumes the instance.

[2]We consider C-CLASSIC and not C-CLASSIC$_{\delta\epsilon}$ as incoherences are only meaningful for strict properties. See [Ventos, 1997] for more details.

[3]Applying this procedure on an incoherent term leads to normalize the term by ⊥ which denotes incoherences.

[4]The algorithm is given in appendix.

normalization δ(Viviparous$^{\epsilon}$) and δViviparous are also added to NF-e$_2$.
Adding the property Viviparous$^{\epsilon}$ makes it possible to highlight that e$_2$ is in conflict with *Animal* $\sqcap$ *Has-teats* $\rightarrow$ *Viviparous.* This information can be useful during the learning process. The normal forms obtained are called extended normal forms. They are used as inputs of the algorithm Eléna$_{DL}$ described in the next section.

3.3 Eléna$_{DL}$

C-CLASSIC$_{\delta\epsilon}$ containing only a limited kind of disjunction (the ONE-OF connective), many target concepts of pratical interest cannot be expressed using a single term of C-CLASSIC$_{\delta\epsilon}$. One way to overcome this limitation is to consider algorithms which learn a disjunction of terms rather than a single term. This is the case of Eléna$_{DL}$ designed in [Ventos, 1997]. Eléna$_{DL}$ learns concepts whose definition is such that Learned-concept $\equiv$ T$_1$ $\vee$ T$_2$ ... $\vee$ T$_n$ where T$_i$'s are terms of C-CLASSIC$_{\delta\epsilon}$. More precisely, each T$_i$ (called prototype) is the Least Common Subsumer of a subset of positive instances and is such that it does not cover any negative instance. Eléna$_{DL}$ is a bottom-up algorithm based on the algorithm described in [Brézellec and Soldano, 1993]. It uses a standard "greedy set covering algorithm" that at each iteration i removes from the learning set the positive instances covered by the selected prototype T$_i$.

In order to illustrate the behavior of Eléna$_{DL}$, let us go back to our example. The first prototype learned by Elena$_{DL}$ is LCS(e$_1$,e$_2$) (i.e. Animal $\sqcap$ δAnimal $\sqcap$ δViviparous[5] $\sqcap$ Vertebrate $\sqcap$ δVertebrate equivalent modulo EQ to Animal $\sqcap$ δViviparous $\sqcap$ Vertebrate). The instances e$_1$ and e$_2$ are removed from the learning set. The next learned prototype is the LCS of e$_3$ and e$_4$ which covers these two positive instances and no negative instances. As the current learning set contains no more positive instances, Elena$_{DL}$ returns the disjunction of the two previous prototypes (Animal $\sqcap$ δ Viviparous $\sqcap$ Vertebrate) $\vee$ (Animal $\sqcap$ δFlies $\sqcap$ Vertebrate).

3.4 Learning in C-CLASSIC$_{\delta\epsilon}$ vs C-CLASSIC

In C-CLASSIC, the descriptions of the five instances of the example presented in section 3.1 are the same as in C-CLASSIC$_{\delta\epsilon}$. The normal forms of these instances are identical to their initial descriptions since no default property can be expressed in C-CLASSIC.

As in C-CLASSIC$_{\delta\epsilon}$, Elena$_{DL}$ computes the Least Common Subsumer of different subsets of positive instances. Now, all the computed LCS cover the negative instance. No generalization can be done without covering the negative instance. Elena$_{DL}$ returns the disjunction of the four positive instances (e$_1$ $\vee$ e$_2$ $\vee$ e$_3$ $\vee$ e$_4$). Note that applying the algorithm LCSLearnDISJ[6] defined in [Cohen and Hirsh, 1994b] also returns the disjunction of the four positive instances.
The concept learned in C-CLASSIC is too specific to have a good predictive power. Thus, the new positive instance whose description is the following: Animal $\sqcap$ Vertebrate $\sqcap$ Lives-in-Australia $\sqcap$ Wings $\sqcap$ Big-feet $\sqcap$ Inapt-to-fly is recognized by the definition learned in C-CLASSIC$_{\delta\epsilon}$ (the instance satisfies the disjunct Animal $\sqcap$ δFlies $\sqcap$ Vertebrate after normalization and extension of its description) but it is not recognized by the definition learned in C-CLASSIC.

4 Conclusion

The paper mainly focuses on practical aspects of learning in C-CLASSIC$_{\delta\epsilon}$. We propose here a learning procedure including a prior extension of instance descriptions. This extension is obtained thanks to the extensive use of default knowledge and incoherence rules. A simple bottom-up learning algorithm is presented allowing to learn target concepts as disjunctions of C-CLASSIC$_{\delta\epsilon}$ terms which can further be inserted in the taxonomy.
>From an inductive point of view, extending C-CLASSIC with default connectives results in the following advantages:
- The expressive power of C-CLASSIC$_{\delta\epsilon}$ being greater than the expressive power of C-CLASSIC, it is possible to learn more concepts by using C-CLASSIC$_{\delta\epsilon}$ than by using C-CLASSIC.
- When the concepts to learn can be expressed using default knowledge, the disjunctive concepts learned in C-CLASSIC$_{\delta\epsilon}$ have less disjuncts than the concepts learned in C-CLASSIC.

Appendix

The extension algorithm is as follows :
an instance e described in C-CLASSIC$_{\delta\epsilon}$, NF-e its normal form, a set R $=\{R_1,\ldots,R_n\}$ of "default rules", a

[5] Note that this property does not belong to the LCS computed from the non extended normal forms of e$_1$ and e$_2$. Now, this property is crucial since it prevents the negative instance to be subsumed (let us remind that ce$_1$ has the properties Animal $\sqcap$ Vertebrate).

[6] This algorithm allows to learn disjunctions of C-CLASSIC terms.

set I = $\{C_1, \ldots, C_m\}$ of incoherent terms.
ENF-e the extended normal form of e.

Remove$\delta\epsilon$(d): transforms a term d of C-CLASSIC$_{\delta\epsilon}$ in a term of C-CLASSIC by removing default and excepted properties of d (since incoherences concern only strict properties).
Subsume(C,D): returns if C subsumes D, C and D being two terms of C-CLASSIC.
NF'(d): computes the normal form of a term d of C-CLASSIC.
NF(d): computes the normal form of a term d of C-CLASSIC$_{\delta\epsilon}$.

ENF-e $\leftarrow$ NF-e
e' $\leftarrow$ Remove$\delta\epsilon$(e)
R_i of R such that Premisse$_i$ $\rightarrow$ Conclusion$_i$ and Subsume(Premisse$_i$,e')

Add $\leftarrow$ {* Add is true if an excepted property has been added *}

NF'(e' $\sqcap$ Conclusion$_i$) = $\perp$ {* Conclusion$_i$ is incoherent with the description of e *}

ENF-e $\leftarrow$ ENF-e $\sqcap$ (Conclusion $_i$)$^{\epsilon}$
Add $\leftarrow$

Add then if there exists in I a term C such that Subsume(C,e' $\sqcap$ Conclusion$_i$)
ENF-e $\leftarrow$ ENF-e $\sqcap$ (Conclusion$_i$)$^{\epsilon}$

ENF-e $\neq$ NF-e then ENF-e $\leftarrow$ NF(ENF-e) {* Normalization of the modified description *}

References

[Bisson, 1992] G. Bisson. Conceptual clustering in first order logic representation. In *ECAI'92*, pages 458–467, Vienna, Austria, 1992.

[Brézellec and Soldano, 1993] P. Brézellec and H. Soldano. Eléna: a bottom-up algorithm. In Morgan-Kaufmann, editor, *10th International Conference on Machine Learning*, pages 9–16, Amherst, USA, 1993.

[Cohen and Hirsh, 1994a] W. W. Cohen and H. Hirsh. The learnability of description logics with equality constraints. *Machine Learning*, 2(4):169–199, 1994.

[Cohen and Hirsh, 1994b] W. W. Cohen and H. Hirsh. Learning the CLASSIC description logic: theoretical and experimental results. In *International Conference on Knowledge Representation and Reasoning*, pages 121–133. 1994.

[Coupey and Fouqueré, 1997] P. Coupey and C. Fouqueré. Extending conceptual definitions with default knowledge. *Computational Intelligence*, 13(2), 1997.

[Dionne *et al.*, 1993] R. Dionne, E. Mays, and F.J. Oles. The equivalence of model-theoretic and structural subsumption in description logics. In *13th International Joint Conference on Artificial Intelligence*, pages 710–716, Chambéry, France, 1993.

[Doyle and Patil, 1991] J. Doyle and R.S. Patil. Two theses of knowledge representation: language restrictions, taxonomic classification, and the utility of representation services. *Artificial Intelligence*, 48(3):261–297, 1991.

[Frazier and Pitt, 1996] M. Frazier and L. Pitt. Classic learning. *Machine Learning*, 25:151–193, 1996.

[Kietz and Morik, 1994] J. U. Kietz and K. Morik. A polynomial approach to the constructive induction of structural knowledge. *Machine Learning*, 14(2):193–217, 1994.

[Michalski *et al.*, 1986] R.S Michalski, I. Mozetic, J. Hong, and N. Lavrac. The multi-purpose incremental learning system aq15 and its testing application to three medical domains. In Morgan Kaufmann, editor, *Proceedings of the Fifth National Conference on Artificial Intelligence*, pages 1041–1045. 1986.

[Valiant, 1984] L.G. Valiant. A theory of the learnable. *Communications of the ACM*, 27:1134–1142, 1984.

[Ventos *et al.*, 1997] V. Ventos, P. Brézellec, P. Coupey, and H. Soldano. Lcs operation in c-classic$_{\delta\epsilon}$: formal properties and applications. In *International KRUSE Symposium Knowledge Retrieval, Use, and Storage for Efficiency*, pages 124–135, Vancouver, Canada, 1997.

[Ventos, 1996] V. Ventos. A deductive study of c-classic$_{\delta\epsilon}$. In *Proceedings of the International Workshop in Description Logics 96*, pages 192–196, Boston, USA, 1996.

[Ventos, 1997] V. Ventos. *C-CLASSIC$_{\delta\epsilon}$: une logique de descriptions pour la définition et l'apprentissage de concepts avec défauts et exceptions.* PhD thesis, Université Paris-Nord, France, 1997.