Mining Hypotheses from Data in OWL

DOI:
10.1007/978-3-319-68288-4_34

Mining Hypotheses from Data in OWL: Advanced Evaluation and Complete Construction

Viachaslau Sazonau and Uli Sattler

The University of Manchester
Oxford Road, Manchester, M13 9PL, UK
{sazonauv, sattler}@cs.manchester.ac.uk

Abstract. Automated acquisition (learning) of ontologies from data has attracted research interest because it can complement manual, expensive construction of ontologies. We investigate the problem of General Terminology Induction in OWL, i.e. acquiring general, expressive TBox axioms (hypotheses) from an ABox (data). We define novel measures designed to rigorously evaluate the quality of hypotheses while respecting the standard semantics of OWL. We propose an informed, data-driven algorithm that constructs class expressions for hypotheses in OWL and guarantees completeness. We empirically evaluate the quality measures on two corpora of ontologies and run a case study with a domain expert to gain insight into applicability of the measures and acquired hypotheses. The results show that the measures capture different quality aspects and not only correct hypotheses can be interesting.

1 Introduction

In computer science, an *ontology* is a machine-processable representation of knowledge about some domain. Ontologies are encoded in ontology languages, such as the expressive Web Ontology Language [11] (OWL) based on Description Logics [3] (DLs). An ontology is a set of logical statements, called *axioms*. Axioms can be universal statements or specific facts. The set of universal statements of an ontology is called the *TBox* and represents schema-level conceptual relationships, or *terminology*. The set of facts of an ontology is called the *ABox* and represents instance-level class and property assertions, or *data*. Besides simple "SubClassOf" relationships and class definitions, OWL allows for encoding complex TBox axioms such as general class inclusions (GCIs) where complex class expressions occur on both sides, e.g. $\exists hasChild.\top \sqsubseteq Mother \sqcup Father$ states that "having a child implies being a mother or father".

Since manual engineering of TBoxes is a difficult, time-consuming task, automated acquisition of them from data has attracted research attention. In this paper, we investigate learning expressive TBox axioms (hypotheses) from a given ABox (data). Our contributions are as follows:

- definitions of novel quality measures that can rigorously evaluate expressive GCIs in OWL respecting its semantics;
- an informed, bottom-up algorithm that efficiently constructs complex class expressions (and thus GCIs) in OWL and guarantees completeness;

- an empirical analysis of the relationships between the quality measures via mutual correlations;
- the design and execution of a case study which confirms the ability of our approach to generate three different kinds of interesting hypotheses and gains insight into relationships of the measures with hypothesis validity and interestingness.

2 Preliminaries

We assume the reader to be familiar with DLs [3] and OWL [11]. We denote an ontology as $\mathcal{O} := \mathcal{T} \cup \mathcal{A}$, where $\mathcal{T}$ and $\mathcal{A}$ are its TBox and ABox, respectively. An *axiom* is denoted as α or η. A general class inclusion (GCI) is an axiom of the form $C \sqsubseteq D$, where C and D are (possibly complex) class expressions, and corresponds to a "SubClassOf" axiom in OWL. An object property inclusion (OPI) is an axiom of the form $R \sqsubseteq S$, where R and S are (possibly complex) object property expressions, and corresponds to a "SubObjectPropertyOf" axiom in OWL. A *hypothesis* is a TBox axiom (GCI or OPI). An ABox axiom, called *fact*, is an assertion of the form $C(a)$ or $R(a, b)$, where C is a class expression, R an object property, a, b individuals. The set of all terms occurring in an ontology $\mathcal{O}$ is called the *signature* of $\mathcal{O}$ and denoted as $\widetilde{\mathcal{O}}$ ($\widetilde{\mathcal{T}}$ is the signature of $\mathcal{T}$). We denote the set of all individuals occurring in $\mathcal{O}$ as $in(\mathcal{O})$. We use $\models$ to denote the usual entailment relation and $\equiv$ to denote logical equivalence. The function $\ell(C)$ returns the usual syntactic length [13, 3] of a class expression C, e.g. $\ell(\exists R.A \sqcap \forall R.(\neg B \sqcup \exists S.B)) = 9$; $\ell(C \sqsubseteq D) = \ell(C) + \ell(D)$; $\ell(\mathcal{O}) = \sum_{\alpha \in \mathcal{O}} \ell(\alpha)$.

3 Related Work

There are different approaches to acquiring TBox axioms from data. The common approach is Class Description Learning [5, 14, 7, 16, 15, 18] (CDL) which aims at inducing a description (class expression) C of a given class name A using a set of positive and negative training examples. Statistical Schema Induction [22] uses Association Rules Mining (ARM) to generate and evaluate candidate axioms using off-the-shelf quality measures [10]. BelNet [23] learns a Bayesian Network from data and uses its structure to generate the corresponding TBox. In contrast to CDL, the last two approaches are not restricted to learning only class descriptions and can generate GCIs with complex class expressions on both sides. However, they require specifying shapes of generated axioms and have so far been considerably limited in *expressivity*, i.e. richness of knowledge that generated axioms are able to capture. Moreover, they tend to view a given ABox (data) under the Closed Word Assumption (CWA) or some form of it [9]. This is unnatural for the standard semantics of OWL allowing for the Open World Assumption (OWA), i.e. *incomplete information*. In addition, the approaches usually ignore the given TBox while generating candidate axioms.

Like ARM-based approaches, we focus on learning GCIs rather than class expressions. The rationale is that the former can express arbitrary implications, e.g. "people who pay dog tax also buy dog food", while the latter cannot since it captures commonalities in the given group of individuals (as positive or negative examples), e.g. "people who pay dog tax". Thus, the goals of learning GCIs and learning class expressions are rather different. To draw further similarities between our approach and ARM, we can view an individual as a transaction that contains class expressions as its items. A class expression is included in the transaction if and only if the individual is an instance of that class expression. However, in contrast to items in ARM, class expressions can be logically related to each other (in light of the TBox) and it can be unknown whether a class expression is in the transaction or not because of the OWA. In addition, unlike items in ARM, class expressions are not usually known in advance and naive generation of them is infeasible in all but trivial cases.

4 Advanced Evaluation of Hypotheses

A candidate axiom, or *hypothesis*, can be evaluated by different quality criteria. One can use the usual axiom *length* and *depth* [4, 13, 3] to evaluate readability. As we suggested in [20], logical quality can be evaluated by *consistency, informativeness, and logical strength (weakness)*: an axiom α is called *consistent* with an ontology $\mathcal{O}$ if $\mathcal{O} \cup \{\alpha\}$ is consistent; α is called *informative* for a TBox $\mathcal{T}$ if $\mathcal{T} \not\models \alpha$; α is said to be *weaker* than another axiom α' if $\alpha' \models \alpha$ and $\alpha \not\models \alpha'$. Statistical quality can be evaluated by *fitness* and *braveness* [20]. Intuitively, fitness counts the number of facts entailed by a hypothesis and braveness counts the number of "guesses" of a hypothesis.

Definition 1 (fitness, braveness). *Let $\mathcal{O} := \mathcal{T} \cup \mathcal{A}$ be an ontology, $\mathbb{C}$ a set of class expressions with their negations included, α a GCI consistent with $\mathcal{O}$. Then, the* fitness *and* braveness *of α are defined as follows:*

$$\begin{aligned} fit(\alpha, \mathcal{O}, \mathbb{C}) &:= dlen(\pi(\mathcal{O}, \mathbb{C}),\ \mathcal{T}) - dlen(\pi(\mathcal{O}, \mathbb{C}),\ \mathcal{T} \cup \{\alpha\}) \\ bra(\alpha, \mathcal{O}, \mathbb{C}) &:= dlen(\psi(\alpha, \mathcal{O}, \mathbb{C}),\ \mathcal{O}) \end{aligned}$$

where $\pi(\mathcal{O}, \mathbb{C}) := \{C(a) \mid \mathcal{O} \models C(a),\ C \in \mathbb{C},\ a \in in(\mathcal{O})\}$,[1] *$\psi(\alpha, \mathcal{O}, \mathbb{C}) := \pi(\mathcal{O} \cup \{\alpha\},\ \mathbb{C}) \setminus \pi(\mathcal{O}, \mathbb{C})$, $dlen(\mathcal{B}, \mathcal{O}) := min\{\ell(\mathcal{B}') \mid \mathcal{B}' \cup \mathcal{O} \equiv \mathcal{B} \cup \mathcal{O}\}$.*

4.1 New Logical Measures

To capture further aspects of logical quality, we propose new logical measures: dissimilarity and complexity. These are *numeric* logical measures measures (compare to consistency, informativeness, and logical strength mentioned above).

Dissimilarity Given a GCI $C \sqsubseteq D$, one can measure how "dissimilar" C and D are with respect to the TBox. Intuitively, the more dissimilar they are, the more "surprising" the axiom is for the TBox. We adapt the class similarity measure from [2].

Definition 2 (Dissimilarity). *Let $\mathcal{O} := \mathcal{T} \cup \mathcal{A}$ be an ontology, $\mathbb{C}$ a set of class expressions, $subs(C, \mathbb{C}, \mathcal{T}) := \{C' \in \mathbb{C} \cup \{C\} \mid \mathcal{T} \models C \sqsubseteq C'\}$. The* dissimilarity *of $\alpha := C \sqsubseteq D$ is defined as follows:*

$$dsim(\alpha, \mathbb{C}, \mathcal{T}) := 1 - \frac{|subs(C, \mathbb{C}, \mathcal{T}) \cap subs(D, \mathbb{C}, \mathcal{T})|}{|subs(C, \mathbb{C}, \mathcal{T}) \cup subs(D, \mathbb{C}, \mathcal{T})|}.$$

[1] It is the result of retrieving instances of every $C \in \mathbb{C}$.

Informally, given a TBox $\mathcal{T}$, the dissimilarity of a GCI $C \sqsubseteq D$ measures how many common subsumers the class expressions C and D have in a set $\mathbb{C}$ of class expressions.

Example 1. Consider the following TBox:

$$\begin{aligned}\mathcal{T} := \{&C_1 \sqsubseteq B_1,\ B_1 \sqsubseteq A_1,\ A_1 \sqsubseteq A,\\ &C_2 \sqsubseteq B_2,\ B_2 \sqsubseteq A_2,\ A_2 \sqsubseteq A\}.\end{aligned}$$

Given $\mathbb{C} := \widetilde{\mathcal{T}}$ (all classes of $\mathcal{T}$), the dissimilarity of $\alpha_1 := C_1 \sqsubseteq C_2$ is higher than the one of $\alpha_2 := A_1 \sqsubseteq C_2$:

$$\begin{aligned}dsim(\alpha_1, \mathbb{C}, \mathcal{T}) &= 1 - \frac{|\{A\}|}{|\{A, A_1, B_1, C_1, A_2, B_2, C_2\}|} = \frac{6}{7}\\ dsim(\alpha_2, \mathbb{C}, \mathcal{T}) &= 1 - \frac{|\{A\}|}{|\{A, A_1, A_2, B_2, C_2\}|} = \frac{4}{5}\end{aligned}$$

The dissimilarity of an OPI is defined analogously and omitted for the sake of brevity. The minimal (maximal) value of dissimilarity implies that all subsumers are the same (different). Dissimilarity is a *symmetric* measure, i.e.

$$dsim(C \sqsubseteq D,\ \mathbb{C}, \mathcal{T}) = dsim(D \sqsubseteq C,\ \mathbb{C}, \mathcal{T}).$$

Complexity Given an axiom α, we can compare the *complexity* of the new theory $\mathcal{T} \cup \{\alpha\}$ with the complexity of the old theory $\mathcal{T}$ by quantifying how many *new entailments* the new theory has. As the set of new entailments is infinite in general, we only consider a finite subset of them.

Definition 3 (Complexity). *Let $\mathcal{O} := \mathcal{T} \cup \mathcal{A}$ be an ontology, $\mathbb{C}$ a set of class expressions. The* complexity *of $\alpha := C \sqsubseteq D$ is defined as follows: $com(\alpha, \mathbb{C}, \mathcal{T}) := |\{\eta \mid \mathcal{T} \cup \{\alpha\} \models \eta,\ \mathcal{T} \not\models \eta,\ \eta = C_1 \sqsubseteq C_2,\ C_1, C_2 \in \mathbb{C}\}|$.*

Thus, we only count new entailments that are subsumptions between class expressions from a fixed set $\mathbb{C}$. The complexity of an OPI is defined analogously and omitted for the sake of brevity. In contrast to dissimilarity, complexity is *asymmetric*. They are rather independent measures, see Example 2.

Example 2. Let us calculate the complexity of the axioms α_1 and α_2 from Example 1:

$$\begin{aligned}com(\alpha_1, \mathbb{C}, \mathcal{T}) &= |\{C_1 \sqsubseteq C_2, C_1 \sqsubseteq B_2, C_1 \sqsubseteq A_2\}| = 3,\\ com(\alpha_2, \mathbb{C}, \mathcal{T}) &= |\{C_1 \sqsubseteq C_2, C_1 \sqsubseteq B_2, C_1 \sqsubseteq A_2,\\ &\qquad B_1 \sqsubseteq C_2, B_1 \sqsubseteq B_2, B_1 \sqsubseteq A_2,\\ &\qquad A_1 \sqsubseteq C_2, A_1 \sqsubseteq B_2, A_1 \sqsubseteq A_2\}| = 9.\end{aligned}$$

Thus, α_1 has lower complexity than α_2 but higher dissimilarity. In addition, consider the axiom $\alpha_3 := B_1 \sqcap C_2 \sqsubseteq A_1$: $com(\alpha_3, \mathbb{C}, \mathcal{T}) = 0$ since $\mathcal{T} \models \alpha_3$ but

$$dsim(\alpha_3, \mathbb{C}, \mathcal{T}) = 1 - \frac{|\{A, A_1\}|}{|\{A, B_1, A_1, C_2, B_2, A_2\}|} = \frac{2}{3}.$$

4.2 New Statistical Measures

We propose new statistical measures that capture further aspects of statistical quality while respecting the standard semantics of OWL and given TBox. They are based on *counting instances* of certain kinds.

Definition 4 (Instance function). *Let $\mathcal{O}$ be an ontology; $\mathring{C} \in \{C, ?C\}$, where C is a class expression. The* instance function *is defined as follows:*

$$inst(\mathring{C}, \mathcal{O}) := \begin{cases} \{a \in in(\mathcal{O}) \mid \mathcal{O} \models C(a)\} & \textit{if } \mathring{C} = C \\ \{a \in in(\mathcal{O}) \mid \mathcal{O} \not\models C(a) \ \wedge \ \mathcal{O} \not\models \neg C(a)\} & \textit{if } \mathring{C} = ?C \end{cases}$$

Basic Measures Let us consider a GCI $C \sqsubseteq D$. The axiom states that all instances of C are also instances of D. Given an ontology $\mathcal{O} := \mathcal{T} \cup \mathcal{A}$, we can check *how well* the data in $\mathcal{A}$ supports this statement taking the background knowledge in $\mathcal{T}$ into account.

Definition 5 (Basic measures). *Given an ontology $\mathcal{O}$, the* basic coverage, support, contradiction, assumption *of $\alpha := C \sqsubseteq D$ are defined, respectively, as follows:*

$$\begin{array}{ll} bcov(\alpha, \mathcal{O}) := |inst(C,\ \mathcal{O})| & bsup(\alpha, \mathcal{O}) := |inst(C \sqcap D,\ \mathcal{O})| \\ bcnt(\alpha, \mathcal{O}) := |inst(C \sqcap \neg D,\ \mathcal{O})| & basm(\alpha, \mathcal{O}) := |inst(C,\ \mathcal{O}) \cap inst(?D,\ \mathcal{O})| \end{array}$$

Support is presumably a *positive* measure, i.e. higher values indicate better quality, while contradiction and assumption are presumably *negative* ones, i.e. lower values indicate better quality. Coverage is neither positive nor negative as it is the sum of support, contradiction, and assumption. Support is a symmetric measure, while others are not. The basic measures respect the OWA via distinguishing assumption and contradiction.

Example 3. Consider the ontology $\mathcal{O} := \mathcal{T} \cup \mathcal{A}$ that models family relations, where the TBox $\mathcal{T}$ and ABox $\mathcal{A}$ are as follows (hc, mt stand for $hasChild$, $marriedTo$).

$$\begin{aligned} \mathcal{T} = \{&Father \sqsubseteq Man,\ Mother \sqsubseteq Woman,\ Man \sqsubseteq \neg Woman,\ mt \sqsubseteq mt^{-}\}, \\ \mathcal{A} = \{&Man(Arthur),\ Father(Chris),\ Father(James),\ Woman(Charlotte), \\ &Woman(Margaret),\ Mother(Penelope),\ Mother(Victoria), \\ &hc(James, Charlotte),\ hc(Victoria, Charlotte),\ hc(Chris, Victoria), \\ &hc(Penelope, Victoria),\ hc(Chris, Arthur),\ hc(Penelope, Arthur), \\ &mt(Chris, Penelope),\ mt(James, Victoria),\ mt(Arthur, Margaret)\}. \end{aligned}$$

Consider the following axioms:

$$\alpha_1 := \exists mt.\top \sqsubseteq Mother, \qquad \alpha_2 := \exists hc.\top \sqsubseteq Mother.$$

Their basic measures are calculated as follows:

$$\begin{array}{llll} bsup(\alpha_1, \mathcal{O}) = 2 & bcnt(\alpha_1, \mathcal{O}) = 3 & basm(\alpha_1, \mathcal{O}) = 1 & bcov(\alpha_1, \mathcal{O}) = 6 \\ bsup(\alpha_2, \mathcal{O}) = 2 & bcnt(\alpha_2, \mathcal{O}) = 2 & basm(\alpha_2, \mathcal{O}) = 0 & bcov(\alpha_2, \mathcal{O}) = 4 \end{array}$$

Thus, α_2 is better than α_1 because its support is the same but its contradiction and assumption are lower.

The basic measures can be defined for an OPI $R \sqsubseteq S$ in the same way as for a GCI $C \sqsubseteq D$. The only difference is that, instead of returning instances of a class expression C, the instance function would return instances of an object property expression R, i.e. individual pairs (a, b) which are entailed to be connected by R. Please note that assumption resembles braveness [20] but counts "guesses" of a hypothesis in a more straightforward way since it depends only on a hypothesis and ontology.

Main Measures The basic measures only consider the "forward" direction of a GCI $C \sqsubseteq D$. According to the semantics of OWL, $C \sqsubseteq D$ has also the "backward" direction. Formally, $C \sqsubseteq D \equiv \neg D \sqsubseteq \neg C$ which is called the law of *contraposition*, where $\neg D \sqsubseteq \neg C$ is called the *contrapositive* of $C \sqsubseteq D$. Thus, $C \sqsubseteq D$ not only implies that all instances of C are instances of D but also implies that all instances of $\neg D$ are instances of $\neg C$. We refine the basic measures using a syntactic trick to "merge" a GCI and its contrapositive into a single GCI.

Definition 6 (Main Measures). *Let $\mathcal{O}$ be an ontology, $\alpha := C \sqsubseteq D$, and $\overline{\alpha} := C \sqcup \neg D \sqsubseteq \neg C \sqcup D$. The* main coverage, support, contradiction, assumption *of α are defined, respectively, as follows:*

$$cov(\alpha, \mathcal{O}) := bcov(\overline{\alpha}, \mathcal{O}) \qquad sup(\alpha, \mathcal{O}) := bsup(\overline{\alpha}, \mathcal{O})$$
$$cnt(\alpha, \mathcal{O}) := bcnt(\overline{\alpha}, \mathcal{O}) \qquad asm(\alpha, \mathcal{O}) := basm(\overline{\alpha}, \mathcal{O})$$

In comparison to the basic measures, see Definition 5, their respective main measures additionally count individuals relevant for the contrapositive. Example 4 shows how a main measure can differ from its basic measure.

Example 4. In Example 3, we evaluate $\alpha_2 := \exists hc.\top \sqsubseteq Mother$ via the basic measures. Its basic assumption is $basm(\alpha_2, \mathcal{O}) = 0$, i.e. α_2 makes no "guesses". However, its main assumption is $asm(\alpha_2, \mathcal{O}) = 1$. Indeed, as $Arthur$ is an instance of $\neg Mother$, the axiom α_2 assumes that $Arthur$ has no children, i.e. he is an instance of $\neg(\exists hc.\top)$.

In contrast to the basic measures, the main measures always return the same values for an axiom and its contrapositive. Thus, they respect the semantics of OWL better than the basic measures. The main measures of an axiom can be represented via the basic measures of that axiom and its contrapositive. These properties are stated by Lemma 1.

Lemma 1. *Let $\mathcal{O}$ be an ontology, $\alpha := C \sqsubseteq D$, and $\alpha' := \neg D \sqsubseteq \neg C$. Then*

$$\begin{aligned}
cov(\alpha, \mathcal{O}) &= cov(\alpha', \mathcal{O}) = bcov(\alpha, \mathcal{O}) + bcov(\alpha', \mathcal{O}) - bcnt(\alpha, \mathcal{O}) \\
sup(\alpha, \mathcal{O}) &= sup(\alpha', \mathcal{O}) = bsup(\alpha, \mathcal{O}) + bsup(\alpha', \mathcal{O}) \\
cnt(\alpha, \mathcal{O}) &= cnt(\alpha', \mathcal{O}) = bcnt(\alpha, \mathcal{O}) = bcnt(\alpha', \mathcal{O}) \\
asm(\alpha, \mathcal{O}) &= asm(\alpha', \mathcal{O}) = basm(\alpha, \mathcal{O}) + basm(\alpha', \mathcal{O})
\end{aligned}$$

Proof. Follows from Definition 4, 5, and 6, see [19] for details.

Clearly, the basic and main measures coincide if $\neg C$ and $\neg D$ have no instances in $\mathcal{O}$, e.g. C and D are $\mathcal{EL}$ class expressions and $\mathcal{O}$ is in $\mathcal{EL}$. Example 5 illustrates how evaluating a disjointness axiom under the OWA differs from evaluating it under the CWA which is commonly made for learning disjointness axioms, see e.g. [8].

Example 5. Consider the ontology

$$\mathcal{O} := \{A(a_1), \dots, A(a_m),\ B(b_1), \dots, B(b_n)\}.$$

Under the CWA, the absence of information in $\mathcal{O}$ is treated as negation:

$$\mathcal{O}^{\neg} := \mathcal{O} \cup \{\neg B(a_1), \dots, \neg B(a_m),\ \neg A(b_1), \dots, \neg A(b_n)\}.$$

Consider the disjointness axiom $\alpha := A \sqsubseteq \neg B$. Under the CWA, it is assumed, perhaps wrongly, to be of high quality: $sup(\alpha, \mathcal{O}^{\neg}) = m + n$, $asm(\alpha, \mathcal{O}^{\neg}) = 0$. In contrast, under the OWA, its evaluation better reflects the state of knowledge in $\mathcal{O}$: $sup(\alpha, \mathcal{O}) = 0$, $asm(\alpha, \mathcal{O}) = m + n$.

Composite Measures As an axiom $C \sqsubseteq D$ in OWL is similar to an association rule $X \Rightarrow Y$ in ARM, rule measures [10] can be adapted to OWL. The challenge is to respect the OWA, i.e. consider that there is $?C$, see Definition 4, in addition to C and $\neg C$. Given a rule measure $f(X, Y)$, we suggest to translate it as follows. First, substitute each positive occurrence of a variable X (Y) in $f(X, Y)$ with a class expression C (D). If neither X nor Y occurs negatively in $f(X, Y)$, then the translation is finished and results in the axiom measure $f(C, D)$. Otherwise, obtain two axiom measures as follows: substitute each negative occurrence $\neg X$ ($\neg Y$) in $f(X, Y)$ with $\neg C$ ($\neg D$), resulting in $f^{\neg}(C, D)$, and with $?C$ ($?D$), resulting in $f^{?}(C, D)$. Following this procedure, we translate the standard rule measures: confidence, lift, and conviction.

Definition 7 (Composite basic measures). *Let $\mathcal{O}$ be an ontology; $\mathring{C} \in \{C, ?C\}$, where C is a class expression;*

$$\mathbf{P}_{\mathcal{O}}(\mathring{C}_1, \dots, \mathring{C}_k) := \frac{1}{|in(\mathcal{O})|} \Big| \bigcap_{i=1}^{k} inst(\mathring{C}_i, \mathcal{O}) \Big|.$$

The basic confidence, lift, negated and assumed conviction *of* $\alpha := C \sqsubseteq D$ *are defined, respectively, as follows:*

$$bconf(\alpha, \mathcal{O}) := \frac{\mathbf{P}_{\mathcal{O}}(C, D)}{\mathbf{P}_{\mathcal{O}}(C)} \qquad blift(\alpha, \mathcal{O}) := \frac{\mathbf{P}_{\mathcal{O}}(C, D)}{\mathbf{P}_{\mathcal{O}}(C) \cdot \mathbf{P}_{\mathcal{O}}(D)}$$

$$bconv^{\neg}(\alpha, \mathcal{O}) := \frac{\mathbf{P}_{\mathcal{O}}(C) \cdot \mathbf{P}_{\mathcal{O}}(\neg D)}{\mathbf{P}_{\mathcal{O}}(C, \neg D)} \qquad bconv^{?}(\alpha, \mathcal{O}) := \frac{\mathbf{P}_{\mathcal{O}}(C) \cdot \mathbf{P}_{\mathcal{O}}(?D)}{\mathbf{P}_{\mathcal{O}}(C, ?D)}$$

The OWA is taken into consideration via distinguishing negated and assumed conviction. The composite basic measures can be rewritten using the basic coverage, support, contradiction, and assumption, see [19] for details.

Example 6. We calculate the composite basic measures of the axioms α_1 and α_2 in Example 3. We first calculate the required probabilities (M stands for $Mother$): $\mathbf{P}_{\mathcal{O}}(M) = \frac{2}{7}$, $\mathbf{P}_{\mathcal{O}}(\neg M) = \frac{3}{7}$, $\mathbf{P}_{\mathcal{O}}(?M) = \frac{2}{7}$. Then, we use them along with the basic measures calculated in Example 3:
$bconf(\alpha_1, \mathcal{O}) = \frac{2}{6} = \frac{1}{3}$, $blift(\alpha_1, \mathcal{O}) = \frac{2}{6 \cdot \frac{2}{7}} = \frac{7}{6}$, $bconv^{\neg}(\alpha_1, \mathcal{O}) = \frac{6 \cdot \frac{3}{7}}{3} = \frac{6}{7}$, $bconv^{?}(\alpha_1, \mathcal{O}) = \frac{6 \cdot \frac{2}{7}}{1} = \frac{12}{7}$; $bconf(\alpha_2, \mathcal{O}) = \frac{2}{4} = \frac{1}{2}$, $blift(\alpha_2, \mathcal{O}) = \frac{2}{4 \cdot \frac{2}{7}} = \frac{7}{4}$, $bconv^{\neg}(\alpha_2, \mathcal{O}) = \frac{4 \cdot \frac{3}{7}}{2} = \frac{6}{7}$, $bconv^{?}(\alpha_2, \mathcal{O}) = \frac{4 \cdot \frac{2}{7}}{0} = \infty$.

The composite basic measures can be refined to treat GCIs according to the standard semantics of OWL, i.e. as being equivalent to their contrapositives.

Definition 8 (Composite main measures). *Let $\mathcal{O}$ be an ontology, $\alpha := C \sqsubseteq D$, and $\overline{\alpha} := C \sqcup \neg D \sqsubseteq \neg C \sqcup D$. The* main confidence, lift, negated and assumed conviction *of α are defined, respectively, as follows:*

$$conf(\alpha, \mathcal{O}) := bconf(\overline{\alpha}, \mathcal{O}) \qquad lift(\alpha, \mathcal{O}) := blift(\overline{\alpha}, \mathcal{O})$$
$$conv^{\neg}(\alpha, \mathcal{O}) := bconv^{\neg}(\overline{\alpha}, \mathcal{O}) \qquad conv^{?}(\alpha, \mathcal{O}) := bconv^{?}(\overline{\alpha}, \mathcal{O})$$

A lemma analogous to Lemma 1 holds for the composite main measures, i.e. they treat a GCI as being equivalent to its contrapositive and can be rewritten using the main measures and hence the basic measures [19].

5 Complete Construction of Hypotheses

We reduce the problem of constructing hypotheses to the problem of constructing class (and property) expressions. Indeed, given a set $\mathbb{C}$ of class expressions of interest, we can generate all possible GCIs using class expressions from $\mathbb{C}$ as a left-hand side or right-hand side, i.e. $\{C \sqsubseteq D \mid C, D \in \mathbb{C}\}$. Thus, the number of generated GCIs is quadratic in the size of $\mathbb{C}$. As we suggested in [20], class expressions $\mathbb{C}$ can be generated from some "seed" signature Σ using certain construction rules (templates), e.g. all pairwise conjunctions, simple existential restrictions, etc. However, it is generally hard to know which templates are likely to produce useful class expressions. Moreover, a brute-force procedure that generates *all* class expressions is doomed even for inexpressive DLs, e.g. $\mathcal{EL}$. For example, given n class and m object property names, a number of all $\mathcal{EL}$ class expressions of length up to 5 grows as fast as $O(n^3 + n^2 \cdot m^2 + n \cdot m^4)$.

We propose an informed, bottom-up algorithm that constructs *all* class expressions $\mathbb{C}$ of length up to ℓ_{max} in a given $\mathcal{DL}$ that have at least s_{min} instances, i.e. sufficient evidence in data. Importantly, the algorithm avoids considering all other class expressions that are numerous, e.g. all class expressions without instances (and many others). We integrate two ideas in one algorithm: enumerating class expressions via a *refinement operator* [14, 7, 16] and pruning unpromising (insufficiently supported by data) class expressions from the search *a priori*. A *downward* refinement operator[2] ρ for $\mathcal{DL}$ specifies a set $\rho(C)$ of specialisations of a class expression C in that $\mathcal{DL}$. Refinement operators normally use the classic subsumption $\sqsubseteq$ as an ordering on class expressions. Thus, $C' \in \rho(C)$ implies $C' \sqsubseteq C$.[3]

Example 7. Given the terms M, W, hc (standing for Man, $Woman$, $hasChild$) from Example 3, the refinement operator ρ can be used to traverse the space of $\mathcal{EL}$ class expressions as follows:

$$\rho(\top) = \{M,\ W,\ \exists hc.\top\}$$
$$\rho(M) = \{M \sqcap M,\ M \sqcap W,\ M \sqcap \exists hc.\top\}$$

[2] It is sufficient to consider only downward refinement operators.

[3] The statement $C' \sqsubseteq C$ is the abbreviation of $\emptyset \models C' \sqsubseteq C$.

$$\rho(W) = \{W \sqcap M,\ W \sqcap W,\ W \sqcap \exists hc.\top\}$$
$$\rho(\exists hc.\top) = \{\exists hc.M, \exists hc.W, \exists hc.\exists hc.\top, \exists hc.\top \sqcap M, \exists hc.\top \sqcap W, \exists hc.\top \sqcap \exists hc.\top\}$$
$$\ldots$$

The mechanics of refinement operators allows for pruning unpromising class expressions from the search without even generating them (and hence without checking their instances). Indeed, a specialisation of a class expression cannot have more instances than the class expression itself has, see Lemma 2.

Lemma 2 (Anti-monotone property of specialisations). *Let $\mathcal{O}$ be an ontology, C a class expression, ρ a (downward) refinement operator. Then, $C' \in \rho(C)$ implies $|inst(C', \mathcal{O})| \leq |inst(C, \mathcal{O})|$.*

Lemma 2 implies that if C has an insufficient number of instances, then so do all its further specialisations. It is essentially the anti-monotone property of itemsets used in the APRIORI algorithm [1] which we have defined for OWL class expressions. Due to this similarity, we call our algorithm of constructing class expressions DL-APRIORI, see Algorithm 1.

Algorithm 1 DL-APRIORI $(\mathcal{O}, \Sigma, \mathcal{DL}, \ell_{max}, s_{min})$

```
 1: inputs
 2:    $\mathcal{O} := \mathcal{T} \cup \mathcal{A}$: an ontology
 3:    $\Sigma$: a finite set of terms such that $\top \in \Sigma$
 4:    $\mathcal{DL}$: a DL for class expressions
 5:    $\ell_{max}$: a maximal length of a class expression such that $1 \leq \ell_{max} < \infty$
 6:    $s_{min}$: a minimal instance threshold (support) such that $0 < s_{min} \leq |in(\mathcal{O})|$
 7: outputs
 8:    $\mathbb{C}$: the set of all class expressions satisfying the input constraints
 9: do
10:    $\mathbb{C} \leftarrow \emptyset$    % initialise the final set of class expressions
11:    $\mathbb{D} \leftarrow \{\top\}$    % initialise the set of class expressions yet to be specialised
12:    $\rho \leftarrow getOperator(\mathcal{DL}, \Sigma, \ell_{max}, \mathcal{T})$    % initialise a refinement operator $\rho$
13:    while $\mathbb{D} \neq \emptyset$ do
14:       $C \leftarrow pick(\mathbb{D})$    % pick a class expression $C$ to be specialised
15:       $\mathbb{D} \leftarrow \mathbb{D} \backslash \{C\}$    % remove $C$ from $\mathbb{D}$
16:       $\mathbb{C} \leftarrow \mathbb{C} \cup \{C\}$    % add $C$ to $\mathbb{C}$
17:       $\mathbb{C}' \leftarrow specialise(C, \rho)$    % specialise $C$ using $\rho$
18:       $\mathbb{D}_C \leftarrow \{D \in urc(\mathbb{C}') \mid \nexists D' \in \mathbb{C} \cup \mathbb{D} : D' \equiv D\}$    % discard syntactic variations
19:       $\mathbb{D} \leftarrow \mathbb{D} \cup \{D \in \mathbb{D}_C \mid |inst(D, \mathcal{O})| \geq s_{min}\}$    % add suitable specialisations
20:    end while
21:    return $\mathbb{C}$
```

DL-APRIORI operates as follows. First, we initialise the refinement operator ρ (see Line 12) with the given logic $\mathcal{DL}$, signature Σ, maximal length ℓ_{max}, and TBox $\mathcal{T}$ such that it only constructs specialisations satisfying the constraints and takes $\mathcal{T}$ into consideration, e.g. its class hierarchy. The construction starts from $\top$, see also Example 7. The operator repeatedly specialises every expression picked from the set $\mathbb{D}$ of candidates and adds its suitable specialisations to $\mathbb{D}$ (see Line 14 – 19). A specialisation is suitable if it is not a syntactic variation of an already constructed one (see Line 18 where the function $urc(\mathbb{C}')$ returns unique representatives of logically equivalent class expressions in a set $\mathbb{C}'$) and satisfies the minimal support s_{min} (see Line 19). Once the set $\mathbb{D}$ is empty, the algorithm terminates. Intuitively, s_{min} acts as a "noise threshold" that prunes expressions with insufficient evidence and therefore should be sufficiently small to avoid missing useful expressions.

Given $\mathcal{DL} \leq \mathcal{SROI}$, DL-APRIORI always *terminates*, guarantees to return all class expressions *modulo equivalence* satisfying the input constraints, i.e. it is *complete*, and only expressions satisfying the constraints, i.e. it is *correct*, see [19] for details. Completeness of DL-APRIORI ensures that no class expression (and thus no GCI) satisfying the input constraints is missed, i.e. all suitable class expressions (modulo equivalence) are returned. Of course, one should specify input constraints cautiously (which is rather easy to do) to avoid missing useful class expressions.

Correctness, completeness, and termination of DL-APRIORI can be proved for DLs with number restrictions $\geq k.C$ and $\leq k.C$, e.g. $\mathcal{SROIQ}$. This would require either making the function $\ell(C)$ (the length of a class expression C) dependent on k or introducing the parameter k_{max} which bounds k. Both ways regain the properties of DL-APRIORI for $\mathcal{SROIQ}$ but complicate the presentation.

6 Empirical Evaluation

We have implemented all presented techniques in a system called DL-MINER (see the source code[4] and demo interface[5]), as it is aimed at *mining*, i.e. constructing and evaluating, axioms in DLs and OWL, see [19]. We use Java (version 8.91), the OWL API [12] (version 3.5.0), and PELLET [21] (version 2.3.1) as a reasoner. All experiments are executed on the following machine: Linux Ubuntu 14.04.2 LTS (64 bit), Intel Core i5-3470 3.20 GHz, 8 GB RAM.

6.1 Mutual Correlations of Hypothesis Quality Measures

It is worthwhile to investigate whether the quality measures indeed capture different aspects of hypothesis quality. This can be clarified by examining their mutual correlations. We investigate the following research question:

RQ Do related measures strongly correlate? Do unrelated measures not correlate?

The experimental data consists of two corpora of ontologies. The first corpus, called *handpicked*, consists of 16 ontologies hand-picked from related work, e.g. from [7, 15]. The second corpus, called *principled*, comprises all BioPortal[6] ontologies taken from

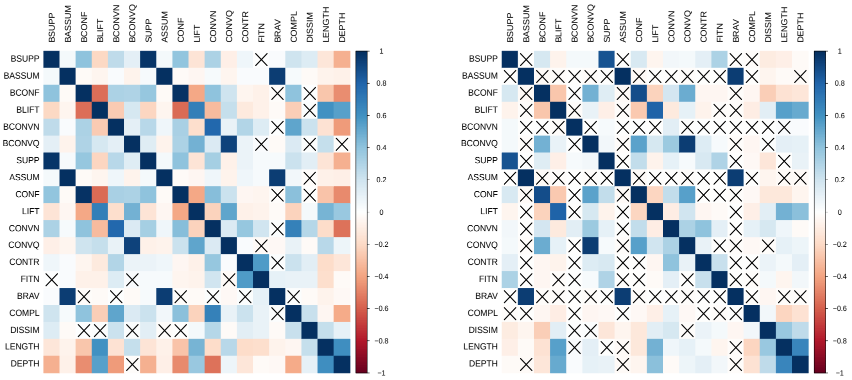
[4] https://github.com/slava-sazonau/dlminer

[5] http://www.dlminer.io

[6] http://bioportal.bioontology.org

[17] which contain some data (at least 100 individuals and 100 facts). It consists of 21 ontologies. In the handpicked and principled corpus, 9 and 14 ontologies, respectively, are at least as expressive as $\mathcal{ALC}$. With regard to the size, 3 and 0 ontologies, respectively, contain less than 100 individuals; 8 and 9 ontologies contain from 100 to 1000 individuals; 5 and 12 ontologies contain more than 1000 individuals. Both corpora are made publicly available [19]. We run the experiment on each corpus independently.

For each ontology $\mathcal{O}$, we run DL-APRIORI, see Algorithm 1, with $\mathcal{DL} := \mathcal{ALC}$, $\ell_{max} := 4$, $s_{min} := 10$. Since $\widetilde{\mathcal{O}}$ can contain many irrelevant terms, the seed signature is selected using the modular structure of the ontology as follows [20]: $\Sigma := crn(\mathcal{M}) \cup \{\top\}$, where $\mathcal{M} := \bot\text{-}module(\mathcal{O},\ crn(\mathcal{A}))$ [6] and $crn(\mathcal{O})$ returns the set of all class and property names occurring in $\mathcal{O}$. Then, we generate all possible GCIs (which can thus have length up to 8) from the constructed class expressions and OPIs with inverse properties and property chains. Using the proposed quality measures and measures from [20], we evaluate 500 randomly selected hypotheses per ontology. Then, we compute mutual correlations of the quality measures across all hypotheses in a corpus. We present the results, see Figure 1, in the form of a *correlation matrix*, which is a symmetric matrix of (Pearson's) correlation coefficients. For each correlation, we additionally run a statistical significance test with significance level 0.05.



(a) Handpicked corpus

(b) Principled corpus

Fig. 1: Mutual correlations of quality measures for handpicked (a) and principled (b) corpus: positive correlations are in blue, negative correlations are in red, crosses mark statistically insignificant correlations (significance level 0.05). The abbreviations are as follows: (B)SUPP – (basic) support, (B)ASSUM – (basic) assumption, (B)CONF – (basic) confidence, (B)LIFT – (basic) lift, (B)CONVN – (basic) negated conviction, (B)CONVQ – (basic) assumed conviction, CONTR – contradiction, FITN – fitness, BRAV – braveness, COMPL – complexity, DISSIM – dissimilarity.

First, we note that all main measures, except negated conviction for the principled corpus, strongly and positively correlate with their basic counterparts (please notice lines of dark blue squares parallel to the main diagonal in Figure 1). This result is expected because the basic measures are approximations of the respective main measures. All the differences are due to the presence of negative information in the ontologies. Another strong and positive correlation occurs between assumption and braveness which is also expected since these measures count (though differently) "guesses" of a hypothesis. Among other observations are the positive correlations between conviction and confidence, particularly in the principled corpus, that capture similar aspects of quality. Interestingly, lift positively correlates with length and depth, i.e. longer hypotheses are likely to be of higher quality as measured by lift. Thus, we can answer RQ as follows: related measures do correlate significantly, while unrelated measures mostly do not. In other words, the measures do capture different aspects of quality.

In addition, we have examined the acquired hypotheses by eyeballing them. Table 1 shows some high-quality hypotheses (please notice two property chains).

$\exists hasBond.\top \sqsubseteq \exists hasAtom.\top$
$AssociateProfessor \sqsubseteq \exists teaches.TeachingCourse$
$Patient \sqcap \exists hasShape.Irregular \sqsubseteq \exists hasDensity.Illdefined$
$\forall siblingof.Human \sqsubseteq Human$
$OKRunningLoan \sqsubseteq \exists hasLoanStatusValue.(\neg ProblemStatus)$
$married \circ hasChild \sqsubseteq hasChild$
$Movie \sqsubseteq \exists cast.Actor$
$BetaSugar \sqcap \exists hasRingForm.\top \sqsubseteq Pyranose$
$clinicallySimilar \circ hasSeverity \sqsubseteq hasSeverity$
$PlanetaryLayer \sqsubseteq \exists hasAstronomicalBody.\top$

Table 1: Examples of acquired hypotheses

6.2 A Case Study

In order to receive human feedback, we run a preliminary case study with one domain expert. The subject of the study is the ontology,[7] in the following called ntds, created using data from the US National Transgender Discrimination Survey[8] and curated by the domain expert. The ontology is in $\mathcal{SROIQ}$ and contains 169,058 individuals. We investigate the following research questions:

RQ1 What kinds of interesting hypotheses (if any) can we mine for the domain expert?
RQ2 Which measures (if any) are indicators of interestingness of a hypothesis?

To answer the research questions, we ask the domain expert to judge a hypothesis by validity and interestingness (which are different notions):

[7] The ontology is not public yet.

[8] http://www.ustranssurvey.org/

- *Validity* shows whether a hypothesis captures a general truth about the domain and can be perceived as an axiom to be added to the ontology.
- *Interestingness* shows how interesting a hypothesis is for a domain expert, i.e. evaluates her curiosity and attention that she pays to a hypothesis.

The domain expert assesses validity of a hypothesis by choosing one of the following three options: *"correct"*, *"wrong"*, *"don't know"*. Interestingness of a hypothesis is rated on the linear scale from 0 (lowest) to 4 (highest). We collect feedback using an online survey. To make a survey, we generate hypotheses as above. Since purely random sampling is likely to result in few (or no) promising hypotheses, we randomly select 30 hypotheses whose confidence exceeds 0.9 and 30 from all the rest to ensure variability of hypothesis quality in the survey which thus consists of 60 hypotheses.

The survey was completed by one domain expert. In the feedback that we received, the domain expert expressed interest in reviewing additional hypotheses and gave us *focus terms*, i.e. class and property names of a certain topic. We ran another survey of 60 hypotheses made analogously but using only the focus terms instead of the (almost) entire signature. The survey was completed by the same domain expert. Thus, 120 hypotheses were judged in total. In the following, we refer to the initial, unfocused survey as *Survey 1* and the follow-up, focused survey as *Survey 2*, see Table 2.

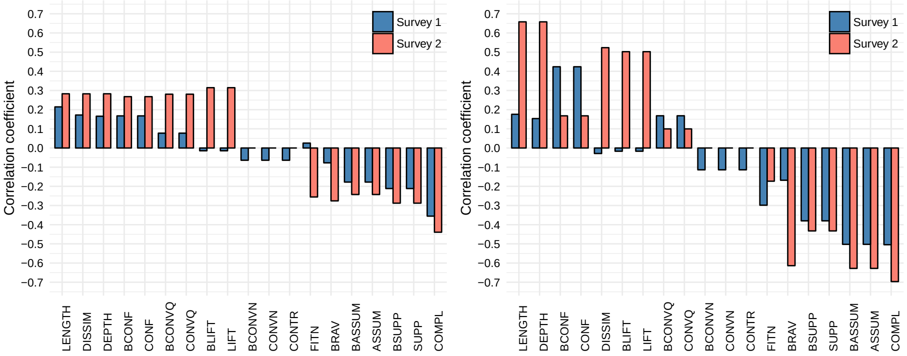
	Validity	**Interestingness**				
		0	*1*	*2*	*3*	*4*
Survey 1 (unfocused)	*Wrong*	6	11	30	-	-
	Don't know	-	1	-	2	4
	Correct	-	-	-	6	-
Survey 2 (focused)	*Wrong*	1	-	1	-	5
	Don't know	-	-	-	-	49
	Correct	-	-	-	-	4

Table 2: Assessment of hypotheses acquired for ntds ("-" denotes zero)

According to Table 2, in Survey 1, unknown and correct hypotheses are rated to be much more interesting than wrong ones: all of them, except one, have high values of interestingness. Amongst those, unknown hypotheses are marked to be the most interesting and, according to the expert's response, require further analysis. The results of Survey 2 are much different from the results of Survey 1. All hypotheses, except two, are marked by the highest value of interestingness, including wrong ones. Moreover, the domain expert informed us in her response that one of the wrong hypotheses not only indicated data bias but revealed an *error in the ontology*.

Thus, a mined hypothesis can be interesting regardless of its validity. More specifically, there are three kinds of interesting hypotheses: a correct hypothesis reflects known domain knowledge which is not yet captured in the ontology (enriches the TBox); an unknown hypothesis captures possibly true but yet unknown domain knowledge worthy of further enquiry; a wrong hypothesis indicates a modelling error or data bias. This answers RQ1 and confirms our observations made in [20].

We now turn our attention to RQ2, i.e. compare measures with expert's judgements. Figure 2 shows correlations between the quality measures and expert's judgements. Dissimilarity, confidence, length, and depth are the strongest positive indicators of validity, see Figure 2a. Lift turns from a non-indicator in Survey 1 to a positive indicator in Survey 2. The strongest negative indicators of validity are complexity, support, and assumption. The result that support is a negative indicator is rather unexpected, considering its definition. A possible explanation is that hypotheses with more evidence seem to be easier to reject for the domain expert because "counterexamples" are easier to recall.



(a) Validity

(b) Interestingness

Fig. 2: Correlations (in descending order) between hypothesis quality measures (abbreviated as in Figure 1) and expert's judgements: validity (a) and interestingness (b).

As Figure 2b shows, confidence is a positive indicator of interestingness in Survey 1. However, it is not in Survey 2: length, depth, dissimilarity, and lift have significantly stronger positive correlations. Thus, lift and dissimilarity turn from non-indicators of interestingness in Survey 1 to its positive indicators in Survey 2. Moreover, length and depth become strong positive indicators of interestingness showing that longer hypotheses are likely to be more interesting. This is not surprising because longer hypotheses are capable of capturing phenomena that shorter ones cannot capture, i.e. they are more *powerful*. Of course, a hypothesis can be "too long" for a domain expert to perceive. As for validity, the strongest negative indicators of interestingness are complexity, assumption, and support. Support appears to be a negative indicator of interestingness because hypotheses with high support are likely to be familiar to the expert since they reflect easily seen patterns in the data. Overall, the results in Figure 2 show that there is *no* single best indicator of hypothesis quality. This further supports our view that we need to consider multiple quality measures to identify promising hypotheses.

7 Future Work

The defined quality measures do not form the "complete list" of hypothesis quality measures. Clearly, there are other possible measures. In particular, additional rule measures can be adapted to OWL, e.g. cosine, Gini index, J-measure [10]. Such adaptation can respect the standard OWL semantics and its OWA using the procedure of translating rule measures into axiom measures presented in this paper.

Our implementation, DL-Miner, currently supports constructing GCIs for $\mathcal{ALC}$ (as well as complex property hierarchies and inverses). It relies on the availability of suitable refinement operators that are currently proposed for $\mathcal{ALC}$ [16]. In order to construct class expressions beyond $\mathcal{ALC}$ while preserving completeness, we need to design suitable refinement operators for more expressive DLs, e.g. $\mathcal{SROIQ}(\mathcal{D})$ [11].

Besides sequentially examining acquired hypotheses, a domain expert can potentially use them for *interactive ontology completion and debugging*. More specifically, approved hypotheses can be added to the ontology which is then used to mine new hypotheses and the step is repeated. Within such an iterative process, modelling errors can be identified using wrong hypotheses and then repaired. After that, a user can continue completing the ontology until it is sufficiently enriched or new errors are found. This scenario and additional investigations of the quality measures are subjects of further case studies.

8 Acknowledgements

We thank Amanda Hicks (the University of Florida) for participating in our case study and giving us valuable feedback and Michael Rutherford (the University of Arkansas for Medical Sciences) for translating data into OWL.

References

1. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proceedings of the 20th International Conference on Very Large Data Bases (VLDB'94). pp. 487–499. Morgan Kaufmann, Santiago de Chile, Chile (September 1994)
2. Alsubait, T., Parsia, B., Sattler, U.: Measuring similarity in ontologies: A new family of measures. In: Proceedings of the 19th International Conference on Knowledge Engineering and Knowledge Management (EKAW'14). Lecture Notes in Computer Science, vol. 8876, pp. 13–25. Springer, Linköping, Sweden (November 2014)
3. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, New York, NY, USA, 2nd edn. (2010)
4. Baader, F., Sertkaya, B., Turhan, A.: Computing the least common subsumer w.r.t. a background terminology. Journal of Applied Logic 5(3), 392–420 (2007)
5. Cohen, W.W., Hirsh, H.: Learning the CLASSIC description logic: Theoretical and experimental results. In: Proceedings of the 4th International Conference on Principles of Knowledge Representation and Reasoning (KR'94). pp. 121–133. Morgan Kaufmann, Bonn, Germany (May 1994)

6. Del Vescovo, C., Klinov, P., Parsia, B., Sattler, U., Schneider, T., Tsarkov, D.: Empirical study of logic-based modules: Cheap is cheerful. In: Proceedings of the 12th International Semantic Web Conference (ISWC'13). Lecture Notes in Computer Science, vol. 8218, pp. 84–100. Springer, Sydney, Australia (October 2013)
7. Fanizzi, N., D'Amato, C., Esposito, F.: DL-FOIL concept learning in Description Logics. In: Proceedings of the 18th International Conference on Inductive Logic Programming (ILP'08). Lecture Notes in Computer Science, vol. 5194, pp. 107–121. Springer, Prague, Czech Republic (2008)
8. Fleischhacker, D., Völker, J.: Inductive learning of disjointness axioms. In: Proceedings of the Confederated International Conference: On the Move to Meaningful Internet Systems (OTM'11). Lecture Notes in Computer Science, vol. 7045, pp. 680–697. Springer, Hersonissos, Crete, Greece (October 2011)
9. Galárraga, L.A., Teflioudi, C., Hose, K., Suchanek, F.: AMIE: Association rule mining under incomplete evidence in ontological knowledge bases. In: Proceedings of the 22nd International World Wide Web Conference (WWW'13). pp. 413–422. International World Wide Web Conferences Steering Committee / ACM, Rio de Janeiro, Brazil (2013)
10. Geng, L., Hamilton, H.J.: Interestingness measures for data mining: A survey. ACM Computing Surveys 38(3) (September 2006)
11. Grau, B.C., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P., Sattler, U.: OWL 2: The next step for OWL. Journal of Web Semantics 6(4), 309–322 (2008)
12. Horridge, M., Bechhofer, S.: The OWL API: A java API for OWL ontologies. Semantic Web 2(1), 11–21 (2011)
13. Horridge, M., Parsia, B., Sattler, U.: Laconic and precise justifications in OWL. In: Proceedings of the 7th International Semantic Web Conference (ISWC'08). Lecture Notes in Computer Science, vol. 5318, pp. 323–338. Springer, Karlsruhe, Germany (October 2008)
14. Iannone, L., Palmisano, I., Fanizzi, N.: An algorithm based on counterfactuals for concept learning in the Semantic Web. Applied Intelligence 26(2), 139–159 (April 2007)
15. Lehmann, J., Auer, S., Bühmann, L., Tramp, S.: Class expression learning for ontology engineering. Journal of Web Semantics 9(1), 71–81 (2011)
16. Lehmann, J., Hitzler, P.: Concept learning in Description Logics using refinement operators. Machine Learning 78(1-2), 203–250 (2010)
17. Matentzoglu, N., Parsia, B.: BioPortal Snapshot 27.01.2015. `https://doi.org/10.5281/zenodo.15667` (February 2015)
18. Ratcliffe, D., Taylor, K.L.: Closed-world concept induction for learning in OWL knowledge bases. In: Proceedings of the 19th International Conference on Knowledge Engineering and Knowledge Management (EKAW'14). Lecture Notes in Computer Science, vol. 8876, pp. 429–440. Springer, Linköping, Sweden (November 2014)
19. Sazonau, V.: General Terminology Induction in Description Logics. Tech. rep., The University of Manchester (2017), `https://doi.org/10.5281/zenodo.579593`
20. Sazonau, V., Sattler, U., Brown, G.: General Terminology Induction in OWL. In: Proceedings of the 14th International Semantic Web Conference (ISWC'15). Lecture Notes in Computer Science, vol. 9366, pp. 533–550. Springer, Bethlehem, PA, USA (October 2015)
21. Sirin, E., Parsia, B., Grau, B.C., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. Journal of Web Semantics 5(2), 51–53 (June 2007)
22. Völker, J., Niepert, M.: Statistical schema induction. In: Proceedings of the 8th Extended Semantic Web Conference (ESWC'11). Lecture Notes in Computer Science, vol. 6643, pp. 124–138. Springer, Heraklion, Crete, Greece (May 2011)
23. Zhu, M., Gao, Z., Pan, J.Z., Zhao, Y., Xu, Y., Quan, Z.: TBox learning from incomplete data by inference in BelNet+. Knowledge-Based Systems 75, 30–40 (2015)