Time-Capped Possibilistic Testing of OWL Axioms Against RDF Facts

Andrea G. B. Tettamanzi[1], Catherine Faron-Zucker[1], and Fabien Gandon[2]

[1] Univ. Nice Sophia Antipolis, I3S, UMR 7271, Sophia Antipolis, France,
andrea.tettamanzi@unice.fr, faron@polytech.unice.fr
[2] INRIA, Sophia Antipolis, France,
fabien.gandon@inria.fr

Abstract. Axiom scoring is a critical task both for the automatic enrichment/learning and for the automatic validation of knowledge bases and ontologies. We develop an axiom scoring heuristics based on possibility theory, which aims at overcoming some limitations of scoring heuristics based on statistical inference and working with an open-world semantics. Since computing the possibilistic score can be computationally quite heavy for some candidate axioms, we propose a method based on time capping to alleviate the computation of the heuristics without giving up the precision of the scores. We evaluate our proposal by applying it to the problem of testing SubClassOf axioms against the DBpedia RDF dataset.

Keywords: ontology learning, open-world assumption, possibility theory

1 Introduction

It is common practice, in the semantic Web, to put a strong emphasis on the construction or reuse of ontologies based on a principled conceptual analysis of a domain of interest, as a prerequisite for the organization of the Linked Open Data (LOD), much like a database schema must be designed before a database can be populated. While this approach is quite successful when applied to specific domains, it does not scale well to more general settings; it is aprioristic and dogmatic; it does not lend itself to a collaborative effort; etc. That is why an alternative, bottom-up, *grass-roots* approach to ontology and knowledge base creation better suits many scenarios: instead of postulating an *a priori* conceptualization of reality (i.e., an ontology) and requiring that our knowledge about facts complies with it, one can start from RDF facts and learn OWL 2 axioms.

Recent contributions towards the automatic creation of OWL 2 ontologies from large repositories of RDF facts include FOIL-like algorithms for learning concept definitions [**?**], statistical schema induction via association rule mining [**?**], and light-weight schema enrichment methods based on the DL-Learner framework [**?**,**?**]. All these methods apply and extend techniques developed within inductive logic programming (ILP) [**?**]. For a recent survey of the wider field of ontology learning, see [**?**].

On a related note, there exists a need for evaluating and validating ontologies, be they the result of an analysis effort or of a semi-automatic learning method. This need is witnessed by general methodological investigations [**?**,**?**] and surveys [**?**] and tools like OOPS! [**?**] for detecting pitfalls in ontologies.

Ontology engineering methodologies, such as METHONTOLOGY [**?**], distinguish two validation activities, namely verification (through formal methods, syntax, logics, etc.) and validation through usage. Whilst this latter is usually thought of as user studies, an automatic process of validation based on RDF data would provide a cheap alternative, whereby the existing linked data may be regarded as usage traces that can be used to test and improve the ontologies, much like log mining can be used to provide test cases for development in the replay approaches. Alternatively, one may regard the ontology as a set of integrity constraints and check if the data satisfy them, using a tool like Pellet integrity constraint validator (ICV), which translates OWL ontologies into SPARQL queries to automatically validate RDF data [**?**]. A similar approach also underlies the idea of test-driven evaluation of linked data quality [**?**]. To this end, OWL ontologies are interpreted under the closed-world assumption and the weak unique name assumption.

Yet this validation process may be seen from a reverse point of view: instead of starting from the *a priori* assumption that a given ontology is correct and verify whether the facts contained in an RDF base satisfy it, one may treat ontologies like hypotheses and develop a methodology to verify whether the RDF facts corroborate or falsify them. Ontology learning and validation are thus strictly related. They could even be seen as an agile and test-driven approach to ontology development, where the linked data is used as a giant test case library not only to validate the schema but even to suggest new developments.

Ontology learning and validation rely critically on (candidate) axiom scoring. In this paper, we will tackle the problem of testing a single, isolated axiom, which is anyway the first step to solve the problem of validating an entire ontology. Furthermore, to keep things reasonably simple, we will restrict our attention to subsumption axioms of the form $\mathsf{SubClassOf}(C\ D)$.

The most popular scoring heuristics proposed in the literature are based on statistical inference. We argue that such a probability-based framework is not always completely satisfactory. We propose an axiom scoring heuristics based on a formalization in possibility theory of the notions of logical content of a theory and of falsification, loosely inspired by Karl Popper's approach to epistemology, and working with an open-world semantics.

Some preliminary results [**?**] indicate that applying a possibilistic approach to the task of testing candidate axioms for ontology learning yields very promising results and hints that the same approach could be beneficial to ontology and knowledge base validation as well. At the same time, the proposed heuristics is much heavier, from a computational point of view, than the probabilistic scores it aims to complement. Fortunately, there is evidence (see [**?**] and Section **??** below) that the time it takes to test an axiom tends to be inversely proportional to its score. This suggests that time-capping the test might be an acceptable

additional heuristics to decide whether to accept or reject a candidate axiom, for an axiom which takes too long to test will likely end up having a very negative score. In this paper, we follow this suggestion and investigate the effectiveness of time-capped possibilistic testing of OWL axioms against the facts contained in an RDF repository. Our research question is, therefore: "Can time capping alleviate the computation of the proposed possibilistic axiom scoring heuristics without giving up the precision of the scores?" This paper is organized as follows: [TO DO]

2 Probability-Based Candidate Axiom Scoring

A statistics-based heuristics for the scoring of candidate axioms used in the framework of knowledge base enrichment [**?**] may be regarded essentially as scoring an axiom by an estimate of the probability that one of its logical consequences is confirmed (or, alternatively, falsified) by the facts stored in the RDF repository. Notice that every formula which logically follows from an axiom is both a potential falsifier (if it is contradicted by facts) and a potential confirmation (if it is verified by facts) for that axiom.

This probability-based approach relies on the assumption of a binomial distribution, which applies when an experiment (here, checking if a logical consequence of a candidate axiom is confirmed by the facts) is repeated a fixed number of times, each trial having two possible outcomes (conventionally labeled *success* and *failure*), the probability of success being the same for each trial, and the trials being statistically independent. We might call these experiment outcomes *confirmation*, if the observed fact agrees with the candidate axiom (success), and *counterexample* or *falsifier*, if the observed fact contradicts it (failure).

We will use the following notation throughout the paper: let ϕ be a candidate axiom; we will denote by u_ϕ the support or sample size for ϕ, i.e., the cardinality of the set of its logical consequences that will be tested in the RDF repository, by u_ϕ^+ the number of such consequences which are true (confirmations), and by u_ϕ^- the number of such consequences which are false (counterexamples). A few interesting properties of these three cardinalities are:

$$u_\phi^+ + u_\phi^- \leq u_\phi; \tag{1}$$

$$u_\phi^+ = u_{\neg\phi}^-, \quad u_\phi^- = u_{\neg\phi}^+, \quad u_\phi = u_{\neg\phi}. \tag{2}$$

As Bühmann and Lehmann point out [**?**], estimating the probability of confirmation of axiom ϕ just by $\hat{p}_\phi = u_\phi^+/u_\phi$ would be too crude and would not take the magnitude of u_ϕ into account. The parameter estimation must be carried out by performing a statistical inference.

One of the most basic analyses in statistical inference is to form a confidence interval for a binomial parameter p_ϕ (probability of confirmation of axiom ϕ), given a binomial variate u_ϕ^+ for support u_ϕ and a sample proportion $\hat{p}_\phi = u_\phi^+/u_\phi$. Most introductory statistics textbooks use to this end the Wald confidence interval, based on the asymptotic normality of $\hat{p}_\phi$ and estimating the standard

error. This $(1-\alpha)$ confidence interval for p_ϕ would be

$$\hat{p}_\phi \pm z_{\alpha/2}\sqrt{\hat{p}_\phi(1-\hat{p}_\phi)/u_\phi}, \tag{3}$$

where z_c denotes the $1-c$ quantile of the standard normal distribution.

However, the central limit theorem applies poorly to this binomial distribution with $u_\phi < 30$ or where $\hat{p}_\phi$ is close to 0 or 1. The normal approximation fails totally when $\hat{p}_\phi = 0$ or $\hat{p}_\phi = 1$. That is why Bühmann and Lehmann [**?**] base their probabilistic score on Agresti and Coull's binomial proportion confidence interval [**?**], an adjustment of the Wald confidence interval which goes: "Add two successes and two failures and then use Formula **??**." Such adjustment is specific for constructing 95% confidence intervals.

A remark about such approaches is in order. They only look for confirmations of ϕ, and treat the absence of a confirmation as a failure in the calculation of the confidence interval. This is like making an implicit closed-world assumption. In reality, definitions of explicit failures can be given (see, e.g., the one we will propose in Section **??**), but then the probability of finding a confirmation and the probability of finding a counterexample do not necessarily add to one, because there is a non-zero probability of finding neither a confirmation nor a counterexample for every logical consequence of an axiom, as stated in Equation **??**. For example, in the DBpedia dataset, there are 85 logical consequences of axiom SubClassOf(dbo:LaunchPad dbo:Infrastructure) which can be tested; 83 out of them are confirmations and 1 is a counterexample; the probability of finding neither a confirmation nor a counterexample for any logical consequence is thus 1.1765%. Bühmann and Lehmann's scoring method should thus be corrected in view of the open-world assumption, for example by using $\hat{p}^* = u_\phi^+/(u_\phi^+ + u_\phi^-)$ as the sample proportion instead of $\hat{p}$.

However, there is a more fundamental critique to the very idea of computing the likelihood of axioms based on probabilities. In essence, this idea relies on the assumption that it is possible to compute the probability that an axiom ϕ is true given some evidence e, for example $e =$ "ψ such that $\phi \models \psi$ is in the RDF repository", or $e =$ "ψ such that $\psi \models \neg\phi$ is in the RDF repository", or $e =$ "ψ such that $\phi \models \psi$ is *not* in the RDF repository", etc., which, by Bayes' formula, may be written as

$$\Pr(\phi \mid e) = \frac{\Pr(e \mid \phi)\Pr(\phi)}{\Pr(e \mid \phi)\Pr(\phi) + \Pr(e \mid \neg\phi)\Pr(\neg\phi)} \tag{4}$$

Therefore, in order to compute (or estimate) such probability, one should at least be able to estimate probabilities such as

- the probability that a fact confirming ϕ is added to the repository given that ϕ holds;
- the probability that a fact contradicting ϕ is added to the repository in error, i.e., given that ϕ holds;
- the probability that a fact confirming ϕ is added to the repository in error, i.e., given that ϕ does not hold;

– the probability that a fact contradicting ϕ is added to the repository given that ϕ does not hold.

Now, it is not hard to argue that the above probabilities may vary as a function of the concepts and properties involved. Let us take a subsumption axiom $\mathsf{SubClassOf}(C\ D)$ as an example. A fact confirming it is $D(a)$, with $C(a)$ in the dataset, whereas a fact contradicting it is $E(a)$, with $C(a)$ in the dataset and $\mathsf{DisjointClasses}(E\ D)$ in the ontology. Assuming that $\mathsf{SubClassOf}(C\ D)$ holds, we may suspect that $D(a)$ is more likely to be found in the repository if D is either very specific (and thus "closer" to a or very general (like $\mathsf{foaf{:}Person}$), and less likely if it is somewhere in the middle. This supposition is based on our expectations of what people are likely to say about a. For instance, an average person, if asked "what is this?" when pointing to a basset hound, is more likely to answer "a dog" or "an animal" than, say, "a carnivore" or "a mammal", which, on purely logical grounds, would be perfectly valid things to say about it [**?**]. There is thus an inherent difficulty with estimating the above probabilities, one which cannot be solved otherwise than by performing a large number of experiments, whose results, then, would be hard to generalize. By this argument, any axiom scoring method based on probability or statistics is doomed to be largely arbitrary and potentially fallacious.

Another key argument for rejecting a probabilistic approach in the specific context of axiom induction from an RDF dataset like DBpedia is that this dataset contains facts automatically extracted from Wikipedia, which is the result of a collaborative effort, whose coverage is not planned and subject to cultural and historical biases.[3] Therefore, there is no reason to assume that the facts contained in an RDF triple store be *representative* of all possible facts that could be recorded, unless that RDF store is the result of a planned and well-designed effort aimed at building a knowledge base providing uniform coverage of a given domain. Indeed, to use the number of facts supporting a hypothesis to estimate its probability one would have to make the very strong assumption that the finite set of facts in the RDF store is a representative sample of the infinite set of all "real" facts, whatever this means. Adopting a probabilistic approach whereas its assumptions are not fulfilled might lead to fallacious results.

3 A Possibilistic Candidate Axiom Scoring Heuristics

We propose an axiom scoring heuristics which captures the basic intuition behind the process of axiom discovery based on possibility theory which is weaker than probability theory.

[3] For example, at the level of pop music, the coverage of DBpedia is very much biased towards anglophone artists. Even in domains, such as geographical data, which one would expect to be much more uniform and extensive, it turns out that the coverage of Wikipedia is far from being uniform.

3.1 Possibility Theory

Possibility theory [**?**] is a mathematical theory of epistemic uncertainty. Given a finite universe of discourse Ω, whose elements $\omega \in \Omega$ may be regarded as events, values of a variable, possible worlds, or states of affairs, a possibility distribution is a mapping $\pi : \Omega \to [0, 1]$, which assigns to each ω a degree of possibility ranging from 0 (impossible, excluded) to 1 (completely possible, normal). A possibility distribution for which there exists a completely possible state of affairs ($\exists \omega^* : \pi(\omega^*) = 1$) is said to be *normalized*.

There is a similarity between possibility distribution and probability density. However, it must be stressed that $\pi(\omega) = 1$ just means that ω is a plausible (normal) situation and therefore should not be excluded. A degree of possibility can then be viewed as an upper bound of a degree of probability. Possibility theory is suitable to represent incomplete knowledge while probability is adapted to represent random and observed phenomena. We invite the reader to see [**?**] for more informations about the relationships between fuzzy sets, possibility, and probability degrees.

A possibility distribution π induces a *possibility measure* and its dual *necessity measure*, denoted by Π and N respectively. Both measures apply to a set $A \subseteq \Omega$ (or to a formula ϕ, by way of the set of its models, $A = \{\omega : \omega \models \phi\}$), and are defined as follows:

$$\Pi(A) = \max_{\omega \in A} \pi(\omega); \tag{5}$$

$$N(A) = 1 - \Pi(\bar{A}) = \min_{\omega \in \bar{A}} \{1 - \pi(\omega)\}. \tag{6}$$

A few properties of possibility and necessity measures induced by a normalized possibility distribution on a finite universe of discourse Ω are the following. For all subsets $A \subseteq \Omega$,

1. $\Pi(\emptyset) = N(\emptyset) = 0, \quad \Pi(\Omega) = N(\Omega) = 1;$
2. $\Pi(A) = 1 - N(\bar{A})$ (duality);
3. $N(A) > 0$ implies $\Pi(A) = 1, \quad \Pi(A) < 1$ implies $N(A) = 0.$

In case of complete ignorance on A, $\Pi(A) = \Pi(\bar{A}) = 1$.

3.2 Support of an Axiom

In Section **??**, we have introduced the notion of support or sample for a candidate axiom ϕ as the number of its logical consequences that will be tested in the RDF repository. We shall now define that notion more precisely.

Let BS be a finite set of *basic statements*, i.e., assertions, like the ones contained in an RDF repository, that may be tested by means of a SPARQL `ASK` query. We define the *content* of an axiom ϕ that we wish to evaluate as the set of its logical consequences, but we restrict it to basic statements, to ensure finiteness and testability:

$$\text{content}(\phi) = \{\psi : \phi \models \psi\} \cap \text{BS}. \tag{7}$$

The cardinality of content(ϕ) is finite, because BS is finite, and every formula $\psi \in$ content(ϕ) may be tested, because it is a basic statement. Now we can define the support of ϕ as the cardinality of content(ϕ):

$$u_\phi = \|\text{content}(\phi)\|. \tag{8}$$

3.3 Possibility and Necessity of an Axiom

The basic principle for establishing the possibility of a formula ϕ should be that the absence of counterexamples to ϕ in the RDF repository means $\Pi(\phi) = 1$, i.e., that ϕ is completely possible.

A hypothesis should be regarded as all the more *necessary* as it is explicitly supported by facts and not contradicted by any fact; and all the more *possible* as it is not contradicted by facts. In other words, given hypothesis ϕ, $\Pi(\phi) = 1$ if no counterexamples are found; as the number of counterexamples increases, $\Pi(\phi) \to 0$ strictly monotonically; $N(\phi) = 0$ if no confirmations are found; as the number of confirmations increases and no counterexamples are found, $N(\phi) \to 1$ strictly monotonically. Notice that a confirmation of ϕ is a counterexample of $\neg\phi$ and that a counterexample of ϕ is a confirmation of $\neg\phi$.

Here are a few postulates, based on our previous discussion, the possibility and necessity functions should obey:

1. $\Pi(\phi) = 1$ if $u_\phi^- = 0$;
2. $N(\phi) = 0$ if $u_\phi^- > 0$ or $u_\phi^+ = 0$;
3. let $u_\phi = u_\psi$; then $\Pi(\phi) > \Pi(\psi)$ iff $u_\phi^- < u_\psi^-$;
4. let $u_\phi = u_\psi$; then $N(\phi) > N(\psi)$ iff $u_\phi^+ > u_\psi^+$ and $u_\phi^- = 0$;
5. let $u_\phi = u_\psi = u_\chi$ and let $u_\psi^- < u_\phi^- < u_\chi^-$: then

$$\frac{\Pi(\psi) - \Pi(\phi)}{u_\phi^- - u_\psi^-} > \frac{\Pi(\phi) - \Pi(\chi)}{u_\chi^- - u_\phi^-},$$

   i.e., the first counterexamples found to an axiom should determine a sharper decrease of the degree to which we regard the axiom as possible than any further counterexamples, because these latter will only confirm our suspicions and, therefore, will provide less and less information;
6. let $u_\phi = u_\psi = u_\chi$ and $u_\psi^- = u_\phi^- = u_\chi^- = 0$, and let $u_\psi^+ < u_\phi^+ < u_\chi^+$: then

$$\frac{N(\phi) - N(\psi)}{u_\phi^+ - u_\psi^+} > \frac{N(\chi) - N(\phi)}{u_\chi^+ - u_\phi^+},$$

   i.e., in the absence of counterexamples, the first confirmations found to an axiom should determine a sharper increase of the degree to which we regard the axiom as necessary than any further confirmations, because these latter will only add up to our acceptance and, therefore, will provide less and less information.

A definition of Π and N which satisfies the above postulates is, for $u_\phi > 0$,

$$\Pi(\phi) = 1 - \sqrt{1 - \left(\frac{u_\phi - u_\phi^-}{u_\phi}\right)^2}; \tag{9}$$

$$N(\phi) = \sqrt{1 - \left(\frac{u_\phi - u_\phi^+}{u_\phi}\right)^2} \quad \text{if } \Pi(\phi) = 1, \text{ 0 otherwise.} \tag{10}$$

Notice that this is by no means the only possible definition, but we choose it because it is the simplest one (it derives from a quadratic equation; a linear equation would not satisfy all the postulates).

It may be shown that the above definition satisfies the duality of possibility and necessity, in that $N(\phi) = 1 - \Pi(\neg\phi)$ and $\Pi(\phi) = 1 - N(\neg\phi)$. As a matter of fact, we will seldom be interested in computing the necessity and possibility degrees of the negation of OWL 2 axioms, for the simple reason that, in most cases, the latter are not OWL 2 axioms themselves. For instance, while $C \sqsubseteq D$ is an axiom, $\neg(C \sqsubseteq D) = C \not\sqsubseteq D$ is not.

We combine the possibility and necessity of an axiom to define a single handy acceptance/rejection index (ARI) as follows:

$$\text{ARI}(\phi) = N(\phi) - N(\neg\phi) = N(\phi) + \Pi(\phi) - 1 \in [-1, 1]. \tag{11}$$

A negative $\text{ARI}(\phi)$ suggests rejection of ϕ ($\Pi(\phi) < 1$), whilst a positive $\text{ARI}(\phi)$ suggests its acceptance ($N(\phi) > 0$), with a strength proportional to its absolute value. A value close to zero reflects ignorance about the status of ϕ.

4 A Framework for Candidate Axiom Testing

We refer to the model-theoretic semantics of OWL 2 (as defined in [**?**]), which defines an interpretation $\mathcal{I}$ with a valuation function $\cdot^\mathcal{I}$ mapping OWL 2 expressions into elements and sets of elements of an interpretation domain $\Delta^\mathcal{I}$. We take the set of all the resources that occur in a given RDF store as $\Delta^\mathcal{I}$ and checking an axiom amounts to checking whether $\mathcal{I}$ is a model of the axiom. Also, calling linked data search engines like Sindice could virtually extend the interpretation domain to the whole LOD cloud.

However, unlike interpretation domains, RDF stores are incomplete and possibly noisy. The open-world hypothesis must be made; therefore, absence of supporting evidence does not necessarily contradict an axiom, and an axiom might hold even in the face of a few counterexamples (exceptions or possible mistakes). For example, out of 541 axioms of the form SubClassOf(C D) in the DBpedia ontology, 143 have an empty content (i.e., class C is empty) and 28 have at least one counterexample in DBpedia 3.9.[4]

[4] And one, namely SubClassOf(dbo:Person dbo:Agent), even has 76 counterexamples!

A general algorithm for testing all the possible OWL 2 axioms in a given RDF store is beyond the scope of this paper. Here, we will restrict our attention to atomic class expressions and ObjectComplementOf expressions, needed to test SubClassOf axioms. The model-theoretic semantics of expressions of the form ObjectComplementOf(C) ($\neg C$ in description logics syntax), where C denotes a concept expression (called *class expression* in OWL 2) is $\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$.

Now, let us define a mapping $Q(E, x)$ from OWL 2 expressions to SPARQL graph patterns, where E is an OWL 2 expression, x is a formal parameter which can be replaced by a SPARQL variable or an RDF term, such that the query `SELECT DISTINCT ?x WHERE` { $Q(E, \texttt{?x})$ } returns all the known instances of class expression E, which we will denote by $[Q(E, x)]$, i.e., the equivalent of $E^{\mathcal{I}}$, and the query `ASK` { $Q(E, a)$ } checks whether $E(a)$ is in the RDF base.

For an atomic concept A, $Q(A, \texttt{?x}) = \texttt{?x a } A$, where A is a valid IRI. For concept negation, things are slightly more complicated, for RDF does not support negation. The obvious definition

$$Q(\neg C, \texttt{?x}) = \{\ \texttt{?x ?p ?o . FILTER NOT EXISTS } Q(C, \texttt{?x})\ \}, \tag{12}$$

has the problem of treating negation as failure, like in databases, where the closed-world assumption is made. Since we want to preserve an open-world semantics, $Q(\neg C, x)$ should be defined differently, as the union of the concepts that are disjoint from C. One might try to express this as the set of individuals x that are instances of a concept C' such that no individual $z \in C^{\mathcal{I}}$ is an instance of C', yielding the query

$$\begin{aligned} Q(\neg C, \texttt{?x}) = \{\ & \texttt{?x a ?dc .} \\ & \texttt{FILTER NOT EXISTS } \{\ \texttt{?z a ?dc .}\ \ Q(C, \texttt{?z})\ \}\ \}, \end{aligned} \tag{13}$$

where `?z` is a variable that does not occur anywhere else in the query. This translation is conceptually more satisfactory than the one in Equation **??**, but it just pushes the problem one step further, because this way of testing whether two concepts are disjoint is based on negation as failure too. The only way to be certain that two classes are disjoint would be to find an axiom to this effect in the ontology:

$$Q(\neg C, \texttt{?x}) = \{\ \texttt{?x a ?dc . ?dc owl:disjointWith } C\ \}, \tag{14}$$

otherwise, either we find an individual which is an instance of both classes, and thus we know the two classes are not disjoint, or we don't, in which case the two classes may or may not be disjoint. The fact is, very few DisjointClasses axioms are currently found in existing ontologies. For example, in the DBpedia ontology, the query `SELECT ?x ?y { ?x owl:disjointWith ?y }` executed on November 22, 2013 returned 17 solutions only.

To compare these three alternative definitions of $Q(\neg C, \texttt{?x})$, we may refer to the diagram in Figure **??**. We wish to estimate the actual extent of $(\neg C)^{\mathcal{I}}$. Clearly, $Q(C, \texttt{?x})$ (in dark grey) underestimates the real extent of $C^{\mathcal{I}}$ (in light gray). Therefore, we may say that Equation **??** overestimates the real extent of

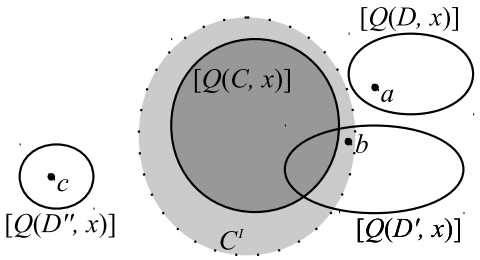


Fig. 1. A schematic illustration of the heuristics used to capture negation under the open world assumption. D'' is a concept which is declared to be disjoint with C in the RDF repository.

$(\neg C)^{\mathcal{I}}$, in the sense that it will regard as instances of $\neg C$ all individuals a for which "a `a` C" is not found in the RDF repository.

Now, if b is such that "b `a` C" is not known, but "b `a` D'" is known for some class D' and some instances of D are known to be also instances of C, then it might well be that b is an instance of C as well, although we do not know. If, however a is such that "a `a` C" is not known, but "b `a` D" is known for some class D but no instance of D is known that is also an instance of C, then we are more likely to believe that a is not an instance of C. Therefore Equation **??** regards as instances of $\neg C$ fewer individuals, those for which it is highly likely that they do not belong in C. It might still overestimate the extent of $(\neg C)^{\mathcal{I}}$, but much less than Equation **??**. In fact, it might even underestimate it, as far as we know.

On the other hand, it is certain that Equation **??** will underestimate $(\neg C)^{\mathcal{I}}$, to the point that it will equate it with the empty set if no triple of the form "D'' `owl:disjointWith` C" is declared in the RDF repository. Furthermore, it might well be that an individual is an instance of $\neg C$ even though it is not an instance of a class disjoint with C!

To sum up, Equation **??** is too optimistic, Equation **??** too pessimistic, and Equation **??** somewhere in the middle. Following the old adage "virtue stands in the middle", adopting Equation **??** looks like a sensible choice.

We will end this section by arguing that a suitable definition of confirmation to adopt in this framework is Scheffler and Goodman's *selective confirmation* [**?**], which characterizes a confirmation as a fact not simply satisfying an axiom, but, further, favoring the axiom rather than its contrary. For instance, the occurence of a black raven *selectively confirms* the axiom $\mathsf{Raven} \sqsubseteq \mathsf{Black}$ because it both confirms it and fails to confirm its negation, namely that there exist ravens that are not black. On the contrary, the observation of a green apple does not contradict $\mathsf{Raven} \sqsubseteq \mathsf{Black}$, but it does not disconfirm $\mathsf{Raven} \not\sqsubseteq \mathsf{Black}$ either; therefore, it does not selectively confirm $\mathsf{Raven} \sqsubseteq \mathsf{Black}$.

5 Evaluation on Subsumption Axiom Testing with Time Capping

5.1 Heuristics Implementation

The semantics of subsumption axioms of the form $\mathsf{SubClassOf}(C\ D)$ ($C \sqsubseteq D$ in description logic syntax) is $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, which may also be written $x \in C^{\mathcal{I}} \Rightarrow x \in D^{\mathcal{I}}$. The content of such axioms may thus be defined as

$$\text{content}(C \sqsubseteq D) = \{C(a) \text{ in the RDF store}\}, \tag{15}$$

and its support $u_{C \sqsubseteq D}$ can be computed with the following SPARQL query:

$$\texttt{SELECT (count(DISTINCT ?x) AS ?u) WHERE } \{Q(C, \texttt{?x})\}. \tag{16}$$

In order to compute $ARI(C \sqsubseteq D)$, we must provide a computational definition of $u^{+}_{C \sqsubseteq D}$ and $u^{-}_{C \sqsubseteq D}$. We start with the following statements:

- confirmations are individuals i such that $i \in [Q(C, x)]$ and $i \in [Q(D, x)]$;
- counterexamples are individuals i such that $i \in [Q(C, x)]$ and $i \in [Q(\neg D, x)]$.

This may be translated into SPARQL queries to compute $u^{+}_{C \sqsubseteq D}$ and $u^{-}_{C \sqsubseteq D}$:

$$\begin{array}{l}\texttt{SELECT (count(DISTINCT ?x) AS ?numConfirmations)} \\ \texttt{WHERE } \{\ Q(C, \texttt{?x})\ Q(D, \texttt{?x})\ \}\end{array} \tag{17}$$

and

$$\begin{array}{l}\texttt{SELECT (count(DISTINCT ?x) AS ?numCounterexamples)} \\ \texttt{WHERE } \{\ Q(C, \texttt{?x})\ Q(\neg D, \texttt{?x})\ \}\end{array} \tag{18}$$

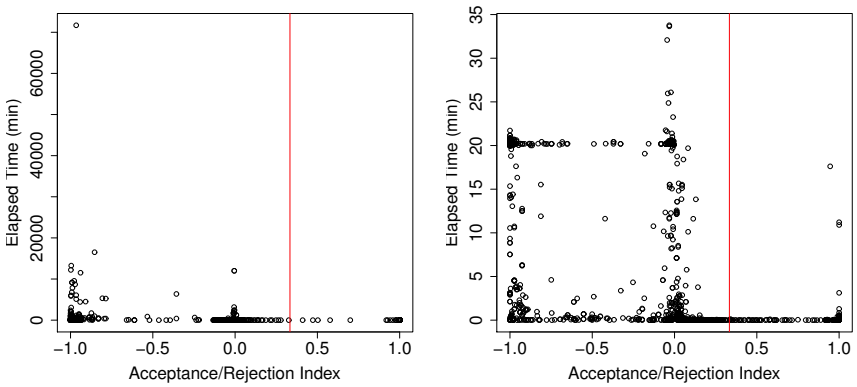
respectively. Notice that an $i \in [Q(C, x)]$ such that $i \notin [Q(D, x)]$ does not contradict $C \sqsubseteq D$, because it might well be the case that the assertion "i `a` D" is just missing. Likewise, an $i \in [Q(\neg D, x)]$ such that $i \in [Q(\neg C, x)]$ will not be treated as a confirmation, based on our choice to regard as evidence in favor of a hypothesis only selective confirmations.

5.2 Experimental Protocol

We evaluated the proposed scoring heuristics by performing tests of subsumption axioms using DBpedia 3.9 in English as the reference RDF fact repository. In particular, on April 27, 2014, we downloaded the DBpedia dumps of English version 3.9, generated in late March/early April 2013, along with the DBpedia ontology, version 3.9. This local dump of DBpedia, consisting of 812,546,748 RDF triples, has been bulk-loaded into Jena TDB and a prototype for performing axiom tests using the proposed method has been coded in Java, using Jena ARQ and TDB to access the RDF repository.

We systematically generated and tested subsumption axioms involving atomic classes only according the following protocol: for each of the 442 classes C referred to in the RDF repository, we construct all axioms of the form $C \sqsubseteq D$ such that C and D share at least one instance. Classes D are obtained with query $\texttt{SELECT DISTINCT ?D WHERE } \{Q(C, \texttt{?x})\ \texttt{?x a ?D}\}$.

Experiments have been performed on two machines:



(a) (b)

Fig. 2. Plots of the time taken for testing the systematically generated SubClassOf axioms without time capping (a) and with a 20-minute time cap (b), as a function of ARI. The red line shows the acceptance threshold $\text{ARI}(\phi) > 1/3$.

- a Fujitsu CELSIUS workstation equipped with twelve six-core Intel Xeon CPU E5-2630 v2 processors at 2.60GHz clock speed, with 15,360 KB cache each, 128 GB RAM, 4 TB of disk space with a 128 GB SSD cache, under the Ubuntu 12.04.4 LTS 64-bit operating system;
- an HP portable PC equipped with four two-cores Intel® Core™ i7-4600U CPUs at 2.10GHz clock speed, with a 4,096 KB cache, 16 GB RAM, 128 GB of disk space, under the Fedora 64-bit Linux operating system.

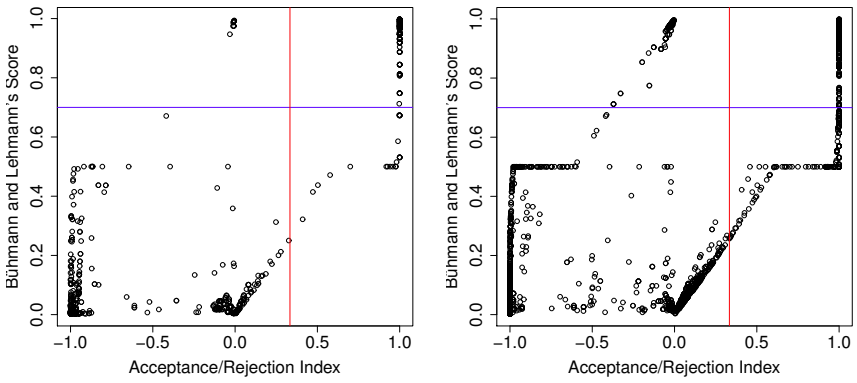
The former was used to test 399 axioms without time capping, before being attacked and becoming unusable. The latter, much less powerful, was then used to obtain all the results with time capping.

5.3 Results without Time Capping

We managed to test 399 axioms without time capping. Figure **??**a compares the results obtained in this preliminary experiment to the probabilistic score proposed in [**?**]. As already discussed in [**?**], the proposed acceptance-rejection index is more accurate than the probabilistic score. However, its increased accuracy comes at a high computational cost.

5.4 Results with Time Capping

The results of our first experimentation described in Section **??** show that the time it takes to test an axiom tends to be inversely proportional to its score (see Figure **??**a): an axiom which takes too long to test will likely end up having a



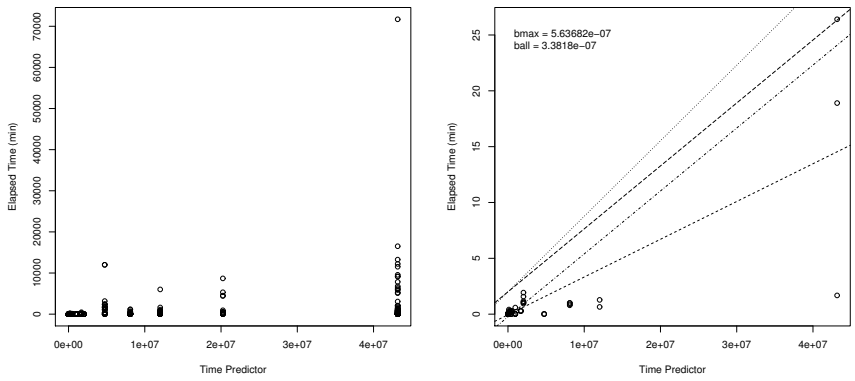
(a) (b)

Fig. 3. Plots comparing the acceptance/rejection index and the probability-based score used in [**?**] on axioms tested without time capping (a) and with a 20-minute time cap (b). The red line shows the acceptance threshold $\mathrm{ARI}(\phi) > 1/3$; the blue line the acceptance threshold of 0.7 for the probabilistic score.

very negative score. More precisely the time it takes to test a SubClassOf axiom of the form $C \sqsubseteq D$ is proportional to the product of the cardinality of C and the number of classes that have at least a known instance in common with C, as shown in Figure **??**. We will call this product the *time predictor*. As a result, we experimentally fixed to 20 min the threshold to time-cap the SPARQL queries to compute $u^{+}_{C \sqsubseteq D}$ and $u^{-}_{C \sqsubseteq D}$ to decide whether to accept or reject a candidate axiom $C \sqsubseteq D$.

For this new round of experiments, we decided to test the axioms with a lower time predictor first in order to get as much tested axioms as possible. Thanks to the greatly reduced overhead (cf. Figure **??**), we were able to test 2,425 axioms at the time of writing (the experiment is still running). The results are shown in Figure **??**b.

There are 98 axioms that were tested both with and without time capping; the outcome of the test is different on just four of them, namely dbo:Road $\sqsubseteq$ gml:_Feature, dbo:Library $\sqsubseteq$ gml:_Feature, schema:School $\sqsubseteq$ dbo:Agent, and schema:School $\sqsubseteq$ gml:_Feature. That represents an error rate of 4.1%. If we take into account the dramatic improvement in terms of speed, this looks like a very reasonable price to pay in terms of accuracy degradation. In addition, it should be observed that, by construction, the errors are all in the same direction, i.e., some axioms which should be accepted are in fact rejected: at least, this is a conservative heuristics, since it does not generate false positives.



(a) (b)

Fig. 4. Plots of the time taken to test the axioms as a function of the time predictor without time capping: (a) all axioms; (b) just the axioms that were accepted.

6 Conclusion

We have presented a possibilistic axiom scoring heuristics which is a viable alternative to statistics-based heuristics. We have tested it by applying it to the problem of testing SubClassOf axioms against the DBpedia database. We have also proposed an additional heuristics to greatly reduce its computational overhead, consisting of setting a time-out on the test of each axiom.

Our results, albeit preliminary, strongly support the validity of our hypothesis that it is possible to alleviate the computation of the ARI without loosing too much in terms of accuracy.

In addition, a human evaluation of the axioms scored by the system shows that most of the axioms accepted by mistake are inverted subClassOf relations between concepts (e.g. dbo:Case $\sqsubseteq$ dbo:LegalCase instead of dbo:LegalCase $\sqsubseteq$ dbo:Case). This occurs when counterexamples are missing (all instances of a class are instances of the other class too and the two axioms are positively scored). Other mistakes are on axioms involving vague concepts (e.g., it seems that anything that can appear on a map could be typed with gml:_Feature and therefore many classes should be subclasses of it, but it is not clear wether this is correct or not) or used in a more general sense thant it could be expected (e.g., dbo:PokerPlayer $\sqsubseteq$ dbo:Athlete; this is not really a mistake in the sense that there are several other such concepts involving dbo:Athlete). Another example of mistakes on is the use of a concept in at least two senses, e.g dbo:Library designating both a building and an institution. Other frequent mistakes are on axioms involving a concept both used as a zoological class name, a taxon, and therefore marked as subclass of dbp:Species, and as a set of animals, and therefore subclass of dbo:Animal and dbo:Eukaryote. The same confusion between

the instance level and the ontological level explain the results on axioms involving skos:Concept.

These considerations confirm the interest of using axiom scoring heuristics like ours not only to learn axioms from the LOD, but also to drive the validation and debugging of ontologies and RDF datasets.