WP6-D2

Stream Reasoning for Open Data

Project full title:	Knowledge Driven Data Exploitation
Project acronym:	K-Drive
Grant agreement no.:	286348
Project instrument:	EU FP7/Maria-Curie IAPP/PEOPLE WP 2011
Document type:	D (deliverable)
Nature of document:	R (report)
Dissemination level:	PU (public)
Document number:	UNIABDN/WP6-D2/D/PU/b1
Responsible editors:	Yuan Ren
Reviewers:	Honghan Wu
Contributing participants:	UNIABDN,IBM
Contributing workpackages:	WP6
Contractual date of deliverable:	30 June 2014
Actual submission date:	30 June 2014

Abstract

Due to the dynamic nature of knowledge and data in semantic applications, *i.e.*, ontology stream querying technologies are essential for knowledge driven data exploitation systems. In this deliverable, we propose a novel approach to ontology stream querying that utilizes existing Database tools. Compared to existing works, this approach has a special advantage: the performance of query answering is automatically improved by the improvement of the state-of-the-art Database technologies. Deductive reasoning and inductive learning are the most common approaches for deriving knowledge. In real world applications when data is dynamic and incomplete, especially those exposed by sensors, reasoning is limited by dynamics of data while learning is biased by data incompleteness. Therefore discovering consistent knowledge from incomplete and dynamic data is a challenging open problem. In our approach the semantics of data is captured through ontologies to empower learning (mining) with (Description Logics) reasoning. Consistent knowledge discovery is achieved by applying generic, significative, representative association semantic rules. The experiments have shown scalable, accurate and consistent knowledge discovery with data from Dublin.

Keyword List

ontology stream, stream reasoning, stream maintenance

Project funded by the European Commission within the 7th Framework Programme/Maria Curie Industry-Academia Partnerships and Pathways schema/PEOPLE Work Programme 2011.

Stream Reasoning for Open Data

Yuan Ren[1], Jeff.Z.Pan[1], Jhonatan Garcia[1] and Alessandro Faraotti[2]

[1] Department of Computing Science, Aberdeen University, UK
Email: {y.ren, jeff.z.pan, jgarcia}@abdn.ac.uk

[2] IBM Italy, Roma, Italy
Email: alessandro.faraotti@it.ibm.com

30 June 2014

Abstract
Due to the dynamic nature of knowledge and data in semantic applications, *i.e.*, ontology stream querying technologies are essential for knowledge driven data exploitation systems. In this deliverable, we propose a novel approach to ontology stream querying that utilizes existing Database tools. Compared to existing works, this approach has a special advantage: the performance of query answering is automatically improved by the improvement of the state-of-the-art Database technologies. Deductive reasoning and inductive learning are the most common approaches for deriving knowledge. In real world applications when data is dynamic and incomplete, especially those exposed by sensors, reasoning is limited by dynamics of data while learning is biased by data incompleteness. Therefore discovering consistent knowledge from incomplete and dynamic data is a challenging open problem. In our approach the semantics of data is captured through ontologies to empower learning (mining) with (Description Logics) reasoning. Consistent knowledge discovery is achieved by applying generic, significative, representative association semantic rules. The experiments have shown scalable, accurate and consistent knowledge discovery with data from Dublin.

Keyword List
ontology stream, stream reasoning, stream maintenance

Contents

1 Introduction and Related Work

Knowledge discovery, as an area focusing upon methodologies for extracting knowledge through deduction (a priori) or from data (a posteriori), has been largely studied in Database and Artificial Intelligence. Deductive reasoning e.g., logic reasoning [Reiter, 1980] gains logically knowledge from pre-established (certain) knowledge statements, while inductive inference such as data mining [Agrawal et al., 1993] or learning [Fanizzi et al., 2010] discovers (uncertain) knowledge by generalizing or extrapolating facts from initial information.

In real world applications data is dynamic, incomplete, especially when exposed through sensors [Labrinidis and Jagadish, 2012]. Tracking phenomena with multiple sensor readings is a challenging problem. From traffic diagnosis [Lécué, 2012], systems monitoring [Song et al., 2014] to disease transmission prediction [Sadilek et al., 2012], all are examples of scenarios where consistent knowledge needs to be derived from dynamic, incomplete data. Reasoning is strongly restricted by dynamics, variance and noisiness of data, enforcing knowledge to be regularly revisited [Anicic et al., 2011]. Recent works in stream reasoning [Valle et al., 2009] handle the dynamics of semantic querying but are very limited in reasoning, and fail in recovering knowledge from data. Although [Lécué and Pan, 2013] exposed benefits in combining reasoning and learning, the shortcomings are lack of scalability, over-specification of rules, non-integration to reasoning systems. On the other hand inductive learning is limited by the large fluctuation of rules abstraction and its (high) theoretical complexity. Learning is also heavily biased by data incompleteness and inconsistency, making knowledge subject to incorrectness [Dietterich and Michalski, 1981]. They follow classic (raw) data mining techniques i.e., identification of patterns using distance metrics [Ramaswamy et al., 2000] between syntactic values. [Srikant and Agrawal, 1995] capture such patterns through frequent itemset mining [Agrawal et al., 1996]. Their approach consists in discovering recurring sequences of syntactic items and potential implication rules among those items e.g., rules "*buying milk* implies *buying bread* with a confidence of 70%" are learnt in market basket analysis. [Dehaspe and Raedt, 1997] constrain all rules to be represented following inductive logic programming (ILP), which significantly improve scalability. Although different metrics have been introduced to measure the *quality* of the derived rules (e.g., confidence), previous approaches fail in deriving scalable, accurate and consistent knowledge.

We address the problem of *"discovering consistent knowledge from dynamic semantic data"*. Given some continuous knowledge, how do we capture a minimal albeit representative set of time-evolving trends to discover accurate, consistent knowledge? The semantics (meaning) of data and its information is captured through ontologies. Ontologies are characterized by using rich logic-based languages, e.g., *Web Ontology Language* OWL, which is underpinned by Description Logics (DL) [Baader and Nutt, 2003]. Key contributions include: (1) We design the first algorithm to learn DL rules, which are strictly more expressive than DL axioms and Datalog rules. (2) By exploiting the expressiveness of DL rules, we introduce the notions of significative, representative association DL rules, enabling precise identification of fundamental rules. The benefits of combining learning and reasoning, validated through experiments with data from Dublin, are: (i) logical representation of learnt rules (n-ary predicates, shared variables), (ii) classification, abstraction of rules, (iii) tight integration of rules in reasoning; (iv) scalability, (v) accuracy of consistent knowledge discovery.

Next section reviews the adopted logic and rule representation together with dynamic ontology (knowledge). Then we present inductive learning in dynamic ontologies. The next section presents how representative rules drive consistent knowledge discovery. Finally, we report experiments with real data from Dublin City and draw some conclusions.

2 Background

Evolving and static background knowledge are represented using an ontology. We focus on DL to define ontologies since this logic offers good reasoning support for most of its expressive families and compatibility to W3C standards e.g., OWL 2. We illustrate our work with DL $\mathcal{EL}^{++}$ [Baader et al., 2005] where satisfiability, subsumption are decidable. The selection of this DL fragment, which is the logic behind the basis of many more expressive DL, has been guided by (i) the semantics expressed by data in our application cf. Experimental Results, lessons learned, (ii) its polynomial time reasoning when combined with $\mathcal{EL}^{++}$ rules. We review (i) DL basics of $\mathcal{EL}^{++}$, (ii) $\mathcal{EL}^{++}$ atomsets and rules, (iii) evolving ontologies and the underlying reasoning.

2.1 Description Logics $\mathcal{EL}^{++}$

A signature Σ, noted $(\mathcal{N}_C, \mathcal{N}_R, \mathcal{N}_I)$ consists of 3 disjoint sets of (i) atomic concepts $\mathcal{N}_C$, (ii) atomic roles $\mathcal{N}_R$, and (iii) individuals $\mathcal{N}_I$. Given a signature, the top concept $\top$, the bottom concept $\bot$, an atomic concept A, an individual a, an atomic role expression r, $\mathcal{EL}^{++}$ concept expressions C and D in $\mathcal{C}$ can be composed with the following constructs:

$$\top \mid \bot \mid A \mid C \sqcap D \mid \exists r.C \mid \{a\}$$

The DL ontology $\mathcal{O} \doteq \langle \mathcal{T}, \mathcal{A} \rangle$ is composed of a TBox $\mathcal{T}$, and an ABox $\mathcal{A}$. A TBox is a set of concept and role axioms. $\mathcal{EL}^{++}$ supports General Concept Inclusion axioms (GCIs e.g. $C \sqsubseteq D$), Role Inclusion axioms (RIs e.g., $r \sqsubseteq s$). An ABox is a set of concept assertion axioms e.g., $C(a)$, role assertion axioms e.g., $R(a, b)$, individual in/equality axioms e.g., $a \neq b$, $a = b$. In this paper, we assume acyclic TBoxes which entail finitely instance statements.

Example 1. ***(TBox and ABox Concept Assertion Axioms)***
Figure 1 presents (i) a TBox $\mathcal{T}$ where $DisruptedRoad$ (4) denotes the concept of "roads which are adjacent to an event causing high disruption", (ii) concept assertions (15-17) denoting the individual r_0 having $r_{i,1 \leq i \leq 3}$ as adjunct roads.

Table 1 sketches some completion rules [Baader et al., 2005] that are used to classify $\mathcal{EL}^{++}$ TBox $\mathcal{T}$ and entail subsumption. Reasoning with such rules is PTime-Complete [Baader et al., 2008].

2.2 $\mathcal{EL}^{++}$ Atom, Atomsets, Binding and Rule

We consider $\mathcal{EL}^{++}$ with (i) concept expressions $\mathcal{C}$, role names $\mathcal{N}_R$, individual names $\mathcal{N}_I$, and (ii) a countable set of first-order variables $\mathcal{V}$.
Atom, Atomset: Given terms $x_1, x_2 \in \mathcal{V} \cup \mathcal{N}_I$, a concept (role) atom is a formula $C(x_1)$ ($R(x_1, x_2)$) with $C \in \mathcal{C}$ ($R \in \mathcal{N}_R$). To simplify the notation, we use finite sets (called *atomsets*) $\mathbb{B}$ of (concepts, roles) atoms for representing the conjunction $\forall \vec{x}. \bigwedge B$ where $\vec{x} \doteq x_1, \cdots, x_n \in \mathcal{V}$ are variables of atoms $B \in \mathbb{B}$. We assume that atoms in atomsets are inter-connected i.e., variables are shared.
Atomset Binding: Atomsets can be seen as conjunctive queries [Glimm et al., 2007] without non-distinguished variables. The arity of an atomset is the number of variables in an atomset. We write $\mathcal{T}, \mathcal{A} \models \mathbb{B}[\vec{a}]$ to denote that $\vec{a} \in \mathcal{N}_I$ is an answer of query $\mathbb{B}$. In other words the variables $\vec{x}$ of atomset $\mathbb{B}$ are bound (mapped) by $\vec{a}$. In the rest of the paper we used the terms *answer* and *binding* interchangeably. $bind(\mathbb{B}, \mathcal{T} \cup \mathcal{A})$ is the set of all bindings to $\mathbb{B}$ w.r.t. $\mathcal{T}$, $\mathcal{A}$.

Example 2. ***(Atomset & Binding w.r.t. $\mathcal{O}$ in Figure 1)***
Given atomset $\mathbb{B} \doteq \{adj(x, r_1)\}$, $bind(\mathbb{B}, \mathcal{O})$ is $\{(r_0)\}$.

$$SocialEvent \sqcap \exists type.Music \sqsubseteq Event \sqcap \exists disruption.Steady \quad (1)$$

$$SocialEvent \sqsubseteq Event \quad (2)$$

$$Incident \sqcap \exists impact.Serious \sqsubseteq Event \sqcap \exists disruption.High \quad (3)$$

$$Road \sqcap \exists adj.(\exists occur.(\exists disruption.High)) \sqsubseteq DisruptedRoad \quad (4)$$

$$Road \sqcap \exists with.Bus \sqsubseteq BusRoad \quad (5)$$

$$BusRoad \sqcap \exists travel.Long \sqsubseteq Road \sqcap \exists with.CongestedBus \quad (6)$$

$$Steady \sqsubseteq High \quad (7)$$

$$Stop \sqsubseteq Long \sqsubseteq Abnormal \quad (8)$$

$$Bus(b_1) \quad (9)$$

$$Bus(b_2) \quad (10)$$

$$Bus(b_3) \quad (11)$$

$$Road(r_1) \quad (12)$$

$$Road(r_2) \quad (13)$$

$$Road(r_3) \quad (14)$$

$$adj(r_0, r_1) \quad (15)$$

$$adj(r_0, r_2) \quad (16)$$

$$adj(r_0, r_3) \quad (17)$$

Figure 1: $\mathcal{O} \doteq \langle \mathcal{T}, \mathcal{A} \rangle$. Sample of TBox $\mathcal{T}$ and ABox $\mathcal{A}$.

Atomset Containment: Let $\mathcal{T}$ be a TBox, $\mathbb{B}, \mathbb{C}$ atomsets with the same arity. Then $\mathbb{B}$ is contained in $\mathbb{C}$ w.r.t. $\mathcal{T}$, written $\mathbb{B} \subseteq_{\mathcal{T}} \mathbb{C}$, if for all consistent ABoxes $\mathcal{A}$ w.r.t. $\mathcal{T}$, we have $bind(\mathbb{B}, \mathcal{T} \cup \mathcal{A}) \subseteq bind(\mathbb{C}, \mathcal{T} \cup \mathcal{A})$.

R_1	If $X \sqsubseteq A,\ A \sqsubseteq B$ then $X \sqsubseteq B$
R_2	If $X \sqsubseteq A_1, \cdots A_n,\ A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B$ then $X \sqsubseteq B$
R_3	If $X \sqsubseteq A,\ A \sqsubseteq \exists r.B$ then $X \sqsubseteq \exists r.B$
R_4	If $X \sqsubseteq \exists r.A,\ A \sqsubseteq A',\ \exists r.A' \sqsubseteq B$ then $X \sqsubseteq B$
R_5	If $X \sqsubseteq \exists r.A,\ A \sqsubseteq \bot$ then $X \sqsubseteq \bot$
R_6	If $X \sqsubseteq \exists r.A,\ r \sqsubseteq s$ then $X \sqsubseteq \exists s.A$
R_7	If $X \sqsubseteq \exists r_1.A,\ A \sqsubseteq \exists r_2.B,\ r_1 \circ r_2 \sqsubseteq r_3$ then $X \sqsubseteq \exists r_3.B$

Table 1: $\mathcal{EL}^{++}$ TBox Completion Rules (no datatypes).

Example 3. ***(Atomset Containment w.r.t. $\mathcal{O}$ in Figure 1)***
Let $\mathbb{B}$, $\mathbb{C}$ be atomsets $\{(18)\}$ and $\{(19)\}$; $\mathcal{A}'$ be $\{Event(e_1), Event(e_2), SocialEvent(e_1)\}$. $\mathbb{B} \subseteq_{\mathcal{T}} \mathbb{C}$ for $\mathcal{A}'$ since all bindings (answers) of $\mathbb{B}$ are also bindings of $\mathbb{C}$.

$$SocialEvent(x) \quad (18) \qquad\qquad Event(x) \quad (19)$$

$\mathcal{EL}^{++}$ Rules: $\mathcal{EL}^{++}$ rules [Krötzsch et al., 2008] extends $\mathcal{EL}^{++}$ expressivity while maintaining polynomial time complexity of many typical inference problems. Given atomsets $\mathbb{B}$, $\mathbb{H}$, and all variables $\vec{x} \in \mathcal{V}$ of atomset $\mathbb{B} \cup \mathbb{H}$, an $\mathcal{EL}^{++}$ *rule* is a formula $\mathbb{B} \twoheadrightarrow \mathbb{H}$, such that $\mathbb{B}$ is cycle free and does not contain atom of the form $R(x, x)$.

Example 4. ***($\mathcal{EL}^{++}$ Rule)***
Below rule denotes "if x_3 is adjacent to a x_2 where a highly disruptive event x_1 occurs then buses are congested in x_3". $\{(22)\}$ is atomset $\{(Road \sqcap \exists with.CongestedBus)(x_3)\}$.

$$(Event \sqcap \exists disruption.High)(x_1) \wedge \quad (20)$$
$$occur(x_2, x_1) \ \wedge \ adj(x_3, x_2) \quad (21)$$
$$\twoheadrightarrow (Road \sqcap \exists with.CongestedBus)(x_3) \quad (22)$$

2.3 Dynamics of Knowledge as Evolving Ontologies

We represent dynamics of knowledge by an evolution of ontologies in Definition 1 [Huang and Stuckenschmidt, 2005].

Definition 1. ***(DL $\mathcal{L}$ Evolving Ontology)***
A DL $\mathcal{L}$ evolving ontology $\mathcal{P}_m^n$ from point of time m to point of time n is a sequence of (sets of) Abox axioms $(\mathcal{P}_m^n(m), \mathcal{P}_m^n(m+1), \cdots, \mathcal{P}_m^n(n))$ w.r.t a static TBox $\mathcal{T}$ in a DL $\mathcal{L}$ where $m, n \in \mathbb{N}$ and $m < n$.

$\mathcal{P}_m^n(i)$ is a snapshot of an evolving ontology $\mathcal{P}_m^n$ at time i, referring to ABox axioms with respect to a TBox in $\mathcal{L}$. We will consider evolving ontologies $\mathcal{P}_0^n$ for the sake of clarity.

Example 5. ***(DL $\mathcal{EL}^{++}$ Evolving Ontology)***
Figure 2 illustrates $\mathcal{EL}^{++}$ evolving ontologies $\mathcal{P}_0^9$, $\mathcal{Q}_0^9$, $\mathcal{R}_0^9$, related to events, travel time, buses, through snapshots at time $i \in \{5, 6, 7\}$. Their dynamic knowledge is captured by evolving ABox axioms e.g., (26) captures e_2 as "a social music event occurring in r_2" at time 6 of $\mathcal{P}_0^9$.

By applying completion rules in Table 1 on background static knowledge $\mathcal{T}$ and evovling ontology $\mathcal{P}_0^n$, we infer axioms which are specific to some snapshots.

$$\mathcal{P}_0^9(5) : (Event \sqcap \exists disruption.High)(e_1), \quad occur(r_1, e_1) \tag{23}$$

$$\mathcal{Q}_0^9(5) : (Road \sqcap \exists travel.Long)(r_1) \tag{24}$$

$$\mathcal{R}_0^9(5) : with(r_1, b_1) \tag{25}$$

$$\mathcal{P}_0^9(6) : (SocialEvent \sqcap \exists type.Music)(e_2), \quad occur(r_2, e_2) \tag{26}$$

$$\mathcal{Q}_0^9(6) : (Road \sqcap \exists travel.Abnormal)(r_2) \tag{27}$$

$$\mathcal{R}_0^9(6) : with(r_2, b_2) \tag{28}$$

$$\mathcal{P}_0^9(7) : (Incident \sqcap \exists impact.Serious)(e_3), \quad occur(r_3, e_3) \tag{29}$$

$$\mathcal{Q}_0^9(7) : (Road \sqcap \exists travel.Stop)(r_3) \tag{30}$$

$$\mathcal{R}_0^9(7) : with(r_3, b_3) \tag{31}$$

Figure 2: Evolving Ontologies $\mathcal{P}_0^9(i), \mathcal{Q}_0^9(i), \mathcal{R}_0^9(i)_{i \in \{5,6,7\}}$.

Example 6. ***(Reasoning in Evolving Ontology)***
(32), (33), *as dynamic knowledge are entailed from axioms of $\mathcal{O}$ in Figure 1 and evolving ontologies $\mathcal{P}_0^9$, $\mathcal{Q}_0^9$, $\mathcal{R}_0^9$ in Figure 2 (cf. references to axioms A in $\models^{(A)}$), by applying completion rules in Table 1 (cf. references to rules R in $\models_R$). E.g., r_0 and r_3 are respectively entailed to be roads with: (i) disruptions, (ii) some congested buses, both at time 7.*

$$\mathcal{O}, \mathcal{P}_0^9(7) \models^{\text{(3-4)},(16),(29)}_{R_1,R_2,R_3,R_7} DisruptedRoad(r_0) \tag{32}$$

$$\mathcal{O}, (\mathcal{Q}_0^9 \cup \mathcal{R}_0^9)(7) \models^{\text{(5-6)},(8),\text{(30-31)}}_{R_1,R_2,R_3,R_4} (\exists with.CongestedBus)(r_3) \tag{33}$$

ABox axioms (30) in $\mathcal{Q}_0^9$, (31) in $\mathcal{R}_0^9$ are both required to fire GCIs (5-6) and entail (33). We say that (33) emphasizes an "*association*" ($\twoheadrightarrow$ in Figure 3) of $\mathcal{Q}_0^9$, $\mathcal{R}_0^9$ through (5-6) in $\mathcal{T}$. Thus dynamic knowledge can be entailed by axioms from single (32) but also "*associated*" (33) evolving ontologies.

3 Inductive Reasoning in Evolving Ontologies

Axioms which enable knowledge association e.g., (5-6) are rarely modeled a priori in a background knowledge $\mathcal{T}$ because of the uncertainty of evolving ontologies. An association of $\mathcal{P}_0^9$ (events) with $\mathcal{Q}_0^9$ (travel time) or $\mathcal{R}_0^9$ (buses information) cannot be derived a priori but only a posteriori by analyzing data. Indeed (5-6) are axioms specific to some cities, where knowledge has been gained from data analysis.

Capturing such "*rules*" (e.g., dashed $\twoheadrightarrow$ in Figure 3) in $\mathcal{T}$ would certainly extend the reasoning impact since they capture evolving knowledge. They could be used for inducing missing knowledge in $\mathcal{Q}_0^9$ or $\mathcal{R}_0^9$. *How to discover knowledge association across evolving ontologies $\mathcal{P}$ and $\mathcal{R}$ or $\mathcal{Q}$ at time 7 with respect to $\mathcal{T}$?* We tackle this problem by mining knowledge associations as $\mathcal{EL}^{++}$ rules.

3.1 Association $\mathcal{EL}^{++}$ Rules

We revisit the concept of association rules [Agrawal et al., 1993] in the context of evolving ontologies through an $\mathcal{EL}^{++}$ rules based representation.

Definition 2. ***(Association $\mathcal{EL}^{++}$ Rule)***
Let $\langle \mathcal{T}, \mathcal{A} \rangle$ be $\mathcal{EL}^{++}$ axioms, $\mathcal{P}_0^n$, $\mathcal{Q}_0^n$ be $\mathcal{EL}^{++}$ evolving ontologies, $\mathbb{B}$, $\mathbb{H}$ be atomsets where variables in $\mathbb{H}$, or $var(\mathbb{H})$, is a subset of the variables in $\mathbb{B}$. An association $\mathcal{EL}^{++}$ rule in $\mathcal{P}_0^n \times \mathcal{Q}_0^n$ is a $\mathcal{EL}^{++}$ rule $\mathbb{B} \twoheadrightarrow \mathbb{H}$ such that $\exists i \in [0, n]$: $bind(\mathbb{B}, \mathcal{T} \cup \mathcal{A} \cup \mathcal{P}_0^n(i))|_{var(\mathbb{H})} = bind(\mathbb{H}, \mathcal{T} \cup \mathcal{A} \cup \mathcal{Q}_0^n(i))$.

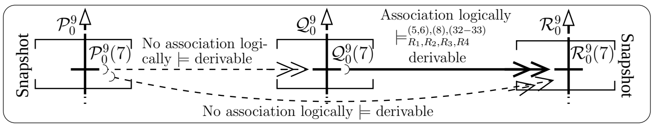

Figure 3: Illustration of an Association ($\twoheadrightarrow$) of Evolving Ontologies $\mathcal{Q}$ and $\mathcal{R}$ at time 7 with Respect to $\mathcal{T}$. Open Challenge: How to Discover such Associations?

Definition 2 identifies an association as an $\mathcal{EL}^{++}$ rule between atomsets $\mathbb{B}$, $\mathbb{H}$ if they have the same bindings for their shared variables across two evolving ontologies at a time i. Contrary to [Lécué and Pan, 2013], representing association as an arbitrary combination of ABox axioms, we consider $\mathcal{EL}^{++}$ rule as a broader framework. Indeed it supports (i) associations which can be modeled by $\mathcal{EL}^{++}$ rules, (ii) more generic rules since variables are accepted in atoms of $\mathcal{EL}^{++}$ rules, instead of only allowing constants in the other case, (iii) native combination of $\mathcal{EL}^{++}$ rule and axioms.

Example 7. ***(Association $\mathcal{EL}^{++}$ Rule)***
The rule (20-22) in Example 4 illustrates an association $\mathcal{EL}^{++}$ rule from $\mathcal{P}_0^9$ to $\mathcal{Q}_0^9 \cup \mathcal{R}_0^9$. The atomset of this rule is bound at times 5 by $\{(e_1, r_0, r_1)\}$ and 7 by $\{(e_3, r_0, r_3)\}$.

We measure the interestingness of association rules by adapting the concepts of *support* (Definition 3) and *confidence* (Definition 5) introduced in the database community.

Definition 3. ***(Atomset Support)***
Given axioms $\mathcal{O} \doteq \langle \mathcal{T}, \mathcal{A} \rangle$, an evolving ontology $\mathcal{P}_0^n$, atomset $\mathbb{B}$, the support of $\mathbb{B}$, noted $\sigma(\mathbb{B})$, in $[0, 1]$ is defined by:

$$\sigma(\mathbb{B}) \doteq \frac{|\{i \in [0, n] \mid \exists \vec{a} \in \mathcal{N}_I : \mathcal{O}, \mathcal{P}_0^n(i) \models \mathbb{B}[\vec{a}]\}|}{n+1} \quad (34)$$

where the expression $|S|$ refers to the cardinality of S.

The support of atomset $\mathbb{B}$ is the proportion of snapshots where $\mathbb{B}$ has at least one binding $\vec{a}$ in $\mathcal{A} \cup \mathcal{P}_0^n$ with respect to $\mathcal{T}$. Since $\mathbb{B}$ may have multiple bindings in $\mathcal{A} \cup \mathcal{P}_0^n$ at any time $i \in [0, n]$, we capture their number in Definition 4.

Definition 4. ***(Atomset Weight)***
Given $\mathcal{EL}^{++}$ axioms $\mathcal{O} \doteq \langle \mathcal{T}, \mathcal{A} \rangle$, evolving ontology $\mathcal{P}_0^n$ and atomset $\mathbb{B}$, the weight of $\mathbb{B}$, noted $\omega(\mathbb{B})$, is defined by:

$$\omega(\mathbb{B}) \doteq \sum_{i=0}^{n} |\{\vec{a} \in \mathcal{N}_I \mid \mathcal{O}, \mathcal{P}_0^n(i) \models \mathbb{B}[\vec{a}]\}| \tag{35}$$

Example 8.*(Atomsets Support and Weight)*
Let $\mathcal{P}_5^7$, $\mathcal{Q}_5^7$, $\mathcal{R}_5^7$ be $\mathcal{P}_0^9$, $\mathcal{Q}_0^9$, $\mathcal{R}_0^9$ restricted to $[5, 7]$ where ABox statements related to e_4 extends $\mathcal{P}_5^7$ at time 7 cf. Table 2. This table illustrates the support σ, weight ω of atomsets e.g., $\{(22)\}$ is bound in $\mathcal{Q}_5^7 \cup \mathcal{R}_5^7$ at times 5 by $\{(r_1)\}$ and 7 by $\{(r_3)\}$, but not at time 6, thus $\sigma(\{(22)\})$ is $^2/_3$. The number of bindings of $\{(22)\}$, noted $\omega(\{(22)\})$, is 2 in $\mathcal{Q}_5^7 \cup \mathcal{R}_5^7$.

Definition 5. ***(Confidence of an Association $\mathcal{EL}^{++}$ Rule)***
Let $\mathbb{B} \twoheadrightarrow \mathbb{H}$ be an association $\mathcal{EL}^{++}$ rule in $\mathcal{P}_0^n \times \mathcal{Q}_0^n$. The confidence γ of $\mathbb{B} \twoheadrightarrow \mathbb{H}$ in $[0, 1]^2$ is:

$$\gamma(\mathbb{B} \twoheadrightarrow \mathbb{H}) \doteq \left(\frac{\sigma(\mathbb{B} \cup \mathbb{H})}{\sigma(\mathbb{B})}, \frac{\omega(\mathbb{B} \cup \mathbb{H})}{\omega(\mathbb{B})} \right) \tag{36}$$

$\sigma(\mathbb{B} \cup \mathbb{H})$, $\omega(\mathbb{B} \cup \mathbb{H})$ are support and weight of $\mathbb{B} \twoheadrightarrow \mathbb{H}$ i.e., respectively: the proportion of snapshots in $\mathcal{P}_0^n \cup \mathcal{Q}_0^n$ where $\mathbb{B} \cup \mathbb{H}$ has at least one binding, and its number of bindings.

The confidence is defined as the percentage of (i) snapshots in $\mathcal{P}_0^n \cup \mathcal{Q}_0^n$ where $\mathbb{B} \cup \mathbb{H}$ has at least an binding with regard to those where $\mathbb{B}$ has an binding, and (ii) bindings of $\mathbb{B} \cup \mathbb{H}$ in $\mathcal{P}_0^n \cup \mathcal{Q}_0^n$ with regard to those which bind $\mathbb{B}$. That is, they represent complementary conditional probabilities:

$$p(\mathcal{O}, \mathcal{Q}_0^n(i) \models \mathbb{H}[\vec{a}] \mid \mathcal{O}, \mathcal{P}_0^n(i) \models \mathbb{B}[\vec{a}]|_{var(\mathbb{H})}) \tag{37}$$

evaluated with respect to the number of (i) snapshots and (ii) bindings i.e., respectively σ- and ω-related entry of (36).

Ontology		$\mathcal{P}_5^7$		$\mathcal{Q}_5^7$		$\mathcal{R}_5^7$	$\mathcal{Q}_5^7 \cup \mathcal{R}_5^7$
Atomset $\mathbb{B}$		$\{(20)\}$	$\{(21)\}$	$\{(27)\}^*$	$\{(30)\}^*$	$\{(31)\}^*$	$\{(22)\}$
Variable $\mathcal{V}$		(x_1)	(x_1, x_2, x_3)	(x_3)	(x_3)	(x_4, x_3)	(x_3)
Binding at Time	5	$\{(e_1)\}$	$\{(e_1, r_1, r_0)\}$	$\{(r_1)\}$		$\{(b_1, r_1)\}$	$\{(r_1)\}$
	6	$\{(e_2)\}$	$\{(e_2, r_2, r_0)\}$	$\{(r_2)\}$		$\{(b_2, r_2)\}$	
	7	$\{(e_3), (e_4)\}$	$\{(e_3, r_3, r_0), (e_4, r_3, r_0)\}$	$\{(r_3)\}$	$\{(r_3)\}$	$\{(b_3, r_3)\}$	$\{(r_3)\}$
$\sigma(\mathbb{B})$		1	1	1	$^1/_3$	1	$^2/_3$
$\omega(\mathbb{B})$		4	4	3	1	3	2

[*] From now on all terms $\mathcal{N}_I$ of (27), (30-31) are variables universally quantified in $\mathcal{V}$.

Table 2: Binding, Support and Weight of Atomsets $\mathbb{B}$.

Example 9. ***(Confidence of an Association $\mathcal{EL}^{++}$ Rule)***
Confidence $\gamma(\mathbb{B} \twoheadrightarrow \mathbb{H})$ with $\mathbb{B}$:$\{(20), (21)\}$, $\mathbb{H}$:$\{(22)\}$ is:

$$\left(\frac{\sigma(\{(20), (21), (22)\})}{\sigma(\{(20), (21)\})}, \frac{\omega(\{(20), (21), (22)\})}{\omega(\{(20), (21)\})} \right) \textit{ i.e., } \left(\frac{^2/_3}{^3/_3}, \frac{3}{4} \right)$$

$\mathbb{B} \twoheadrightarrow \mathbb{H}$ *is correct in $^2/_3$ of time $[5, 7]$ and its atomset $\mathbb{B} \cup \mathbb{H}$ is bound by $^3/_4$ of bindings of atomset* $\mathbb{B}$: $\{(20), (21)\}$.

Remark 1. ***(Rule Confidence and Ordering)***
The level of confidence of rules can be compared by analyzing their supports and weights since confidence is defined as a tuple (Definition 5) e.g., $\gamma(\mathbb{B}_1 \twoheadrightarrow \mathbb{H}_1) > \gamma(\mathbb{B}_2 \twoheadrightarrow \mathbb{H}_2)$ if

$$\frac{\sigma(\mathbb{B}_1 \cup \mathbb{H}_1)}{\sigma(\mathbb{B}_1)} > \frac{\sigma(\mathbb{B}_2 \cup \mathbb{H}_2)}{\sigma(\mathbb{B}_2)} \qquad \frac{\omega(\mathbb{B}_1 \cup \mathbb{H}_1)}{\omega(\mathbb{B}_1)} > \frac{\omega(\mathbb{B}_2 \cup \mathbb{H}_2)}{\omega(\mathbb{B}_2)}$$

Alternatively, in case of conflicts e.g., the value of the first element of $\gamma(\mathbb{B}_1 \twoheadrightarrow \mathbb{H}_1)$ is better than the first element of $\gamma(\mathbb{B}_2 \twoheadrightarrow \mathbb{H}_2)$ but worse for the second element, we compare a weighted average of their normalised components.

3.2 Mining Association $\mathcal{EL}^{++}$ Rules

The problem of mining association $\mathcal{EL}^{++}$ rules is to generate all its rules with a minimum support $\sigma_{\min}$, confidence $\gamma_{\min}$, weight $\omega_{\min}$. By revisiting [Agrawal et al., 1993], it can be decomposed by: (i) finding all significant atomsets i.e., atomsets with support, weight above $\sigma_{\min}, \omega_{\min}$, (ii) using them to generate rules that meet $\gamma_{\min}$.

Algorithm 1 revisits `WARMR` [Dehaspe and Raedt, 1997] to support $\mathcal{EL}^{++}$ atomsets bindings, containment and weight. It finds all significant atomsets by exploiting the lattice structure the containment relation $\subseteq_{\mathcal{T}}$ imposes on the space of atomsets to perform a breadth-first search. $\mathcal{S}_k$ refers to the set of significant atomsets mined at depth k in the lattice while $\mathcal{C}_k$ is the set of potential candidates for $\mathcal{S}_k$. Their elements are called k-atomsets i.e., atomsets of k atoms. Initially (line 4) the set of significant 1-atomsets $\mathcal{S}_1$ is determined. A subsequent pass (line 5), say pass k, consists of two phases. First (line 6), the significant atomsets $\mathcal{S}_{k-1}$ found in the $(k-1)^{th}$ pass are used to generate the candidate atomsets $\mathcal{C}_k$, using Algorithm 2. $\mathcal{P}_0^n$ is then analyzed (line 7) to determine support (line 10) and weight (line 11) of candidates $\mathbb{C} \in \mathcal{C}_k$, when bound in $\mathcal{P}_0^n$ (line 9).

Algorithm 1: `atomsets-mining`$\langle \mathcal{O}_0^n, \sigma_{\min}, \omega_{\min} \rangle$.

1 **Input**: $\mathcal{EL}^{++}$ axioms $\mathcal{O} \doteq \langle \mathcal{T}, \mathcal{A} \rangle$; $\mathcal{EL}^{++}$ evolving ontology $\mathcal{P}_0^n$; Min. threshold of support $\sigma_{\min}$ and weight $\omega_{\min}$.
2 **Result**: Set of significant atomsets.
3 **begin**
4 $\quad \mathcal{S}_1 \leftarrow$ Set of significant 1-atomsets; *% Initialization*
5 $\quad$ **foreach** $k \geq 2 \mid \mathcal{S}_{k-1} \neq \emptyset$ **do** *% k-atomsets on top of k-1 ones*
6 $\quad\quad \mathcal{C}_k \leftarrow$ *atomsets-gen*$\langle \mathcal{S}_{k-1} \rangle$; *% k-atomsets Candidates*
7 $\quad\quad$ **foreach** *point of time* $i \in [0, n]$ **do** *% Snapshot* $\mathcal{P}_0^n(i)$
8 $\quad\quad\quad$ *% Atomsets with bindings* $\vec{a}$ *in* $\mathcal{P}_0^n(i)$
9 $\quad\quad\quad$ **foreach** $\mathbb{C} \in \mathcal{C}_k \mid \exists \vec{a} : \mathcal{T}, \mathcal{A} \cup \mathcal{P}_0^n(i) \models \mathbb{C}[\vec{a}]$ **do**
10 $\quad\quad\quad\quad$ *count*$(\mathbb{C}) \leftarrow$ *count*$(\mathbb{C}) + 1$; *% Needed for* $\sigma(\mathbb{C})$
11 $\quad\quad\quad\quad \omega(\mathbb{C}) \leftarrow \omega(\mathbb{C}) + \left|\{\vec{a} \mid \mathcal{T}, \mathcal{A} \cup \mathcal{P}_0^n(i) \models \mathbb{C}[\vec{a}]\}\right|$;
12 $\quad\quad$ *% Only significant atomsets are considered*
13 $\quad\quad \mathcal{S}_k \leftarrow \{\mathbb{C} \in \mathcal{C}_k \mid \frac{count(\mathbb{C})}{(n+1)} \geq \sigma_{\min},\ \omega(\mathbb{C}) \geq \omega_{\min}\}$;
14 $\quad$ **return** $\bigcup_k \mathcal{S}_k$;

Algorithm 2 generates candidate $(k+1)$-atomsets from significant k-atomsets. In join-step (lines 5-7) where atomsets are sorted in their lexicographic order ($\preceq_{lex}$), $\mathcal{S}_k$ is combined with itself on the basis of its k common atoms. In the prune-step, we discard all atomsets that have a subset which does meet $\sigma_{\min}, \omega_{\min}$ i.e., not in $\mathcal{S}_k$ (line 10). It is trivial to show that any atomset in $\mathcal{C}_{k+1}$ is significant only if all subsets of size k are also significant by revisiting the results of [Agrawal et al., 1996] for $\mathcal{EL}^{++}$ atomsets. Thus line 10 maintains completeness. We also prune $(k+1)$-atomsets which are redundant i.e., already contain(ed by) $\mathcal{S}_k$ (line11).

Algorithm 2: Atomsets Generation: `atomsets-gen`$\langle \mathcal{S}_k \rangle$

1 **Input**: A terminology $\mathcal{T}$; Set of $\mathcal{S}$ignificant k-atomsets $\mathcal{S}_k$.
2 **Result**: Set of $\mathcal{C}$andidate $(k+1)$-atomsets $\mathcal{C}_{k+1}$.
3 **begin**
4 *% Join-step of k-atomsets from the same level k to obtain* $\mathcal{C}_{k+1}$
5 $\mathcal{C}'_{k+1} \leftarrow \{\{S_1, ..., S_k, T_k\} \mid \{S_1, ..., S_{k-1}, S_k\} \in \mathcal{S}_k,$
6 $\{S_1, ..., S_{k-1}, T_k\} \in \mathcal{S}_k,$
7 $S_k \preceq_{lex} T_k\}$
8 *% Prune-step of* $(k+1)$*-atomsets*
9 $\mathcal{C}_{k+1} \leftarrow \mathcal{C}'_{k+1} \setminus \{ \{S_1, ..., S_k, T_k\} \mid$
10 (i) $\exists \mathbb{S} \subseteq \{S_1, ..., S_k, T_k\}$ with $|\mathbb{S}| = k$ such that $\mathbb{S} \notin \mathcal{S}_k$,
11 (ii) $\{T_k\} \subseteq_{\mathcal{T}} (\supseteq_{\mathcal{T}})\{S_k\}$ } *% Redundant k-atomsets*
12 **return** $\mathcal{C}_{k+1}$;

The complexity of Algorithm 1 is $\Theta(|\mathbb{S}|^2 \times (n+1))$ where $|\mathbb{S}|$ is the number of atoms in $\mathcal{P}_0^n$. We have scalability in the number of snapshots but quadratic dependence on the number of atoms. The prune-step (lines 10-11) in Algorithm 2 controls the exponential growth of the candidate atomsets.

Example 10. **(Atomset Mining - Case $k = 2$)**
Let $\{(20)\}$,$\{(21)\}$,$\{(19)\}$ *be significant* 1*-atomsets. After the join-step of Algorithm 2 we obtain* 2*-atomsets* $\{(20), (21)\}$, $\{(21), (19)\}$, $\{(20), (19)\}$. *The last one is discarded during pruning (line 11) since* $\{(20)\} \subseteq_{\mathcal{T}} \{(19)\}$.

For generating $\mathcal{EL}^{++}$ rules with minimum confidence (Definition 5), we refer to `ap-genrules` [Agrawal et al., 1996]. This procedure for large itemsets, with minor modifications e.g., rules representation (Definition 2), applies to significant atomsets. Its complexity is $\Theta(\max_k |\mathcal{S}_k| \times k)$, where $|\mathcal{S}_k|$ is the number of significant k-atomsets.

4 Knowledge Discovery in Evolving Ontologies

We tackle the problem of discovering knowledge in evolving ontologies by (i) determining its representative DL association rules (Definition 7) and (i) exploiting their combination with background knowledge (Algorithm 3).

4.1 Representative Association $\mathcal{EL}^{++}$ Rules

Although measures of support, weight, confidence largely pruned uninteresting rules, the remaining ones are not necessarily all fundamental to discover new knowledge. Indeed some can be logically derived from a minimal set of rules, which represents the "*representative*" inductive rules. Definition 7 formalizes this concept of representative rule by refining the notion of "*cover*" [Zaki, 2000] in Definition 6.

Definition 6. ***(Association $\mathcal{EL}^{++}$ Rules Cover)***
Let $\mathcal{R}$ *be association* $\mathcal{EL}^{++}$ *rules. The cover of a rule* $\rho : \mathbb{B} \twoheadrightarrow \mathbb{H}$ *in* $\mathcal{R}$ *is defined by* $\Gamma(\rho) \doteq \{\mathbb{A} \twoheadrightarrow \mathbb{H}$ *in* $\mathcal{R} \mid \mathbb{A} \subseteq_{\mathcal{T}} \mathbb{B}\}$.

The rule ρ, covering rules in $\Gamma(\rho)$, synthesizes all rules deriving same consequents as ρ from any contained antecedents. It captures all rules which convey to similar knowledge but requiring more specific knowledge than ρ.

Definition 7. ***(Representative Association $\mathcal{EL}^{++}$ Rules)***
A set of representative association DL rules of $\mathcal{R}$, denoted by $\mathcal{R}^*$, is defined by $\{\rho \in \mathcal{R} \mid \nexists \rho' \in \mathcal{R},\ \rho' \neq \rho,\ \rho \in \Gamma(\rho')\}$.

A set of representative association DL rules is a least set of rules covering (Definition 6) all rules in $\mathcal{R}$. It captures a synthesis set of $\mathcal{R}$, which is required to derive any rule in $\mathcal{R}$. Thus all rules in a cover can be derived from their representative rule by atomset containment.

Proposition 1. ***(Minimum Support, Weight of Rules in $\mathcal{R}^*$)***
$\sigma(\rho^*) \geq \sigma_{min}$ *and* $\omega(\rho^*) \geq \omega_{\min}$ $\forall \rho^* \in \mathcal{R}^*$ *iff* $\exists \rho \in \mathcal{R} \mid$ *(i)* $\sigma(\rho) \geq \sigma_{min}$*, (ii)* $\omega(\rho) \geq \omega_{\min}$ *and (iii)* $\rho \in \Gamma(\rho^*)$.

Proposition 1 states that any representative rule has minimum support, weight if it covers a rule with minimum support, weight. The proof is trivial per Definitions 6 and 7.

Example 11. ***(Rules Cover and Representative Rules)***
Let $\mathcal{R}$ be $\{(19),(21)\} \twoheadrightarrow \{(22)\}$, $\{(18),(21)\} \twoheadrightarrow \{(22)\}$, $\{(20),(21)\} \twoheadrightarrow \{(22)\}$ *in Figure 4. The first rule covers the last two since antecedents* (18), (20) *are contained by* (19) *cf. Examples 3, 10 while their consequent are similar.* $\mathcal{R}^*$ *is* $\{(19),(21)\} \twoheadrightarrow \{(22)\}$*, covering the other two rules.*

The semantics of atomsets is crucial to prune large sets of rules through its representative rules. The more semantic relations (atomsets containment) among evolving ontologies the more (resp. less) covered (resp. representative) rules.

4.2 Consistent Inductive Knowledge Discovery (CIKD)

Algorithm 3 presents our general approach `CIKD` for inducing consistent knowledge from $\mathcal{P}_0^n$ to $\mathcal{Q}_0^n$ at time i.

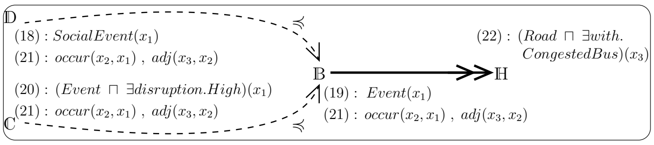

Figure 4: $\mathbb{B} \twoheadrightarrow \mathbb{H}$ as Representative Rule of $\mathbb{C} \twoheadrightarrow \mathbb{H}$, $\mathbb{D} \twoheadrightarrow \mathbb{H}$.

All significant rules $\mathcal{R}$ in $\mathcal{P}_0^n \times \mathcal{Q}_0^n$, which are rules generated from significant atomsets (Algorithms 1, 2) in line 5, are identified by adapting `ap-genrules` [Agrawal et al., 1996] in line 7. Its set of representative rules $\mathcal{R}^*$ is then elaborated (line 8). Proposition 1 ensures minimum support, weight of such rules. All representative rules $\rho : \mathbb{B} \twoheadrightarrow \mathbb{H}$, filtered by confidence (line 10), are evaluated at time i of $\mathcal{P}_0^n$, $\mathcal{Q}_0^n$ (line 13). In details we checked the consistency of $\mathcal{Q}_0^n(i)$ with the knowledge $\mathbb{H}$ derived from ρ, especially for all bindings $\vec{a}$ of $\mathbb{B}$ with respect to $\mathcal{P}_0^n(i)$. This ensures that only consistent knowledge $\mathbb{H}(\vec{a})$ could be derived from $\mathbb{B}(\vec{a})$ and added to $\mathcal{Q}_0^n(i)$. This condition, interpreted as $\mathcal{T} \cup (\mathcal{P}_0^n \cup \mathcal{Q}_0^n)(i) \cup \{\rho\} \not\models \perp$, validates the consistent combination TBox, ABox, rules axioms at time i of $\mathcal{P}_0^n$, $\mathcal{Q}_0^n$.

Its complexity is polynomial since (i) the representative rules generation (lines 4-8) is polynomial in the number of snapshots, atoms (line 5 i.e., Algorithms 1,2), k-atomsets (line 7), rules (Definition 7); (ii) their consistency checking (line 13), together with (iii) atomset binding, containment are polynomial in $\mathcal{EL}^{++}$ [Bienvenu et al., 2012].

Algorithm 3: $\texttt{CIKD}\langle \mathcal{T}, \mathcal{P}_0^n, \mathcal{Q}_0^n, i, \sigma_{\min}, \omega_{\min}, \gamma_{\min}\rangle$

Input: Terminology $\mathcal{T}$; Evolving Ontologies $\mathcal{P}_0^n$, $\mathcal{Q}_0^n$; Time i; Min. support $\sigma_{\min}$, weight $\omega_{\min}$, confidence $\gamma_{\min}$.
Result: Consistent knowledge $\mathcal{Q}_0^n(i)$ induced from $\mathcal{P}_0^n \times \mathcal{Q}_0^n$.
begin
  % $\mathcal{S}$: Set of significant atomsets in evolving ontology $\mathcal{P}_0^n \cup \mathcal{Q}_0^n$
  $\mathcal{S} \leftarrow \texttt{atomsets-mining}\langle \mathcal{P}_0^n \cup \mathcal{Q}_0^n, \sigma_{\min}, \omega_{\min}\rangle$;
  % $\mathcal{R}$: Set of rules in $\mathcal{P}_0^n \times \mathcal{Q}_0^n$ with minimum confidence $\gamma_{\min}$
  $\mathcal{R} \leftarrow \texttt{ap-genrules}\langle \mathcal{S}, \mathcal{P}_0^n \times \mathcal{Q}_0^n, \gamma_{\min}\rangle$; *% Version adapted*
  $\mathcal{R}^* \leftarrow \{\rho \in \mathcal{R} \mid \nexists \rho' \in \mathcal{R},\ \rho' \neq \rho,\ \rho \in \Gamma(\rho')\}$; *% Definition 7*
  % All representative rules $\rho : \mathbb{B} \twoheadrightarrow \mathbb{H}$ with highest confidence
  foreach *rule* $\rho : \mathbb{B} \twoheadrightarrow \mathbb{H}$ *in* $\mathcal{R}^* \mid \nexists \rho' \in \mathcal{R}^*, \gamma(\rho') > \gamma(\rho)$ **do**
    % Consistency of rule ρ at time i of $\mathcal{P}_0^n \times \mathcal{Q}_0^n$
    if $\mathcal{T} \cup \mathcal{Q}_0^n(i) \cup \{\mathbb{H}(\vec{a})\} \not\models \bot$
      $\big(\forall \vec{a} \mid \mathcal{T} \cup \mathcal{P}_0^n(i) \models \mathbb{B}[\vec{a}]|_{var(\mathbb{H})}\big)$
      % $\mathcal{Q}_0^n(i)$ and knowledge $\mathbb{H}[\vec{a}]$ from inductive reasoning
    then $\mathcal{Q}_0^n(i) \leftarrow \mathcal{Q}_0^n(i) \cup \{\mathbb{H}[\vec{a}]\}$; ;
  return $\mathcal{Q}_0^n(i)$;

Example 12. ***(Consistent Knowledge Discovery)***
Suppose that $\mathcal{Q}_5^7$, $\mathcal{R}_5^7$ are not exposed at time 5 because of defective sensors. By definition $\mathcal{R}_5^7(5)$ cannot be deduced from $\mathcal{T}$ neither $\mathcal{P}_5^7(5)$. Discovering it consists in applying Algorithm 3 with e.g., $\langle \mathcal{T}, \mathcal{P}_5^7, \mathcal{R}_5^7, 5, 2/3, 2, (2/3, 3/4)\rangle$. This leads to the representative rule (among others) in Example 11, which derives consistent knowledge in $\mathcal{R}_5^7(5)$: $(Road \sqcap \exists with.CongestedBus)(r_1)$. Applying the same approach to $\mathcal{R}_5^7(6)$ would reach to consistent but not accurate knowledge, hence the importance of minimum support, weight and confidence (cf. Experimental Results).

5 Experimental Results

We report (i) scalability, (ii) accuracy of Algorithm 3 (noted [A3]). The experiments have been conducted on a server of 4 Intel(R) Xeon(R) X5650, 2.67GHz cores, 6GB RAM.

• **Context:** Dynamic data is: [a] weather, [b] travel time, [c] incident, [d] event, [e] bus location in Dublin (Table 3). Data, synchronized through its temporal dimension, is transformed in $\mathcal{EL}^{++}$ using mapping techniques. We used a fixed (off-line) window of $n = 4,320$ snapshots (48 hours) for mining. We considered an ontology $\mathcal{T}$ of 55 concepts and 19 roles. The objective is to derive the status of buses (by mining $\mathcal{EL}^{++}$ rules across semantic data i.e., association of types of weather, travel condition, incident, events, bus delays) when not retrievable (due to 34% of missing data).

DataSet	Size (Mb / Day)	#Axioms / Update	#RDF Triples / Update
[a] Weather	3	53	318
[b] Travel Time	43	270	810
[c] Incident	0.1	81	324
[d] Event	9.5	480	1,150
[e] Bus	120	3,000	12,000

Table 3: Dynamic Data / Evolving Ontologies Details.

• **Scalability Result:** Figure 5 reports scalability of [A3] ($(\sigma_{\min}, \omega_{\min}, \gamma_{\min})$ being $(1/2, n, (2/3, 3/4))$) and compares its computation time with [Dehaspe and Raedt, 1997] using inductive logic noted [D97], [Srikant and Agrawal, 1995] using basic taxonomies noted [S95] and [Lécué and Pan, 2013] using ABox axioms-related rules noted [L13]. The evaluation is achieved on different (i) variations of $\mathcal{T}$ i.e., $\mathcal{T}[0]$, $\mathcal{T}[50]$, $\mathcal{T}[100]$ capturing a proportion of 0%, 50%, 100% of GCIs and RIs, (ii) number of evolving ontologies $|s|$ i.e., $\{1, 3, 5\}$ for respectively [e], [c,d,e], [a,b,c,d,e] in Table 3.

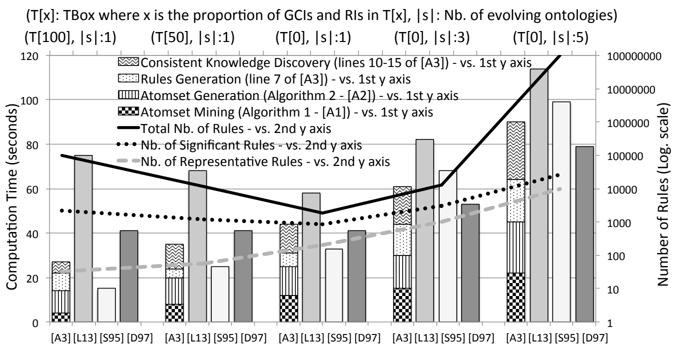

Figure 5: Scalability. 1^{st} x axis: Our approach vs. Related Works. on 5 Test Cases (cf. 2^{nd} x Axis for Configuration Settings). 1^{st} y axis: Computation Time in Seconds. 2^{nd} x axis: 5 Types of Semantic and Evolving Ontologies Configurations. 2^{nd} y axis: Search Space of Association Rules.

The scalability of all approaches decreases with the number of evolving ontologies, axioms. [D97] is the most scalable since it supports pruning strategies. Its performance remains unchanged for any variation of $\mathcal{T}$ since no semantics is supported. [S95], [A3] improve their scalability by clustering rules using semantics, which highly reduces the number of representative rules for [A3]. The off-line version of [L13] is the least scalable since knowledge discovery is based on the number significant rules, which is high. Our approach outperforms (i) [S95] when numerous evolving ontologies occur, (ii) [D97] when more semantics is captured by $\mathcal{T}$.

• **Accuracy Result:** Figure 6 reports accuracy with Table 4 as configuration. The weight is interpreted in [A3]. $\mathcal{T}[50]$ and all ontologies are considered. Accuracy is measured by validating induced knowledge over $10,000$ past situations in Dublin where buses status is known. All approaches can be compared since they expose similar high level results where only their semantic representation and expressivity differ.

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
$\sigma_{\min}$	.4	.4	.4	.4	.8	.8	.8	.8
$\gamma_{\min}$	(.4, .4)	(.4, .8)	(.8, .4)	(.8, .8)	(.4, .4)	(.4, .8)	(.8, .4)	(.8, .8)

Table 4: Support $\sigma_{\min}$, Confidence $\gamma_{\min}$ Configuration.

[A3] outperforms all approaches, even significantly when (i) weight associated to confidence $\gamma_{\min}$ is higher than $.4$, (ii) support $\sigma_{\min}$ is $.8$. The experiments v.s. [L13] show that genericity, representativeness, weight of rules largely contribute in reducing their quantity while improving quality.

• **Lessons Learnt:** The semantics of rules and its atomsets together with representativeness benefits classical rules mining approaches. Our approach, as a variant of [L13], [D97], [S95], benefits from (i)

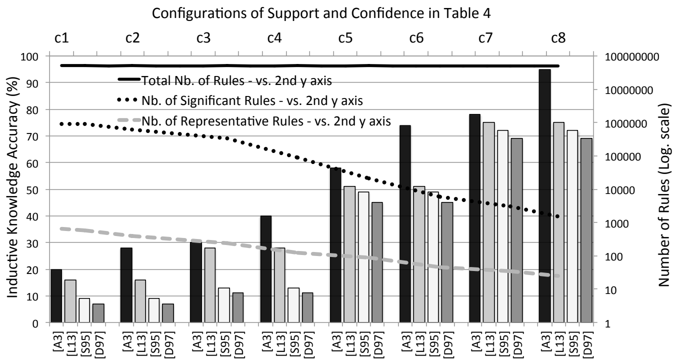

Figure 6: Accuracy. 1^{st} x axis: Our approach vs. Related Works on 8 Test Cases (cf. 2^{nd} x Axis for Configuration Settings). 1^{st} y axis: Accuracy of Discovered Knowledge. 2^{nd} x axis: 8 Types of Support / Confidence Configurations (cf. Table 4). 2^{nd} y axis: Search Space of Association Rules.

[L13] to discover association, (ii) [D97] to prune atomsets (for scalability), (iii) [S95] to capture their semantics (for accuracy). The scalability (accuracy) of knowledge discovery is negatively (positively) impacted by the number of data sources, snapshots, axioms, which impacts Algorithms 1, 2. Their number are critical as they drive heterogeneity in the rules elaboration, which could improve accuracy, but not scalability. It would be worst with more expressive DLs due to binding and containment checks, which confirms our initial choice of $\mathcal{EL}^{++}$.

6 Conclusion and Future Work

Our approach, combining inductive and deductive reasoning, discovers consistent knowledge by mining and applying association $\mathcal{EL}^{++}$ rules across DL-augmented dynamic data. Semantics was essential for (i) capturing consistent knowledge across evolving ontologies, (ii) raising its accuracy, (iii) improving scalability through identification of representative rules. Experiments have shown scalable, accurate, consistent knowledge discovery with data from Dublin.

In future work we will investigate scalable and incremental re-adjustment of rules in a context of streaming data.

Acknowledgment

This research has been funded by the European Commission within the 7th Framework Programme/Maria Curie Industry-Academia Partnerships and Pathways schema/PEOPLE Work Programme 2011 project K-Drive number 286348 (cf. `http://www.kdrive-project.eu`).

References

[Agrawal et al., 1993] Agrawal, R., Imielinski, T., and Swami, A. N. (1993). Mining association rules between sets of items in large databases. In *SIGMOD Conference*, pages 207–216.

[Agrawal et al., 1996] Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., and Verkamo, A. I. (1996). Fast discovery of association rules. In *Advances in Knowledge Discovery and Data Mining*, pages 307–328. AAAI/MIT Press.

[Anicic et al., 2011] Anicic, D., Fodor, P., Rudolph, S., and Stojanovic, N. (2011). Ep-sparql: a unified language for event processing and stream reasoning. In *Proceedings of the 20th international conference on World wide web*, pages 635–644. ACM.

[Baader et al., 2005] Baader, F., Brandt, S., and Lutz, C. (2005). Pushing the el envelope. In *IJCAI*, pages 364–369.

[Baader et al., 2008] Baader, F., Brandt, S., and Lutz, C. (2008). Pushing the el envelope further. In *OWLED*.

[Baader and Nutt, 2003] Baader, F. and Nutt, W. (2003). In *The Description Logic Handbook: Theory, Implementation, and Applications*.

[Bienvenu et al., 2012] Bienvenu, M., Lutz, C., and Wolter, F. (2012). Query containment in description logics reconsidered. In *KR*.

[Dehaspe and Raedt, 1997] Dehaspe, L. and Raedt, L. D. (1997). Mining association rules in multiple relations. In *ILP*, pages 125–132.

[Dietterich and Michalski, 1981] Dietterich, T. G. and Michalski, R. S. (1981). Inductive learning of structural descriptions: Evaluation criteria and comparative review of selected methods. *Artificial intelligence*, 16(3):257–294.

[Fanizzi et al., 2010] Fanizzi, N., d'Amato, C., and Esposito, F. (2010). Induction of concepts in web ontologies through terminological decision trees. In *ECML/PKDD (1)*, pages 442–457.

[Glimm et al., 2007] Glimm, B., Horrocks, I., Lutz, C., and Sattler, U. (2007). Conjunctive query answering for the description logic shiq. In *IJCAI*, pages 399–404.

[Huang and Stuckenschmidt, 2005] Huang, Z. and Stuckenschmidt, H. (2005). Reasoning with multiversion ontologies: A temporal logic approach. In *ISWC*, pages 398–412.

[Krötzsch et al., 2008] Krötzsch, M., Rudolph, S., and Hitzler, P. (2008). Description logic rules. In *ECAI*, pages 80–84.

[Labrinidis and Jagadish, 2012] Labrinidis, A. and Jagadish, H. (2012). Challenges and opportunities with big data. *Proceedings of the VLDB Endowment*, 5(12):2032–2033.

[Lécué, 2012] Lécué, F. (2012). Diagnosing changes in an ontology stream: A dl reasoning approach. In *AAAI*.

[Lécué and Pan, 2013] Lécué, F. and Pan, J. Z. (2013). Predicting knowledge in an ontology stream. In *IJCAI*.

[Ramaswamy et al., 2000] Ramaswamy, S., Rastogi, R., and Shim, K. (2000). Efficient algorithms for mining outliers from large data sets. In *ACM SIGMOD Record*, volume 29, pages 427–438. ACM.

[Reiter, 1980] Reiter, R. (1980). A logic for default reasoning. *Artificial intelligence*, 13(1):81–132.

[Sadilek et al., 2012] Sadilek, A., Kautz, H. A., and Silenzio, V. (2012). Predicting disease transmission from geo-tagged micro-blog data. In *AAAI*.

[Song et al., 2014] Song, X., Zhang, Q., Sekimoto, Y., and Shibasaki, R. (2014). Intelligent system for urban emergency management during large-scale disaster. In *AAAI*, pages 458–464.

[Srikant and Agrawal, 1995] Srikant, R. and Agrawal, R. (1995). Mining generalized association rules. In *VLDB*, pages 407–419.

[Valle et al., 2009] Valle, E. D., Ceri, S., van Harmelen, F., and Fensel, D. (2009). It's a streaming world! reasoning upon rapidly changing information. *IEEE Intelligent Systems*, 24(6):83–89.

[Zaki, 2000] Zaki, M. J. (2000). Generating non-redundant association rules. In *KDD*, pages 34–43.