An Ontology-driven Probabilistic Soft Logic Approach to Improve NLP Entity Annotations

Marco Rospocher

Fondazione Bruno Kessler – IRST
Via Sommarive 18, Trento, I-38123, Italy
rospocher@fbk.eu

Abstract. Many approaches for Knowledge Extraction and Ontology Population rely on well-known Natural Language Processing (NLP) tasks, such as Named Entity Recognition and Classification (NERC) and Entity Linking (EL), to identify and semantically characterize the entities mentioned in natural language text. Despite being intrinsically related, the analyses performed by these tasks differ, and combining their output may result in NLP annotations that are implausible or even conflicting considering common world knowledge about entities. In this paper we present a Probabilistic Soft Logic (PSL) model that leverages ontological entity classes to relate NLP annotations from different tasks insisting on the same entity mentions. The intuition behind the model is that an annotation likely implies some ontological classes on the entity identified by the mention, and annotations from different tasks on the same mention have to share more or less the same implied entity classes. In a setting with various NLP tools returning multiple, confidence-weighted, candidate annotations on a single mention, the model can be operationally applied to compare the different annotation combinations, and to possibly revise the tools' best annotation choice. We experimented applying the model with the candidate annotations produced by two state-of-the-art tools for NERC and EL, on three different datasets. The results show that the joint " a posteriori" annotation revision suggested by our PSL model consistently improves the original scores of the two tools.

1 Introduction

The problem of identifying and semantically characterizing the entities mentioned in a natural language text has been extensively investigated over the years. Several Natural Language Processing (NLP) tasks have been defined and investigated. Some of them, such as Named Entity Recognition and Classification (NERC) and Entity Linking (EL), directly tackle the problem of recognizing the entities in a text, characterizing them according to some predefined categories (NERC) or disambiguating them with respect to a reference Knowledge Base (EL). Other tasks, though conducting different analyses than explicitly identifying entities, may also contribute to their characterization: an example is Semantic Role Labeling (SRL), the task of identifying the role (e.g., seller, buyer, goods) of words, and thus also entities, in a sentence.

Several tools have been proposed to effectively perform these tasks. However, despite the good performances on the single tasks, when combining them, as for instance

in Knowledge Extraction frameworks (e.g., NewsReader [1], PIKES [2]), the output of these tools may result in unlikely or even contradictory information. Consider for instance the sentence "Lincoln is based in Michigan.". Here, the entity mention "Lincoln" refers to the company "Lincoln Motor Company". [1] However, using two state-of-the-art NLP tools, one for NERC (Stanford NER[2]) and one for EL (DBpedia Spotlight[3]), the first correctly identifies "Lincoln" as an organization, while the second wrongly links it to the DBpedia entity corresponding to "Abraham Lincoln". As another example, on the sentence "San Jose is one of the strongest hockey team.", the NERC tool wrongly identifies the mention "San Jose" as a location, while the EL one correctly links it to the entity "San Jose Sharks".[4]

In this paper we present PSL4EA, a novel approach based on Probabilistic Soft Logic (PSL) that, leveraging ontological background knowledge, enables relating the entity annotations produced by different NLP tools on the same entity mentions, and to assess their coherence. In a nutshell, given the mention of an entity in a text, the proposed PSL model enables:

1. to express the ontological entity classes of the background knowledge likely implied by the involved annotations; and,
2. to assess the coherence of the annotations, as the extent to which they share the same implied ontological entity classes.

If available, information on the confidence of the tools on the provided annotations can be included in the model, and it is taken in consideration when assessing the coherence of the annotations. As a consequence, if the considered tools provide multiple *candidate* annotations — i.e., alternative annotations on the same mention, weighted with a confidence score — the model can be applied to select the combination of annotations (one for each tool) that maximizes the annotation coherence in light of their confidences, possibly overruling the best candidate choices of the tools.

We present the creation of the model for a concrete scenario involving NERC and EL annotations, leveraging YAGO [3] as background ontological knowledge. To assess the effectiveness of the approach, we applied the model on the candidate annotations produced by two state-of-the-art tools for NERC (Stanford NER [4]) and EL (DBpedia Spotlight [5]), on three reference evaluation datasets (AIDA CoNLL-YAGO [6], MEANTIME [7], TAC-KBP [8]), showing experimentally that the joint annotation revision suggested by the model consistently improves the scores of the considered tools. We also discuss how to extend the model to (entity) annotations beyond NERC and EL.

While PSL was previously applied [9] for Knowledge Graph Identification (i.e., deriving a knowledge graph from triples automatically extracted from text), to the best of our knowledge this is the first work exploiting this powerful framework, with ontological knowledge, to assess the coherence and to improve NLP entity annotations. Differently from other approaches that have investigated jointly trained NERC and EL

[1] https://en.wikipedia.org/wiki/Lincoln_Motor_Company (last accessed on April 1, 2018)
[2] http://nlp.stanford.edu:8080/corenlp/ (last accessed on April 1, 2018)
[3] http://demo.dbpedia-spotlight.org/ (last accessed on April 1, 2018)
[4] https://en.wikipedia.org/wiki/San_Jose_Sharks (last accessed on April 1, 2018)

models (e.g., [10,11]), PSL4EA works "a posteriori" on the annotations for the considered tasks, leveraging ontological knowledge. This makes the approach applicable to many existing NLP tools for entity annotation.

The paper is structured as follows. Section 2 briefly recaps the main aspects of Probabilistic Soft Logic. Section 3 presents our novel, ontology-driven PSL approach for jointly assessing the coherence and revising NLP annotations. Section 4 reports the empirical assessment of using PSL4EA to improve the performances of Stanford NER and DBpedia Spotlight on three reference datasets for NERC and EL. Section 5 discusses some aspects of the proposed approach, including the extension to other (entity) annotation types (e.g., Semantic Role Labeling). Section 6 compares with relevant related works, while Section 7 concludes.

2 Background on Probabilistic Soft Logic

Probabilistic Soft Logic (PSL) [12] is a powerful, general-purpose probabilistic programming language that enables users to specify rich probabilistic models over continuous variables. It is a statistical relational learning framework that uses first-order logic to compactly define Markov networks, and comes with methods for performing efficient probabilistic inference for the resulting models. Differently from other related works, variables in PSL are continuous in the range [0, 1] rather than binary.

A PSL program consists of a PSL model and some data. A PSL model is composed of a set of weighted if-then, first-order logic rules, such as:

$$1.2: \ \mathsf{WorksFor}(b,c) \ \& \ \mathsf{BossOf}(b,e) \rightarrow \mathsf{WorksFor}(e,c) \tag{1}$$

stating that employees are likely to work for the same company as their boss. Here: 1.2 is the *weight* of the rule; b, c, and e are universally-quantified *variables*; WorksFor and BossOf are *predicates*; $\mathsf{WorksFor}(b,c)$ is an *atom*; the part on the left of the arrow is called *body*, while the part on the right is named *head*. The *grounding* of a rule is the substitution of variables in the rule's atoms with constants (e.g., the ground atom $\mathsf{WorksFor}(B,C)$ results by assigning constants B and C to variables b and c), and ground atoms take a *soft-truth value* in the range [0, 1].

To compute soft-truth values for logical formulas, PSL adopts *Lukasiewicz t-norm* and *co-norm* to provide a relaxation of the logical conjunction ($\wedge$), disjunction ($\vee$) and negation($\neg$). Let I (*interpretation*) be an assignment of soft-truth values to ground atoms, and let a_1 and a_2 be two ground atoms, we have:

$$\begin{aligned} I(a_1) \wedge I(a_2) &= max\{I(a_1) + I(a_2) - 1, 0\} \\ I(a_1) \vee I(a_2) &= min\{I(a_1) + I(a_2), 1\} \\ \neg I(a_1) &= 1 - I(a_1) \end{aligned} \tag{2}$$

Given a rule r, with body r_b and head r_h, r is said to be *satisfied* if and only if $I(r_b) \leq I(r_h)$. For instance, with $I(\mathsf{WorksFor}(B,C)) = 0.6$, $I(\mathsf{BossOf}(B,E)) = 0.6$ and $I(WorksFor(E,C)) = 0.5$, rule (1) is satisfied. Otherwise, PSL defines a *distance to satisfaction*

$d(r) = max\{0, I(r_b) - I(r_h)\}$, capturing how far a rule is from being satisfied. For instance, with $I(\mathsf{WorksFor}(B,C)) = 0.8$, $I(\mathsf{BossOf}(B,E)) = 0.9$ and $I(\mathsf{WorksFor}(E,C)) = 0.3$, rule (1) has a distance to satisfaction equal to 0.4.

By leveraging the distance to satisfaction, PSL defines a *probability distribution*

$$f(I) = \frac{1}{Z} \exp\left[- \sum_{r \in R} w_r d(r)^p \right] \tag{3}$$

over interpretations, where Z is a normalization constant, w_r is the weight of rule r, R is the set of all rules, and $p \in \{1,2\}$ identifies a linear or quadratic loss function.

Different inference tasks can be investigated on a PSL program. One relevant for this paper is Most Probable Explanation (MPE) inference and corresponds to finding the overall interpretation with the maximum probability (i.e., the most likely soft-truth values of unknown ground atoms) given a set of known ground atoms. That is, the interpretation that minimizes the distance to satisfaction by trying to satisfy all rules as much as possible.

3 A PSL model for NERC and EL

In this section, we outline PSL4EA (*PSL for Entity Annotations*), the PSL model we propose to jointly assess the coherence, and possibly revise, the entity annotations produced for some NLP tasks. We present the approach focusing on the two typical NLP tasks for entity annotation,[5] namely:

- **Named Entity Recognition and Classification (NERC)**: the task of labeling mentions in a text that refer to named things such as persons, organizations, etc., and choosing their type according to some predefined categories (e.g., PER, ORG);
- **Entity Linking (EL)**: the task of aligning an entity mention in a text to its corresponding entity in a Knowledge Base (e.g., YAGO [3], DBpedia [13]).

The approach is based on the assumption that, given the mention of a named entity in a text, the entity can be typed with all its ontological classes[6] defined in a given Knowledge Base K, our ontological background knowledge.

We discuss the general case where we have multiple alternative annotations (*candidates*) for each task on the same mention. That is, given a mention M, and assuming to have n_N NERC and n_E EL candidates on M, we indicate with $A^N_1, \ldots, A^N_{n_N}$ and $A^E_1, \ldots, A^E_{n_E}$ the NERC and EL candidates, while $w(M, A^i_j)$ indicates the confidence score assigned to annotation A^i_j on mention M.

The PSL model comprises two parts: the first one exploiting the relation between NLP annotations and ontological classes from the background knowledge; and, the second one capturing the coherence of the NLP annotations via these ontological classes.

[5] The extension to other types of entity annotations is discussed later in Section 5.

[6] Typically, an entity is typed with many ontological classes, cf. rdf:type assertions from YAGO on http://dbpedia.org/page/Lincoln_Motor_Company (last accessed on April 1, 2018)

3.1 Classes implied by NLP annotations

The intuition behind this part of the model is that given an annotation for an entity mention, if this annotation is compatible with some ontological classes of the background knowledge, then the ontological classes characterizing the entity should be among them.

Given a mention M and a NERC annotation A_i^N, we define the rule:

$$w(M,A_i^N):\ \mathsf{Ann}_N(M,A_i^N)\ \&\ \mathsf{ImpCl}_N(A_i^N,c)\rightarrow \mathsf{ClAnn}_N(M,A_i^N,c) \tag{4}$$

where:

- $\mathsf{Ann}_N(x,y)$ relates a mention x to a NERC annotation y. The grounding of the predicate has value 1 if the mention is annotated with that NERC type, 0 otherwise;
- $\mathsf{ImpCl}_N(x,y)$ captures to which extent seeing a certain NERC annotation x implies that the entity is typed with the ontological class y. This quantity can be learned from gold data (see Section 3.1);
- $\mathsf{ClAnn}_N(x,y,z)$ captures that mention x corresponds to an entity that is instance of class z due to annotation y.

For the first two predicates, the soft-truth value of the atoms is known (input data), while the value for the ground atoms of ClAnn_N has to be determined by the model. Furthermore, the rule is partly grounded, i.e., the only variable is the ontological class c. Given a mention M on which we have n_N NERC candidates, we have n_N such rules, one for each candidate, weighted according to the corresponding confidence score.

Similarly, given a mention M and an EL annotation A_i^E, we define the rule:

$$w(M,A_i^E):\ \mathsf{Ann}_E(M,A_i^E)\ \&\ \mathsf{ImpCl}_E(A_i^E,c)\rightarrow \mathsf{ClAnn}_E(M,A_i^E,c) \tag{5}$$

where $\mathsf{Ann}_E(x,y)$, $\mathsf{ImpCl}_E(x,y)$, $\mathsf{ClAnn}_E(x,y,z)$ are defined analogously to the NERC case. Again, note that we have n_E such rules.

Determining ImpCl_N and ImpCl_E $\mathsf{ImpCl}_N(x,y)$ captures the "likelihood" that a certain NERC annotation implies an ontological class. The higher the soft-truth value for a given NERC type x and ontological class y, the higher are the chances that if an entity mention is NERC annotated with x, than the entity is an instance of class y. To determine $\mathsf{ImpCl}_N(x,y)$ we assume the availability of a gold standard corpus G where each entity mention is annotated with both (i) its NERC type and (ii) all its ontological classes from the background knowledge, or, alternatively, an annotation deterministically alignable to them (e.g., an EL annotation, with the entity typed according to the ontological classes). We then use G as data for another PSL program, with rules:

$$\begin{aligned} &1.0:\mathsf{Gold}_N(m,t)\ \&\ \mathsf{ImpCl}_N(t,c)\rightarrow \mathsf{Gold}_C(m,c)\\ &1.0:\mathsf{Gold}_N(m,t)\ \&\ \neg\mathsf{ImpCl}_N(t,c)\rightarrow \neg\mathsf{Gold}_C(m,c)\end{aligned} \tag{6}$$

where $\mathsf{Gold}_N(m,t)$ is 1 if mention m is annotated with t in G, and 0 otherwise, while $\mathsf{Gold}_C(m,c)$ is 1 if c is one of the ontological classes of the entity denoted by the mention m, and 0 otherwise. That is, the soft-truth values of the ground atoms of Gold_C and

Gold_N are known, while the value for the ground atoms of ImpCl_N has to be determined by this specific model. Note that two rules are used in (6): they respectively account for the cases where mentions, NERC annotated with a type t, are annotated (i) also with class c, and (ii) not with class c, so to properly capture the "likelihood" that a NERC type implies some classes but not others.

The model has to estimate ImpCl_N for all possible NERC types and ontological classes. While all possible NERC types are typically occurring in G, some very specific class c of the background knowledge K may be observed few times (or even not at all) in it. However, especially for coarse-grain NERC types such as the classical 4-type (PER, ORG, LOC, MISC) model, there is little benefit in considering rarely observed, very specific ontological classes. We thus restrict our attention to popular classes, those observed at least $\bar{n}$ times (an hyperparameter of our approach) in G, typically general classes in the class taxonomy, filtering out any remaining class in K.

For EL, if the entities in the target EL Knowledge Base and the background knowledge K are aligned,[7] the soft-truth value of the ImpCl_E atoms can be deterministically obtained via such alignment: $\mathsf{ImpCl}_E(x,y)$ has soft-truth value 1 if y is one of the ontological classes of the entity z corresponding to x in the alignment, 0 otherwise.[8]

3.2 Annotation Coherence via Classes

The second part of the PSL model puts in relation the predicates ClAnn_N and ClAnn_E via ontological classes:

$$\begin{aligned} &w_1 : \mathsf{ClAnn}_N(m,t,c)\ \&\ \mathsf{ClAnn}_E(m,e,c) \rightarrow \mathsf{Ann}_{PSL}(m,t,e) \\ &w_2 : \mathsf{ClAnn}_N(m,t,c)\ \&\ \neg\mathsf{ClAnn}_E(m,e,c) \rightarrow \neg\mathsf{Ann}_{PSL}(m,t,e) \\ &w_3 : \neg\mathsf{ClAnn}_N(m,t,c)\ \&\ \mathsf{ClAnn}_E(m,e,c) \rightarrow \neg\mathsf{Ann}_{PSL}(m,t,e) \end{aligned} \tag{7}$$

where Ann_{PSL} is the predicate we use to estimate the coherence of a couple of NERC and EL candidate annotations on a given mention. The intuition here is that a NERC and an EL annotation implying the same classes[9] from the ontological background knowledge are likely to be coherent, and thus the soft-truth value of the corresponding Ann_{PSL} atom should be higher than when the annotations imply different classes. Note that these rules are not grounded. Rule weights w_1, w_2, w_3 are hyperparameters of our approach: the higher their values, the stronger the satisfaction of those rules — and hence coherence enforcement — is accounted for during inference.

Note that the two parts of the model have one important distinctive feature: for the actual construction of the model, the first part is *dynamic*, in the sense that the (partially-grounded) rules are instantiated based on the actual annotations and confidence scores available, while the second part is *static*, with rules involving only variables (and no constants) and thus defined once for all.

[7] This clearly includes the special case where the EL Knowledge Base is actually K.

[8] This assumes that K contains complete information about entity classes (closed-world assumption), which usually holds for the most general classes in the class taxonomy.

[9] Note that, for a given grounding of m, t and e, the value of Ann_{PSL} results from the contribution of several classes c.

$0.9 : \mathsf{Ann}_N(\mathrm{L}, \mathrm{ORG}) \;\&\; \mathsf{ImpCl}_N(\mathrm{ORG}, c) \rightarrow \mathsf{ClAnn}_N(\mathrm{L}, \mathrm{ORG}, c)$

$0.1 : \mathsf{Ann}_N(\mathrm{L}, \mathrm{PER}) \;\&\; \mathsf{ImpCl}_N(\mathrm{PER}, c) \rightarrow \mathsf{ClAnn}_N(\mathrm{L}, \mathrm{PER}, c)$

$0.5 : \mathsf{Ann}_E(\mathrm{L}, \text{A. Lincoln}) \;\&\; \mathsf{ImpCl}_E(\text{A. Lincoln}, c) \rightarrow \mathsf{ClAnn}_E(\mathrm{L}, \text{A. Lincoln}, c)$

$0.3 : \mathsf{Ann}_E(\mathrm{L}, \text{Lincoln MC}) \;\&\; \mathsf{ImpCl}_E(\text{Lincoln MC}, c) \rightarrow \mathsf{ClAnn}_E(\mathrm{L}, \text{Lincoln MC}, c)$

$0.2 : \mathsf{Ann}_E(\mathrm{L}, \text{Lincoln UK}) \;\&\; \mathsf{ImpCl}_E(\text{Lincoln UK}, c) \rightarrow \mathsf{ClAnn}_E(\mathrm{L}, \text{Lincoln UK}, c)$

$0.9 : \mathsf{Ann}_N(\mathrm{M}, \mathrm{LOC}) \;\&\; \mathsf{ImpCl}_N(\mathrm{LOC}, c) \rightarrow \mathsf{ClAnn}_N(\mathrm{M}, \mathrm{LOC}, c)$

$0.05 : \mathsf{Ann}_N(\mathrm{M}, \mathrm{PER}) \;\&\; \mathsf{ImpCl}_N(\mathrm{PER}, c) \rightarrow \mathsf{ClAnn}_N(\mathrm{M}, \mathrm{PER}, c)$

$0.05 : \mathsf{Ann}_N(\mathrm{M}, \mathrm{ORG}) \;\&\; \mathsf{ImpCl}_N(\mathrm{ORG}, c) \rightarrow \mathsf{ClAnn}_N(\mathrm{M}, \mathrm{ORG}, c)$

$0.9 : \mathsf{Ann}_E(\mathrm{M}, \text{Michigan}) \;\&\; \mathsf{ImpCl}_E(\text{Michigan}, c) \rightarrow \mathsf{ClAnn}_E(\mathrm{M}, \text{Michigan}, c)$

$0.1 : \mathsf{Ann}_E(\mathrm{M}, \text{U. of Michigan}) \;\&\; \mathsf{ImpCl}_E(\text{U. of Michigan}, c) \rightarrow \mathsf{ClAnn}_E(\mathrm{M}, \text{U. of Michigan}, c)$

$10 : \mathsf{ClAnn}_N(m, t, c) \;\&\; \mathsf{ClAnn}_E(m, e, c) \rightarrow \mathsf{Ann}_{PSL}(m, t, e)$

$10 : \mathsf{ClAnn}_N(m, t, c) \;\&\; \neg\mathsf{ClAnn}_E(m, e, c) \rightarrow \neg\mathsf{Ann}_{PSL}(m, t, e)$

$10 : \neg\mathsf{ClAnn}_N(m, t, c) \;\&\; \mathsf{ClAnn}_E(m, e, c) \rightarrow \neg\mathsf{Ann}_{PSL}(m, t, e)$

Fig. 1: Instantiation of the PSL model for the sentence "Lincoln is based in Michigan."

Figure 1 shows an example of instantiation of the model on the sentence "Lincoln is based in Michigan.", with two mentions $m_1 =$ Lincoln and $m_2 =$ Michigan (shortened for compactness to L and M, respectively), and assuming to have two NERC (ORG [0.9], PER [0.1]) and three EL (A. Lincoln [0.5], Lincoln MC [0.3], Lincoln UK [0.2]) confidence-weighted candidates on the first, and three NERC (LOC [0.9], PER [0.05], ORG [0.05]) and two EL (Michigan [0.9], U. of Michigan [0.1]) confidence-weighted candidates on the second.

The PSL model is further complemented with negative priors, i.e., additional rules stating that by default all open ground atoms (i.e., whose value has to be determined by the model) of investigated predicates (ClAnn_N, ClAnn_E, Ann_{PSL}) have 0 soft-truth value.

By running MPE inference on the model, we can compute the soft-truth value of all the ground atoms of Ann_{PSL}. Intuitively, the higher this value, the more likely a NERC annotation and an EL annotation are coherent on the given mention, with the combination of candidates scoring the highest value being the best NERC and EL annotation for the model, in light of their original confidence scores and the ontological knowledge.

By comparing the soft-truth value of the resulting Ann_{PSL} ground atoms with a threshold value θ (an hyperparameter of our approach), we can decide to which extent to rely on the prediction of the model, especially when revising (and possibly overruling) the best-choice candidate annotations proposed by some NERC and EL tools.

4 Evaluation

We conduct an evaluation, in a scenario where both NERC and EL analyses are run, to show that our PSL approach, leveraging some ontological background knowledge and applied "a posteriori" on the confidence-weighted candidate annotations returned

by a NERC tool and a EL tool, suggests better annotations than the highest score ones independently returned by the given tools. The data used by the PSL model (including the soft-truth values for ImpCl_N and ImpCl_E ground atoms), the evaluation package (excluding copyrighted dataset material), and additional result tables are available on the PSL4EA web-folder.[10]

4.1 Background Knowledge and Tools

As background knowledge we use YAGO [3]. We materialize, applying RDF_{pro} [14], all the inferable classes for an entity based on the YAGO TBox (e.g., subclass axioms), obtaining class information for 6,016,695 entities taken from a taxonomy of 568,255 classes.

To produce the NERC and EL annotations, we exploit two state-of-the-art tools:

- **Stanford NER** [4]: a reference tool for NERC. We use Stanford NER with the traditional CoNLL 2003 model consisting of 4 NERC types: Location (LOC), Person (PER), Organization (ORG), and Miscellaneous (MISC). By default, Stanford NER returns the best NERC labeling of a sentence, but it can be instructed to provide many alternative weighted NERC labelings of a sentence, from which it is possible to derive NERC candidates (and their confidences) for a mention;
- **DBpedia Spotlight** [5]: a reference tool for EL that uses DBpedia [13] as target knowledge base. Via its *candidates* service, DBpedia Spotlight can be instructed to return ten EL candidates (and their confidences) for a given mention.

4.2 Datasets

To verify the capability of our approach to generalize over different annotated data, we use three distinct datasets in our evaluation. They consist of textual documents together with gold-standard annotations, both for NERC and EL:[11]

- **AIDA CoNLL-YAGO** [6]: it consists of 1,393 English news articles from Reuters, hand-annotated with named entity types (PER, ORG, LOC, MISC) and YAGO2 entities (and Wikipedia page URLs). It is organized in three parts: eng.train (946 docs), eng.testa (216 docs), eng.testb (231 docs);
- **MEANTIME** [7]: it consists of 480 news articles from Wikinews, in four languages. In our evaluation, we only use all the 120 articles of the English section. The dataset includes manual annotations (limited to the first 5 sentences of the articles) for named entity types (only PER, ORG, LOC) and DBpedia entities;
- **TAC-KBP** [8]: it consists of 2,231 English documents (news article, newsgroup and blog posts, forum discussions). For each document, it is known that all the mentions of one or a few *query* entities can be linked to a certain Wikipedia page and to a specific NERC type (only PER, ORG, LOC), thus giving rise to a (partially) annotated gold standard for NERC and EL.

[10] http://pikes.fbk.eu/psl4ea.html

[11] We choose these datasets, among many available ones for NERC and for EL as they have both NERC and EL annotations that can be used to evaluate the improvement on both tasks.

4.3 Research Question and Evaluation Measures

We address the following research question:

> *Does the ontology-driven PSL4EA a posteriori joint revision of Stanford NER and DBpedia Spotlight annotations improve their NERC and EL performances?*

In investigating this research question, we remark that by construction the PSL model relies on the mentions detected by the NLP tools used, so the model may revise the NERC types and/or the EL entities proposed by the tools, but does not alter other aspects such as the mention span (i.e., the textual tokens that constitute the mention). As such, meaningful measures for our evaluation are the following ones, typically adopted in NERC and EL evaluation campaigns:

- type: a mention is counted as correct if it has the same span and NERC type as a gold annotation. It is the measure used in the CoNLL2003 NER evaluation, and corresponds to strong_typed_mention_match in the TAC-KBP official scorer;[12]
- link: a mention is counted as correct if it has the same span and EL entity as a gold annotation. It corresponds to strong_link_match in the TAC-KBP official scorer;
- type+link: an entity mention is counted as correct if it has the same span, NERC type, and EL entity as a gold annotation. It corresponds to strong_typed_link_match in the TAC-KBP official scorer.

For evaluating the performance on these measures, we use the standard metrics, namely precision (P), recall (R), and F_1, computed using the TAC-KBP official scorer on the predicted and gold standard annotations as follow:

- true positives (TP) = predicted annotations, in the gold standard;
- false positives (FP) = predicted annotations, not in the gold standard;
- false negatives (FN) = gold standard annotations, not predicted;
- $P = \frac{TP}{TP+FP}$, $R = \frac{TP}{TP+FN}$ and $F_1 = \frac{2 \cdot P \cdot R}{P+R}$.

4.4 Evaluation Procedure

We use AIDA eng.train as the gold standard G for determining ImpCl_N — Table 1 provides, for each NERC type, an overview of the YAGO classes of the top 10 soft-truth value ground atoms of ImpCl_N — while ImpCl_E is deterministically obtained directly via the DBpedia-YAGO alignment. We use AIDA eng.testa to optimize the PSL4EA model hyperparameters (cf. Section 3), namely $\bar{n}$ (=200),[13] w_1, w_2, w_3 (=10.0), and θ (=0.2). We adopt the quadratic loss function (cf. equation (3)).

All datasets are preprocessed in order to use entity URIs from the same version of DBpedia (namely, 2016-04) as the used DBpedia Spotlight version. In particular, the Wikipedia URLs in AIDA and TAC-KBP are aligned to the 2016-04 DBpedia URIs via DBpedia's 'Redirects', 'Revision URIs', and 'Wikipedia Links' datasets.

The experiment is conducted comparing the metric scores for the considered measures in two settings, without (*standard*) and with (*with PSL4EA*) the contribution of

[12] https://github.com/wikilinks/neleval (last accessed on April 1, 2018)

[13] With $\bar{n} = 200$, the background knowledge used in the model is reduced to 214 YAGO classes.

Table 1: Top 10 YAGO classes for each NERC type according to the soft-truth value (in parentheses) of ImpCl_N ground atoms learned from AIDA eng.train.

NERC type	YAGO Classes
PER	PhysicalEntity100001930 (.991), CausalAgent100007347 (.988), Object100002684 (.963), YagoLegalActorGeo (.963), Whole100003553 (.962), YagoLegalActor (.961), LivingThing100004258 (.960), Organism100004475 (.960), Person100007846 (.960), WikicatLivingPeople (.850)
ORG	YagoPermanentlyLocatedEntity (.945), Abstraction100002137 (.945), YagoLegalActorGeo (.938), YagoLegalActor (.925), Group100031264 (.924), SocialGroup107950920 (.923), Organization108008335 (.914), Association108049401 (.642), Club108227214 (.637), Unit108189659 (.340)
LOC	YagoPermanentlyLocatedEntity (.986), YagoLegalActorGeo (.967), PhysicalEntity100001930 (.909), Object100002684 (.907), YagoGeoEntity (.905), Location100027167 (.889), Region108630985 (.883), District108552138 (.866), AdministrativeDistrict108491826 (.865), Country108544813 (.524)
MISC	YagoPermanentlyLocatedEntity (.843), YagoLegalActorGeo (.679), PhysicalEntity100001930 (.614), Object100002684 (.609), YagoGeoEntity (.591), Location100027167 (.572), Region108630985 (.571), AdministrativeDistrict108491826 (.568), District108552138 (.568), Country108544813 (.549)

the PSL4EA model: in the *standard* setting we annotate the documents of the three corpora directly using the highest confidence score NERC type and EL entity proposed by Stanford NER and DBpedia spotlight; instead, in the *with PSL4EA* setting, the PSL4EA model picks, among all the confidence-weighted candidate annotations returned by the tools on the same mention, the ⟨NERC type, EL entity⟩ combination with the highest soft-truth value for Ann_{PSL}.[14]

We remark that our approach is not a complete NER+EL solution on its own but relies on annotations provided by NERC and EL tools (e.g., Stanford NER and DBpedia Spotlight as in the considered experiment), revised "a posteriori" using ontological knowledge. Therefore, in line with the investigated research question, we focus our study on comparing the scores between the two aforementioned settings, rather than analyzing the absolute scores obtained, which inherently depend also on the performances of the tools providing the candidate annotations (i.e., changing the tools would likely results in different overall P, R, and F_1 scores).

Furthermore, as some datasets are only partially annotated (e.g., TAC-KBP), in the paper we focus the evaluation only on the mentions detected by the tools (i.e., annotated with NERC and/or EL) — which we recall are the same in both settings — that are in the gold standard, in order to better compare performances across the different datasets, and to avoid obtaining scores, namely P and F_1, overly biased by FP in both settings. For completeness, scores considering all mentions returned by the tools as well as macro-averaged variants (by document, by NERC type) are provided on the web-folder.

4.5 Results and Discussion

Table 2 reports precision, recall, and F_1 (micro-averaged) for the evaluation measures on all the datasets, for both settings considered.

For all the metrics computed over the three datasets, the scores are consistently higher in the *with PSL4EA* setting than in the *standard* one, with improvements ranging from .004 to .032. Most of the improvements (24 out of 27) are statistically signifi-

[14] If the highest soft-truth value on a mention is below the threshold θ, the approach falls back to the best NERC and EL candidate annotations suggested by the tools on it.

Table 2: Precision, recall, and F_1 scores for type, link, and type+link measures for both settings on the three datasets (number of gold standard mentions in parentheses). Score differences (*with PSL4EA* − *standard*) are reported, with statistical significance ones marked in bold.

		type			link			type+link		
		P	R	F_1	P	R	F_1	P	R	F_1
AIDA (5616)	*standard*	.943	.875	.908	.662	.652	.656	.634	.625	.630
	with PSL4EA	.947	.879	.912	.670	.659	.665	.646	.635	.640
	Δ	**.004**	**.004**	**.004**	**.008**	**.007**	**.009**	**.012**	**.010**	**.010**
MEANTIME (792)	*standard*	.882	.695	.777	.703	.556	.621	.635	.502	.561
	with PSL4EA	.902	.711	.795	.714	.564	.630	.667	.527	.589
	Δ	**.020**	**.016**	**.018**	.011	.008	.009	**.032**	**.025**	**.028**
TAC-KBP (4969)	*standard*	.911	.652	.760	.401	.423	.412	.367	.386	.376
	with PSL4EA	.925	.662	.772	.408	.430	.419	.384	.404	.394
	Δ	**.014**	**.010**	**.012**	**.007**	**.007**	**.007**	**.017**	**.018**	**.018**

cant ($p < 0.05$) according the Approximate Randomization test. Similar outcomes (cf. PSL4EA web-folder for all the detailed data) are observed when:

- considering all mentions returned by the tools (rather than just those in the gold standard): improvements ranging from .003 to .025;
- macro-averaging by document: improvements ranging from .003 to .029;
- macro-averaging by NERC type: improvements ranging from .003 to .020.

Improvements for type+link (from .010 to .032), besides being all statistically significant, are always higher than the ones for the other two measures (type and link), thus confirming that the model is particularly effective in proposing, for a given mention, the correct ⟨NERC, EL⟩ annotation combination among the available candidates.

Analyzing more in detail the results, it is worth remarking that the model used for the evaluation, while trained only on AIDA eng.train, performs reasonably well also on the other two datasets, as confirmed by the substantially higher scores for the *with PSL4EA* setting over the *standard* one, with statistical significant improvements in most of the cases. This may suggest that the instantiated model generalizes well over different document collections, something we plan to further confirm with additional experiments in future work.

Summing up, the results on multiple datasets show that exploiting the PSL4EA model to "a posteriori" revise the annotations provided by Stanford NER and DBpedia Spotlight allows to consistently improve their NERC and EL scores, and thus we can positively answer our research question.

5 Discussion

Peculiarity of the PSL4EA model with respect to other PSL applications PSL has been applied for different structural relational learning tasks, including the distillation of a Knowledge Graph from candidate relation triples extracted from text [9]. In that

work, the authors encode the confidence score of extracted relation triples as the soft-truth value of the corresponding atoms, instead of rule weights like in PSL4EA. We experimented also with such configuration for the NERC and EL joint annotation revision setting, achieving however worse performances than modeling confidences as rule weights.

Applicability to other NERC and EL tools In the experiments discussed in Section 4, we applied PSL4EA to jointly revise the NERC and EL annotations produced by Stanford NER and DBpedia Spotlight. However, we remark that PSL4EA works on NERC and EL candidate *annotations*, and thus its applicability is not limited only to those specific tools. Indeed, the model used for the evaluation can be applied as-is to any couple of NERC and EL tools provided that: (i) the NERC tool annotates with the 4-type CoNLL2003 NERC categories (or its popular 3-type version omitting MISC); and, (ii) the EL tool annotates with DBpedia URIs. Clearly, the model can be adapted to other NERC categories and EL reference Knowledge Bases, revising ImpCl_N and ImpCl_E.

Implementation and Performances We implemented the PSL4EA approach used in the evaluation as a Java module[15] of PIKES [2], an open-source knowledge extraction framework exploiting several NLP analyses, including NERC (via Stanford NER) and EL (via DBpedia Spotlight). For the PSL inference, we use the open-source Java PSL software [12].[16] In details, the module (i) builds a PSL model and data dynamically for each named entity mention having both NERC and EL annotations, (ii) performs MPE inference, and (iii) saves the results in the PIKES output. Computationally, the performances of the module are roughly comparable to the annotation costs.[17]

Extension to other types of entity annotations In Section 3 we presented an ontology-driven PSL model for assessing the coherence and jointly revising NERC and EL annotations. That model can be extended to other typologies of annotations, that may involve (named) entities. Here we briefly discuss some ideas on how these additional annotations could contribute to the model, leaving the actual development of the model (and its evaluation) to future work.

Semantic Role Labeling (SRL) is the task of finding the semantic role of each argument of each (verbal or nominal) predicate in a sentence. For instance, in the sentence "Sergio Mattarella is the president of Italy", "president" evokes a *Leadership* frame (according to FrameNet [15]), and has two arguments, "Sergio Mattarella" (with role *Leader*) and "Italy" (with role *Governed*). Clearly, role annotations may contribute to further characterize entities, and, similarly to NERC and EL, they may imply some ontological classes. For instance, a *Leader* role annotation is more likely to occur on the mention of an entity of type "Leader109623038" in YAGO than an entity of type "Airplane102691156". We can thus think to include role annotations in PSL4EA with rules similar to the ones for NERC and EL:

$$w(M,A_i^R) : \mathsf{Ann}_R(M,A_i^R) \;\&\; \mathsf{ImpCl}_R(A_i^R,c) \rightarrow Cl\mathsf{Ann}_R(M,A_i^R,c) \tag{8}$$

[15] To be distributed with the next PIKES release.

[16] https://github.com/linqs/psl

[17] Note that substantial improvements of running time performances can be achieved with further engineering and optimization, out-of-scope for the purposes of this work.

where predicate ImpCl_R, capturing the ontological classes implied by role annotations, can be learned from data as described in Section 3.1.[18] However, to more precisely handle SRL annotations, the PSL model should be further extended to capture the fact that role annotations on different mentions (e.g., the *Leader* on "Sergio Mattarella" and the *Governed* on "Italy" in the example considered) but originating from the same predicate have to be related (i.e., selecting one candidate on one mention may affect the candidates on the others). Furthermore, the addition of the SRL annotations requires the extension of the rules ensuring the annotation coherence — cf. (7).

Another typology of annotation that may extend the PSL4EA model is entity coreference, i.e., the task of identifying that two or more mentions in a text refer to the same entity. Coreference should instruct the model to propagate the same annotations on all coreferring mentions, as suggested by the following rule for two coreferring mentions:

$$w_C(M_1,M_2) : \mathsf{Ann}_{PSL}(M_1,t,e) \;\&\; \mathsf{Coref}(M_1,M_2) \rightarrow \mathsf{Ann}_{PSL}(M_2,t,e) \tag{9}$$

where $\mathsf{Coref}(M_1,M_2)$ and $w_C(M_1,M_2)$ capture the coreference annotation and its confidence.

6 Related Work

We briefly overview some literature works related to our contribution.

PSL Application to Knowledge Extraction and NLP Probabilistic Soft Logic has been applied for some information extraction and NLP tasks. In [9] the authors apply PSL for Knowledge Graph Identification (KGI), that is the task of distilling a knowledge graph from the noisy output (subject-predicate-object triples) of information extractors (cf. also later in this section). The approach combines different strategies (e.g., entity classification, relational link prediction) together with constraints from existing ontologies. In [17] PSL is used to combine logical and distributional representations of natural-language meaning for the task of semantic textual similarity (STS). In [18] PSL is exploited to classify events mentioned in text leveraging event-event associations and fine-grained entity types. In [19] PSL is applied for the lexical inference problem, i.e., to guess unknown word meaning by leveraging linguistic and contextual features.

In our work PSL is applied to assess the coherence and revise entity annotations, exploiting ontological background knowledge. We are not aware of other works applying PSL to specifically improve NLP annotations.

NLP Annotation Improvement Some previous works have tackled the problem of improving the performances of some NLP tasks by leveraging or combining related analyses, focusing mainly on NERC and EL. In some works, one NLP analysis is used to influence the performance of another NLP task, in a pipeline, one-direction fashion. For instance, in [10,20] named entities are firstly recognized (NERC) and used to influence the entity disambiguation step (EL). Joint models for multiple tasks, in particular for NERC and EL, have also been developed, applying different techniques such

[18] A dataset to derive such information is presented in [16], where FrameNet frame elements (i.e., roles) are related to "compatible" WordNet synsets, which in turns can be directly mapped to YAGO classes.

as re-ranking mechanisms [21], conditional random field (CRF) extensions [22], semi-Markov structured linear classifiers [23], and probabilistic graphical models [11]. In [24], a joint model implemented as a structured CRF has been proposed, where NERC and EL analyses are complemented by coreference information.

Our work differs from all these approaches under several aspects. First, our approach is not a complete joint NERC and EL solution, but it works a posteriori on produced candidate annotations. This makes our approach applicable to many existing NERC and EL approaches as-is (i.e., without re-training their models or changing their implementations) granted they provide confidence-weighted candidate annotations. Second, it does not impose a directionality on the influence between the considered tasks, like in approaches such as [10,20]. Third, our approach stands out for the central role of the ontological background knowledge, exploited as "interlingua" to assess the coherence of the annotations from different NLP tasks. This is similar to the approach adopted by JPARK [25], where a pure probabilistic model — derived from some conditional independence assumptions, and leveraging class sets rather than individual class contributions like in PSL4EA — is used to revise entity annotations.

Knowledge Graph Construction Approaches for Knowledge Graph construction from text (e.g., Google's Knowledge Vault [26] and DeepDive [27]) have tackled the problem of determining the correctness of large sets of potentially noisy subject-predicate-object triples, obtained via information *extractors* from various types of content (e.g., documents, tables). Some of these works exploit ontological knowledge to constrain the selection of the extracted candidate triples. In NELL (Never-Ending Language Learning) [28], ontological constraints (e.g., a person cannot be a city) are used to filter the extracted triples. In other works, ontological knowledge is integrated directly in a probabilistic model, together with the confidence values of extractor candidates, such as in [29] (exploiting Markov Logic Networks) and the previously discussed PSL approach in [9]. Instead, a MAX-SAT algorithm is proposed in [30], to select high confidence triples that maximize the number of satisfied ontological constraints.

Our work differs from all these approaches and it is not directly comparable with them. To begin with, our approach works at the level of NLP annotations, rather than triples typically returned by relation extractors, and aims at improving the coherence of these annotations on a given mention, rather filtering extracted triples in order to be compliant with or to maximize the given set of ontological constraints. Furthermore, in all these approaches the relation extractors are aligned by construction with the relations and classes of the ontology used for constraining the triple selection, while in our work determining the ontological knowledge classes likely implied by the annotations is part of the problem and encoded into the PSL model.

7 Conclusions

In this paper we presented PSL4EA, an approach based on Probabilistic Soft Logic that, leveraging ontological background knowledge, aims at improving the joint annotation of entity mentions by NLP tools, for tasks such as NERC and EL. NLP annotations for different tasks are mapped to ontological classes of a common background

knowledge, then exploited to jointly assess the annotation coherence. Given confidence-weighted candidate annotations by multiple NLP tools for different tasks on the same textual entity mention, PSL4EA can be operationally applied to jointly revise the best annotation choices performed by the tools, in light of the coherence of the candidate annotations via the ontological knowledge.

We developed the approach for NERC and EL, leveraging YAGO as ontological background knowledge. We experimented with the model on the NERC and EL candidate annotations provided by two state-of-the-art tools, Stanford NER and DBpedia Spotlight, on three distinct reference datasets. The results show the capability of PSL4EA to jointly improve their annotations, as confirmed by the higher scores on all measures and metrics when applying the model.

As discussed in the paper, our future work mainly aims at concretely extending the proposed model to other NLP annotations than NERC and EL, starting with SRL and entity coreference. Furthermore, for the NERC and EL scenario, we plan to experiment with different training sets, possibly produced by combining different datasets, in order to further improve the generality and representativeness of the model obtained using the training part of the AIDA CoNLL-YAGO dataset.

Acknowledgments The author would like to thank Dr. Francesco Corcoglioniti for some useful suggestions and fruitful discussions while developing the idea.

References

1. Vossen, P., Agerri, R., Aldabe, I., Cybulska, A., van Erp, M., Fokkens, A., Laparra, E., Minard, A., Aprosio, A.P., Rigau, G., Rospocher, M., Segers, R.: Newsreader: Using knowledge resources in a cross-lingual reading machine to generate more knowledge from massive streams of news. Knowl.-Based Syst. **110** (2016) 60–85
2. Corcoglioniti, F., Rospocher, M., Aprosio, A.P.: Frame-based ontology population with PIKES. IEEE Trans. Knowl. Data Eng. **28**(12) (2016) 3261–3275
3. Hoffart, J., Suchanek, F.M., Berberich, K., Weikum, G.: YAGO2: A Spatially and Temporally Enhanced Knowledge Base from Wikipedia. Artif. Intell. **194** (2013) 28–61
4. Finkel, J.R., Grenager, T., Manning, C.: Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. In: Proceedings of ACL '05. (2005) 363–370
5. Daiber, J., Jakob, M., Hokamp, C., Mendes, P.N.: Improving Efficiency and Accuracy in Multilingual Entity Extraction. In: Proceedings of I-Semantics. (2013)
6. Hoffart, J., Yosef, M.A., Bordino, I., Fürstenau, H., Pinkal, M., Spaniol, M., Taneva, B., Thater, S., Weikum, G.: Robust Disambiguation of Named Entities in Text. In: Proceedings of EMNLP '11. (2011)
7. Minard, A.L., Speranza, M., Urizar, R., Altuna, B., van Erp, M., Schoen, A., van Son, C.: MEANTIME, the NewsReader Multilingual Event and Time Corpus. In: Proceedings of LREC 2016. (2016)
8. Ji, H., Grishman, R., Dang, H.: Overview of the TAC2011 Knowledge Base Population Track. In: TAC 2011 Proceedings Papers. (2011)
9. Pujara, J., Miao, H., Getoor, L., Cohen, W.: Knowledge Graph Identification. In: International Semantic Web Conference (ISWC). (2013)
10. Stern, R., Sagot, B., Béchet, F.: A Joint Named Entity Recognition and Entity Linking System. In: Proceedings of HYBRID '12. (2012) 52–60

11. Nguyen, D.B., Theobald, M., Weikum, G.: J-NERD: joint named entity recognition and disambiguation with rich linguistic features. TACL **4** (2016) 215–229
12. Bach, S.H., Broecheler, M., Huang, B., Getoor, L.: Hinge-loss Markov random fields and probabilistic soft logic. Journal of Machine Learning Research (JMLR) **18**(109) (2017) 1–67
13. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., van Kleef, P., Auer, S., Bizer, C.: Dbpedia - A large-scale, multilingual knowledge base extracted from wikipedia. Semantic Web **6**(2) (2015) 167–195
14. Corcoglioniti, F., Rospocher, M., Mostarda, M., Amadori, M.: Processing billions of rdf triples on a single machine using streaming and sorting. In: Proceedings of the 30th Annual ACM Symposium on Applied Computing. SAC '15, ACM (2015) 368–375
15. Baker, C.F., Fillmore, C.J., Lowe, J.B.: The Berkeley FrameNet Project. In: Proc. of ACL '98. (1998) 86–90
16. Tonelli, S., Bryl, V., Giuliano, C., Serafini, L.: Investigating the semantics of frame elements. In: EKAW. Volume 7603 of Lecture Notes in Computer Science., Springer (2012) 130–143
17. Beltagy, I., Erk, K., Mooney, R.J.: Probabilistic soft logic for semantic textual similarity. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL-14), Baltimore, MD (2014) 1210–1219
18. Liu, S., Liu, K., He, S., Zhao, J.: A probabilistic soft logic based approach to exploiting latent and global information in event classification. In: AAAI, AAAI Press (2016) 2993–2999
19. Wang, W.C., Ku, L.W.: Identifying chinese lexical inference using probabilistic soft logic. In: 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). (Aug 2016) 737–743
20. Plu, J., Rizzo, G., Troncy, R.: A Hybrid Approach for Entity Recognition and Linking. In: Semantic Web Evaluation Challenges ESWC 2015 - Revised Selected Papers. Volume 548. (2015) 28–39
21. Sil, A., Yates, A.: Re-ranking for joint named-entity recognition and linking. In: Proceedings of CIKM '13. (2013) 2369–2374
22. Luo, G., Huang, X., Lin, C.Y., Nie, Z.: Joint Named Entity Recognition and Disambiguation. In: Proceedings of EMNLP '15. (2015) 879–888
23. Leaman, R., Lu, Z.: TaggerOne: joint named entity recognition and normalization with semi-Markov Models. Bioinformatics **32**(18) (2016) 2839–2846
24. Durrett, G., Klein, D.: A Joint Model for Entity Analysis: Coreference, Typing, and Linking. TACL **2** (2014) 477–490
25. Rospocher, M., Corcoglioniti, F.: Joint Posterior Revision of NLP Annotations via Ontological Knowledge. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence and the 23rd European Conference on Artificial Intelligence, IJCAI-ECAI 2018, Stockholm, Sweden, July 13-19, 2018, ijcai.org (2018)
26. Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., Strohmann, T., Sun, S., Zhang, W.: Knowledge Vault: A Web-scale Approach to Probabilistic Knowledge Fusion. In: Proceedings of ACM KDD '14. (2014) 601–610
27. De Sa, C., Ratner, A., Ré, C., Shin, J., Wang, F., Wu, S., Zhang, C.: DeepDive: Declarative Knowledge Base Construction. SIGMOD Rec. **45**(1) (2016) 60–67
28. Mitchell, T., Cohen, W., Hruschka, E., Talukdar, P., Betteridge, J., Carlson, A., Dalvi, B., Gardner, M., Kisiel, B., Krishnamurthy, J., Lao, N., Mazaitis, K., Mohamed, T., Nakashole, N., Platanios, E., Ritter, A., Samadi, M., Settles, B., Wang, R., Wijaya, D., Gupta, A., Chen, X., Saparov, A., Greaves, M., Welling, J.: Never-Ending Learning. In: Proceedings of AAAI-15. (2015)
29. Jiang, S., Lowd, D., Dou, D.: Learning to Refine an Automatically Extracted Knowledge Base Using Markov Logic. In: Proc. of ICDM '12. (2012) 912–917
30. Suchanek, F.M., Sozio, M., Weikum, G.: SOFIE: A Self-Organizing Framework for Information Extraction. In: WWW 2009. (2009)