N-ary Relation Extraction for Simultaneous T-Box and A-Box Knowledge Base Augmentation

Marco Fossati [a,*], Emilio Dorigatti [b], and Claudio Giuliano [c]

[a] *Data and Knowledge Management Unit, Fondazione Bruno Kessler, via Sommarive 18, 38123 Trento, Italy*
E-mail: fossati@fbk.eu

[b] *Department of Computer Science, University of Trento, via Sommarive 9, 38123 Trento, Italy*
E-mail: emilio.dorigatti@studenti.unitn.it

[c] *Future Media Unit, Fondazione Bruno Kessler, via Sommarive 18, 38123 Trento, Italy*
E-mail: giuliano@fbk.eu

Abstract. The Web has evolved into a huge mine of knowledge carved in different forms, the predominant one still being the free-text document. This motivates the need for *Intelligent Web-reading Agents*: hypothetically, they would skim through disparate Web sources corpora and generate meaningful structured assertions to fuel Knowledge Bases (KBs). Ultimately, comprehensive KBs, like WIKIDATA and DBPEDIA, play a fundamental role to cope with the issue of information overload. On account of such vision, this paper depicts the FACT EXTRACTOR, a complete Natural Language Processing (NLP) pipeline which reads an input textual corpus and produces machine-readable statements. Each statement is supplied with a confidence score and undergoes a disambiguation step via Entity Linking, thus allowing the assignment of KB-compliant URIs. The system implements four research contributions: it (1) executes N-ary relation extraction by applying the Frame Semantics linguistic theory, as opposed to binary techniques; it (2) simultaneously populates both the T-Box and the A-Box of the target KB; it (3) relies on a single NLP layer, namely part-of-speech tagging; it (4) enables a completely supervised yet reasonably priced machine learning environment through a crowdsourcing strategy. We assess our approach by setting the target KB to DBpedia and by considering a use case of $52,000$ Italian Wikipedia soccer player articles. Out of those, we yield a dataset of more than $213,000$ triples with an estimated 81.27% F_1. We corroborate the evaluation via (i) a performance comparison with a baseline system, as well as (ii) an analysis of the T-Box and A-Box augmentation capabilities. The outcomes are incorporated into the Italian DBpedia chapter, can be queried through its SPARQL endpoint, and/or downloaded as standalone data dumps. The codebase is released as free software and is publicly available in the DBpedia Association repository.

Keywords: Information Extraction, Natural Language Processing, Frame Semantics, Crowdsourcing, Machine Learning

1. Introduction

The World Wide Web is nowadays one of the most prominent sources of information and knowledge. Despite the constantly increasing availability of semi-structured or structured data, a major portion of its content is still represented in an unstructured form, namely free text: understanding its meaning is a complex task for machines and yet relies on subjective human interpretations. Hence, there is an ever growing need for *Intelligent Web-reading Agents*, i.e., Artificial Intelligence systems that can read and comprehend human language in documents across the Web. Ideally, these agents should be robust enough to interchange between heterogeneous sources with agility, while maintaining equivalent reading capabilities. More specifically, given a set of input corpora (where an item corresponds to the textual content of a Web source), they should be able to

[*] Corresponding author. E-mail: fossati@fbk.eu

navigate from corpus to corpus and to extract comparable structured assertions out of each one. Ultimately, the collected data would feed a target *Knowledge Base* (KB), namely a repository that encodes areas of human intelligence into a richly shaped representation. Typically, KBs are made of graphs, where real-world and abstract entities are bound together through relationships, and classified according to a formal description of the world, i.e., an ontology. The terminological component (*T-Box*) and the assertional component (*A-Box*) represent the core parts of an ontology: the former accounts for the conceptual schema, bearing definitions of classes, e.g., `a soccer player is an athlete`, and properties, e.g., `a soccer player is member of a soccer club`; the latter provides assertions about entities that conform to the T-Box, e.g., `Roberto Baggio is a soccer player`, and `Roberto Baggio is member of the Italy national soccer team`.

In this scenario, the encyclopedia Wikipedia contains a huge amount of data, which may represent the best digital approximation of human knowledge. Recent efforts, most notably DBpedia [37], Freebase [11], YAGO [32], and Wikidata [56], attempt to extract semi-structured data from Wikipedia in order to build KBs that are proven useful for a variety of applications, such as question answering, entity summarization and Entity Linking (EL), just to name a few. The idea has not only attracted a continuously rising commitment of research communities, but has also become a substantial focus of the largest Web companies. As an anecdotal yet remarkable proof, Google acquired Freebase in 2010,[1] embedded it in its Knowledge Graph,[2] and has lately opted to shut it down to the public.[3] Currently, it is foreseen that Freebase data will eventually migrate to Wikidata[4] via the *primary sources* tool,[5] which aims at standardizing the flow for data donations.

However, the trustworthiness of a general-purpose KB like Wikidata is an essential requirement to ensure reliable (thus high-quality) content: as a support for their plausibility, data should be validated against third-party resources. Even though the Wikidata community strongly agrees on the concern,[6] few efforts have been approached towards this direction. The addition of references to external (i.e., non-Wikimedia), authoritative Web sources can be viewed as a form of validation. Consequently, such real-world setting further consolidates the need for an intelligent agent that harvests structured data from raw text and produces, e.g., Wikidata statements with reference URLs. Besides the prospective impact on the KB augmentation and quality, the agent would also dramatically shift the burden of manual data addition and curation, by pushing the (intended) fully human-driven flow towards an assisted paradigm, where automatic suggestions of pre-packaged statements just require to be approved or rejected. Figure 1 depicts the current state of the primary sources tool interface for Wikidata editors, which is in active development yet illustrates such future technological directions. Our system already takes part in the process, as it feeds the tool back-end.

On the other hand, the DBpedia Extraction Framework[7] is pretty much mature when dealing with Wikipedia semi-structured content like infoboxes, links and categories. Nevertheless, unstructured content (typically text) plays the most crucial role, due to the potential amount of extra knowledge it can deliver: to the best of our understanding, no efforts have been carried out to integrate an unstructured data extractor into the framework. For instance, given the Germany football team article,[8] we aim at extracting a set of meaningful facts and structure them in machine-readable statements. The sentence `In Euro 1992, Germany reached the final, but lost 0–2 to Denmark` would produce a list of *triples*, such as:

```
(Germany, defeat, Defeat_01)
(Defeat_01, winner, Denmark)
(Defeat_01, loser, Germany)
(Defeat_01, score, 0–2)
(Defeat_01, competition, Euro 1992)
```

To fulfill both Wikidata and DBpedia duties, we aim at investigating in what extent can the *Frame Semantics* theory [23,24] be leveraged to perform Information Extraction over Web documents. The main purpose

[1] https://googleblog.blogspot.it/2010/07/deeper-understanding-with-metaweb.html

[2] https://www.google.com/intl/en_us/insidesearch/features/search/knowledge.html

[3] https://plus.google.com/109936836907132434202/posts/bu3z2wVqcQc

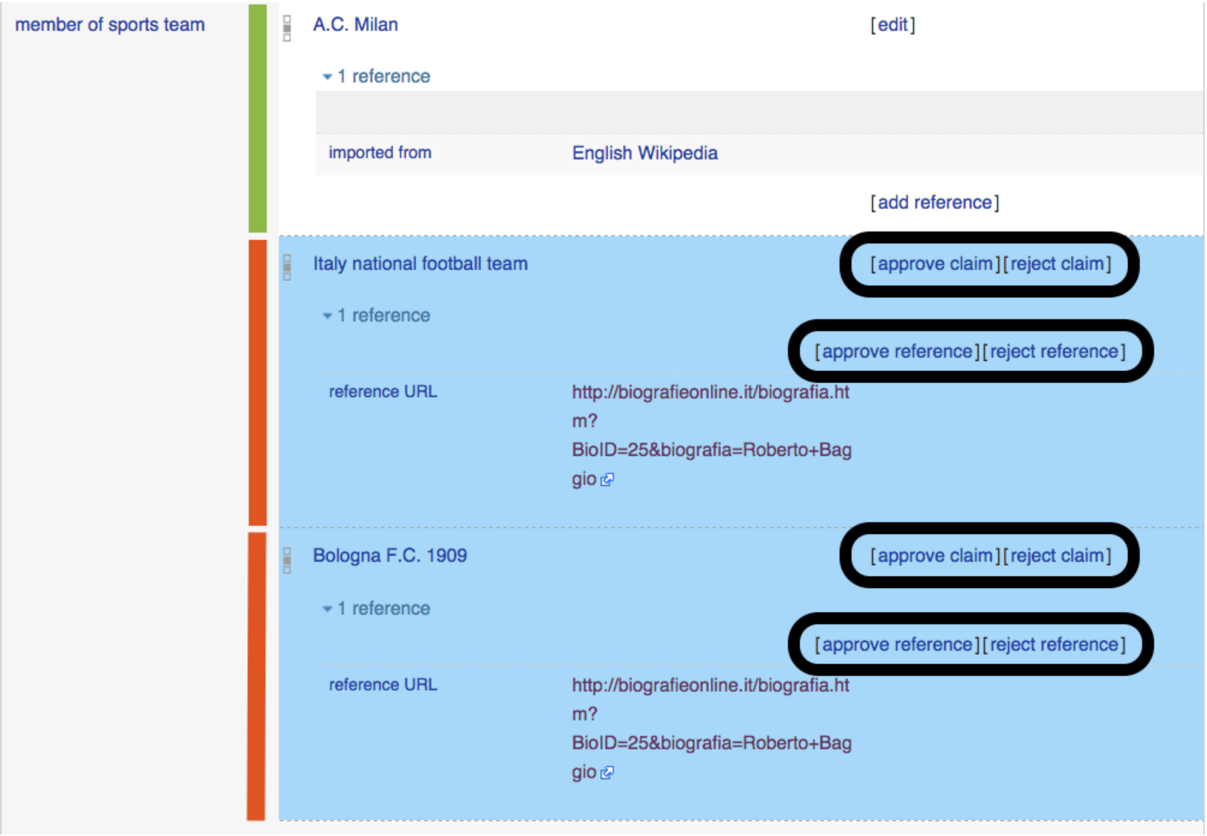
[4] https://www.wikidata.org/wiki/Wikidata:WikiProject_Freebase

[5] https://www.wikidata.org/wiki/Wikidata:Primary_sources_tool

[6] https://www.wikidata.org/wiki/Wikidata:Referencing_improvements_input, http://blog.wikimedia.de/2015/01/03/scaling-wikidata-success-means-making-the-pie-bigger/

[7] https://github.com/dbpedia/extraction-framework

[8] http://en.wikipedia.org/wiki/Germany_national_football_team



member of sports team

A.C. Milan

imported from: English Wikipedia

Italy national football team

reference URL: http://biografieonline.it/biografia.htm?BioID=25&biografia=Roberto+Baggio

Bologna F.C. 1909

reference URL: http://biografieonline.it/biografia.htm?BioID=25&biografia=Roberto+Baggio

[edit]
1 reference
[add reference]
[approve claim][reject claim]
1 reference
[approve reference][reject reference]
[approve claim][reject claim]
1 reference
[approve reference][reject reference]

Figure 1. Screenshot of the Wikidata primary sources gadget activated in ROBERTO BAGGIO's page. The statement highlighted with a green vertical line already exists in the KB. Automatic suggestions are displayed with a blue background: these statements require validation and are highlighted with a red vertical line. They can be either approved or rejected by editors, via the buttons highlighted with black circles.

of Information Extraction is to gather structured data from free text via Natural Language Processing (NLP), while Frame Semantics originates from linguistic research in Artificial Intelligence. A *frame* can be informally defined as an event triggered by some term in a text and embedding a set of participants, or *Frame Elements* (FEs). Hence, the aforementioned sentence would induce the DEFEAT frame (triggered by `lost`) together with the WINNER, COMPETITION, and SCORE participants. In seminal work [26], frames have already been proposed as atomic units of meaning. Furthermore, the theory has led to the creation of FRAMENET [6,7], namely a lexical database with manually annotated examples of frame usage in English. FrameNet currently adheres to a rigorous protocol for data annotation and quality control. The activity is known to be expensive with respect to time and cost, thus constituting an encumbrance for the extension of the resource [5], both in terms of additional labeled sentences and of languages.

To alleviate this, crowdsourcing the annotation task is proven to dramatically reduce the financial and temporal expenses. Consequently, we foresee to exploit the novel annotation approach described in [25], which provides full frame annotation in a *single* step and in a bottom-up fashion (i.e., *from FEs up to frames*), thus being also more compliant with the definitions as per [24]. While we acknowledge that crowdsourcing still entails a manual effort, it is worth to highlight that the whole process can be automated by programmatically interacting with a crowdsourcing platform API. Therefore, we may consider this duty not to require any direct manual intervention, other than the creation of a small amount of test annotations, acting as a protection mechanism against cheating.

1.1. Contributions

In this paper, we focus on Wikipedia as the source corpus and on DBpedia as the target KB. We propose to

Table 1

Fact extraction examples on the Germany national football team article

Sentence	Extracted statements
The first manager of the Germany national team was Otto Nerz	(Germany, roster, Roster_01), (Roster_01, team manager, Otto Nerz)
Germany has won the World Cup four times	(Germany, trophy, Trophy_01), (Trophy_01, competition, World Cup), (Trophy_01, count, 4)
In the 70s, Germany wore Erima kits	(Germany, wearing, Wearing_01), (Wearing_01, garment, Erima), (Wearing_01, period, 1970)

apply NLP techniques to Wikipedia text in order to harvest structured facts that can be used to automatically add novel statements to DBpedia. Our FACT EXTRACTOR is set apart from related state of the art thanks to the combination of the following contributions:

1. **N-ary relation extraction**, as opposed to binary standard approaches, e.g., [22,4,3,55,21,12], and in line with the notion of knowledge pattern [26];
2. **simultaneous T-Box and A-Box population** of the target KB, in contrast to, e.g., [19];
3. **shallow NLP machinery**, only requiring the grammatical analysis (i.e., part-of-speech tagging) layer, with no need for syntactic parsing (e.g., [39]) nor semantic role labeling (e.g., [35,34,36,15,10]);
4. **low-cost yet supervised machine learning** paradigm, via training set crowdsourcing, which ensures full supervision without the need for expert annotators.

1.2. Problem and Solution

The main research challenge is formulated as a KB population problem: specifically, we tackle how to automatically enrich DBpedia resources with novel statements extracted from the text of Wikipedia articles. We conceive the solution as a machine learning task implementing the Frame Semantics linguistic theory [23,24]: we investigate how to recognize meaningful factual parts given a natural language sentence as input. We cast this as a classification activity falling into the supervised learning paradigm. In particular, we focus on the construction of a new extractor, to be integrated into the current DBpedia infrastructure. Frame Semantics will enable the discovery of relations that hold between entities in raw text. Its implementation takes as input a collection of documents from Wikipedia (i.e., the corpus) and outputs a structured dataset composed of machine-readable statements.

The remainder of this paper is structured as follows. We introduce a use case in Section 2, which will drive the implementation of our system. Its high-level architecture is then described in Section 3, and devises the core modules, which we detail in Section 4, 5, 6, 7, and 8. A baseline system is reported in Section 9: this enables the comparative evaluation presented in Section 10, among with an assessment of the T-Box and A-Box enrichment capabilities. In Section 11, we gather a list of research and technical considerations to pave the way for future work. The state of the art is reviewed in Section 12, before our conclusions are drawn in Section 13.

2. Use Case

Soccer is a widely attested domain in Wikipedia: according to the ITALIAN DBPEDIA,[9] the Italian Wikipedia counts a total of $59,517$ articles describing soccer-related entities, namely 2.63% of the whole chapter. Moreover, infoboxes on those articles are generally very rich (cf. for instance the Germany national football team article). On account of these observations, the soccer domain properly fits the main challenge of this effort. Table 1 displays three examples of candidate statements from the Germany national football team article text, which do not exist in the corresponding DBpedia resource. In order to facilitate the readability, the examples stem from the English chapter, but also apply to Italian.[10]

3. System Description

The implementation workflow is intended as follows, depicted in Figure 2, and applied to the use case in Italian language:

[9] As per the 2015 release, based on the Wikipedia dumps from January 2015.

[10] https://it.wikipedia.org/wiki/Nazionale_di_calcio_della_Germania

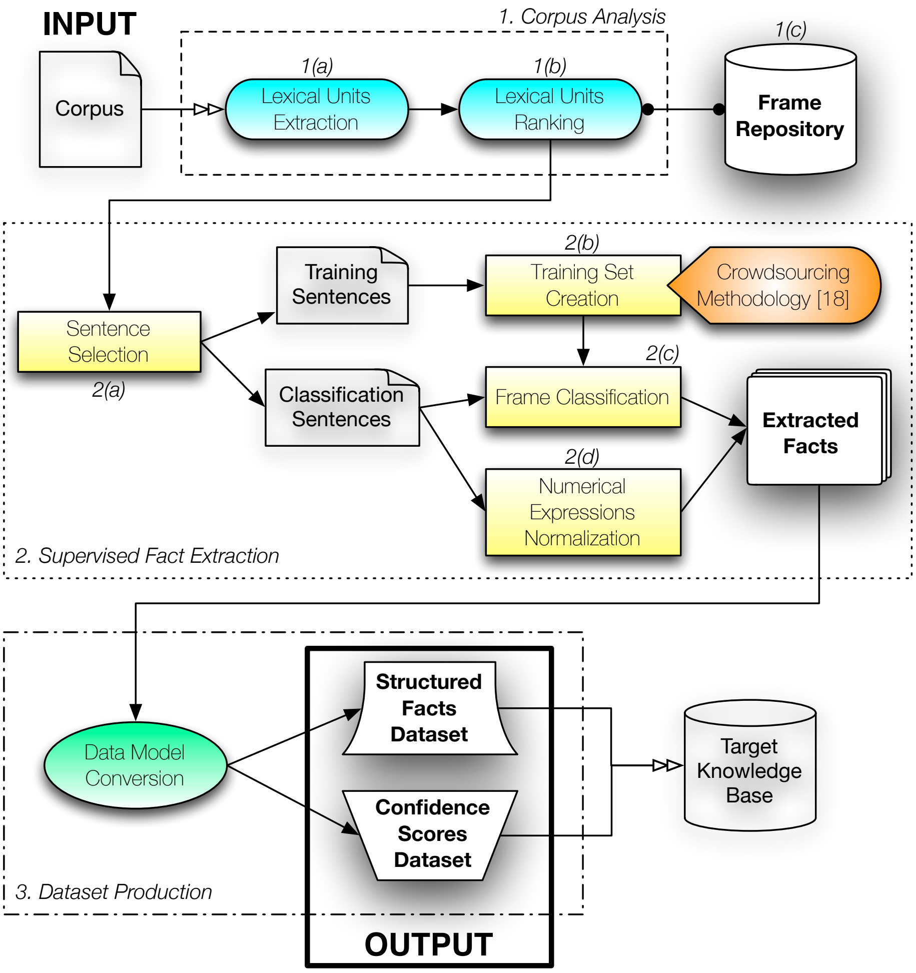


Figure 2. High level overview of the *Fact Extractor* system

1. *Corpus Analysis*
   (a) **Lexical Units (LUs) Extraction** via text tokenization, lemmatization, and part-of-speech (POS) tagging. LUs serve as the frame triggers;
   (b) **LUs Ranking** through lexicographical and statistical analysis of the input corpus. The selection of top-N meaningful LUs is produced via a combination of term weighting measures (i.e., TF-IDF) and purely statistical ones (i.e., standard deviation);
   (c) each selected LU will trigger one or more frames together with their FEs, depending on the definitions contained in a given **frame repository**. The repository also holds the input labels for two automatic classifiers (the former handling FEs, the latter frames) based on Support Vector Machines (SVM).
2. *Supervised Fact Extraction*
   (a) **Sentence Selection**: two sets of sentences are gathered upon the candidate LUs, one for training examples and the other for the actual classification;
   (b) **Training Set Creation**: construction of a fully annotated training set via crowdsourcing;
   (c) **Frame Classification**: massive frame and FEs extraction on the input corpus seed sentences, via the classifiers trained with the result of the previous step.
3. *Dataset Production*: structuring the extraction results to fit the target KB (i.e., DBpedia) **data model** (i.e., RDF). A frame would map to a property, while participants would either map to subjects or to objects, depending on their role.

We proceed with a simplification of the original Frame Semantics theory with respect to two aspects: (a) LUs may be evoked by additional POS (e.g., nouns), but we focus on verbs, since we assume that they are more likely to trigger factual information; (b) depending on the frame repository, full lexical coverage may not be guaranteed (i.e., some LUs may not trigger any frames), but we expect that ours will, otherwise LU candidates would not generate any fact.

4. Corpus Analysis

Since Wikipedia also contains semi-structured data, such as formatting templates, tables, references, images, etc., a pre-processing step is required to obtain the raw text representation only. To achieve this, we leverage a third-party tool, namely the WIKIEXTRACTOR.[11] From the entire Italian Wikipedia corpus, we slice the use case subset by querying the Italian DBpedia chapter[12] for the Wikipedia article IDs of relevant entities.

4.1. Lexical Units Extraction

Given the use case corpus, we first extract the complete set of verbs through a standard NLP pipeline: tokenization, lemmatization and POS tagging. POS information is required to identify verbs, while lemmas are needed to build the ranking. TREETAGGER[13] is exploited to fulfill these tasks.

4.2. Lexical Units Selection

The unordered set of extracted verbs needs to undergo a further analysis, which aims at discovering the most representative verbs with respect to the corpus. As a matter of fact, lexicon (LUs) in text is typically distributed according to the Zipf's law,[14] where few highly occurring terms cater for a vast portion of the corpus. Of course, grammatical words (stopwords) are the top-occurring ones, although they do not bear any meaning, and must be filtered. We can then focus on the most frequent LUs and benefit from two advantages: first, we ensure a wide coverage of the corpus with few terms; second, we minimize the annotation cost. To achieve this, we need to frame the selection as a ranking problem, where we catch a frequency signal in order to calculate a score for each LU. It is clear that processing the long tail of lowly occurring LUs will be very expensive and not particularly fruitful.

Two measures are leveraged to generate a score for each verb lemma. We first compute the term frequency–inverse document frequency (TF-IDF) of each verb lexicalization t belonging to the set of occurring tokens T over each document d in the corpus C: this weighting measure $\alpha_{t,d}$ is intended to capture the *lexicographical* relevance of a given verb, namely how important it is with respect to other terms in the whole corpus. Then, we determine the standard deviation value out of the TF-IDF scores set A_t: this *statistical* measure β_t is meant to catch heterogeneously distributed verbs, in the sense that the higher the standard deviation is, the more variably the verb is used, thus helping to understand its overall usage signal over the corpus. Ultimately, we produce the final score s and assign it to a verb lemma by averaging all its lexicalizations scores B. To clarify how the two measures are combined, we formalize the LU selection problem as follows.

$$\forall t \in T, \forall d \in C \text{ let } \alpha_{t,d} = tfidf(t,d);$$

$$A_t = \bigcup_{d \in C} \{\alpha_{t,d}\}; \quad \beta_t = stdev(A_t);$$

$$B = \bigcup_{t \in T} \{\beta_t\}; \quad s = avg(B)$$

The ranking is publicly available in the code repository.[15] The top-N lemmas serve as candidate LUs, each evoking one or more frames according to the definitions of a given frame repository.

5. Use Case Frame Repository

Among the top 50 LUs that emerged from the corpus analysis phase, we manually selected a subset of 5 items to facilitate the full implementation of our pipeline. Once the approach has been tested and evaluated, it can scale up to the whole ranking (cf. Section 11 for more observations). The selected LUs comply with two criteria: first, they are picked from both the best and the worst ranked ones, with the purpose of assessing the validity of the corpus analysis as a whole; second, they fit the use case domain, instead of being generic. Con-

[11] https://github.com/attardi/wikiextractor

[12] http://it.dbpedia.org/sparql

[13] http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

[14] https://en.wikipedia.org/wiki/Zipf%27s_law

[15] https://github.com/dbpedia/fact-extractor/blob/master/resources/stdevs-by-lemma.json

sequently, we proceed with the following LUs: esordire (to start out), giocare (to play), perdere (to lose), rimanere (to stay, remain), and vincere (to win).

The next step consists of finding a language resource (i.e., frame repository) to suitably represent the use case domain. Given a resource, we first need to define a relevant subset, then verify that both its frame and FEs definitions are a relevant fit. After an investigation of FrameNet and KICKTIONARY [53], we notice that:

- to the best of our knowledge, no suitable domain-specific Italian FrameNet or Kicktionary are publicly available, in the sense that neither LU sets nor annotated sentences for the Italian language match our purposes;
- FrameNet is too coarse-grained to encode our domain knowledge. For instance, the FINISH_COMPETITION frame may seem a relevant candidate at a first glimpse, but does not make the distinction between a victory and a defeat (as it can be triggered by both to win and to lose LUs), thus rather fitting as a super-frame (but no sub-frames exist);
- Kicktionary is too specific, since it is built to model the speech transcriptions of football matches. While it indeed contains some in-scope frames such as VICTORY (evoked by to win), most LUs are linked to frames that are not likely to appear in our input corpus, e.g., to play with PASS (occurring in sentences like Ronaldinho played the ball in for Deco).

Therefore, we adopted a custom frame repository, maximizing the reuse of the available ones as much as possible, thus serving as a hybrid between FrameNet and Kicktionary. Moreover, we tried to provide a challenging model for the classification task, prioritizing FEs overlap among frames and LU ambiguity (i.e., focusing on very fine-grained semantics with subtle sense differences). We believe this does not only apply to machines, but also to humans: we can view it as a stress test both for the machine learning and the crowdsourcing parts. A total of 6 frames and 15 FEs are modeled with Italian labels as follows:

- ATTIVITÀ (activity), FEs AGENTE (agent), COMPETIZIONE (competition), DURATA (duration), LUOGO (place), SQUADRA (team), TEMPO (time). Evoked by esordire (to start out), giocare (to play), rimanere (to stay, remain), as in Roberto Baggio played with Juventus in Serie A between 1990 and 1995. Frame label translated from FrameNet ACTIVITY, FEs from a subset of FrameNet ACTIVITY;
- PARTITA (match), FEs SQUADRA_1 (team 1), SQUADRA_2 (team 2), COMPETIZIONE, LUOGO, TEMPO, PUNTEGGIO (score), CLASSIFICA (ranking). Evoked by giocare, vincere (to win), perdere (to lose), as in Juventus played Milan at the UEFA cup final (2-0). Frame label translated from Kicktionary MATCH, FEs from a subset of FrameNet COMPETITION, LU shared by both;
- SCONFITTA (defeat), FEs PERDENTE, VINCITORE, COMPETIZIONE, LUOGO, TEMPO, PUNTEGGIO, CLASSIFICA. Sub-frame of PARTITA, evoked by perdere, as in Milan lost 0-2 against Juventus at the UEFA cup final. Frame label translated from Kicktionary DEFEAT, FEs from a subset of FrameNet BEAT_OPPONENT, LU from Kicktionary;
- STATO (status), FEs ENTITÀ (entity), STATO (status), DURATA, LUOGO, SQUADRA, TEMPO. Evoked by rimanere, as in Roberto Baggio remained faithful to Juventus until 1995. Custom frame and FEs derived from corpus evidence, to augment the rimanere LU ambiguity;
- TROFEO (trophy), FEs AGENTE, COMPETIZIONE, SQUADRA, PREMIO (prize), LUOGO, TEMPO, PUNTEGGIO, CLASSIFICA. Sub-frame of PARTITA, evoked by vincere, as in Roberto Baggio won a UEFA cup with Juventus in 1992. Custom frame label, FEs from a subset of FrameNet WIN_PRIZE, LU from FrameNet;
- VITTORIA (victory), FEs VINCITORE, PERDENTE, COMPETIZIONE, LUOGO, TEMPO, PUNTEGGIO, CLASSIFICA. Evoked by vincere, as in Juventus won 2-0 against Milan at the UEFA cup final. Frame label translated from Kicktionary VICTORY, FEs from a subset of FrameNet BEAT_OPPONENT, LU from Kicktionary.

6. Supervised Fact Extraction

The first stage involves the creation of the training set: we leverage the crowdsourcing platform CROWDFLOWER[16] and a one-step frame annotation method, which we briefly illustrate in Section 6.2. The training set has a double outcome, as it will feed two classifiers:

[16] http://www.crowdflower.com/

one will identify FEs, and the other is responsible for frames.

Both frame and FEs recognition are cast to a multiclass classification task: while the former can be related to text categorization, the latter should answer questions such as *"can this entity be this FE?"* or *"is this entity this FE in this context?"*. Such activity boils down to semantic role labeling (cf. [38] for an introduction), and usually requires a more fine-grained text analysis. Previous work in the area exploits deeper NLP layers, such as syntactic parsing (e.g., [39]). We alleviate this through Entity Linking (EL) techniques, which perform word sense disambiguation by linking relevant parts of a source sentence to URIs of a target KB. We leverage THE WIKI MACHINE,[17] a state-of-the-art [40] approach based on [28] and conceived for connecting text to Wikipedia URLs, thus inherently entailing DBpedia URIs. EL results are part of the FE classifier feature set. We claim that EL enables the automatic addition of features based on existing entity attributes within the target KB (notably, the class of an entity, which represents its semantic type).

Given as input an unknown sentence, the full frame classification workflow involves the following tasks: tokenization, POS tagging, EL, FE classification, and frame classification.

6.1. Sentence Selection

The sentence selection procedure allows to harvest meaningful sentences from the input corpus, and to feed the classifier. Therefore, its outcome is two-fold: to build a representative training set and to extract relevant sentences for classification. We experimented multiple strategies as follows. They all share the same base constraint, i.e., each seed must contain a LU lexicalization.

- *Baseline*: the seed must be comprised in a given interval of length in words;
- *Sentence splitter*: the seed forms a complete sentence extracted with a sentence splitter. This strategy requires training data for the splitter;
- *Chunker grammar*: the seed must match a pattern expressed via a context-free chunker grammar. This strategy requires a POS tagger and engineering effort for defining the grammar (e.g., a noun phrase, followed by a verb phrase, followed by a noun phrase);
- *Syntactic*: the seed is extracted from a parse tree obtained through immediate constituent analysis, the idea being to split long and complex sentences into shorter ones. This strategy requires a suitable grammar and a parser;
- *Lexical*: the seed must match a pattern based on lexicalizations of candidate entities. This strategy requires querying a KB for instances of relevant classes (e.g., soccer-related ones as per the use case).

First, we note that all the strategies but the baseline necessitate an evident cost overhead in terms of language resources availability and engineering. Furthermore, given the soccer use case input corpus of $52,000$ articles circa, all strategies but the syntactic one dramatically reduce the number of seeds, while the baseline performed an extraction with a $.95$ article/seed ratio (despite some noise). Compared to the sentence splitter strategy, the syntactic one brought an increase of roughly 4x in the number of seeds, at a cost of 375x in processing time, which we deemed not worth. These numbers arise from an experiment carried out for Wikidata, with a larger corpus composed of $500,000$ documents circa from heterogeneous Web sources (cf. Section 11.3).

Consequently, we decided to leverage the baseline for the sake of simplicity and for the compliance to our contribution claims. We set the interval to $5 < w < 25$, where w is the number of words. The selection of relatively concise sentences is motivated by empirical and conceptual reasons:

(a) it is known that crowdsourced NLP tasks should be as simple as possible [54]. Hence, it is vital to maximize the accessibility, otherwise the job would be too confusing and frustrating, with a consistent impact in quality and execution time;
(b) frame annotation is a particularly complex task [5], even for expert linguists. Therefore, the inter-annotator agreement is expected to be fairly low. Compact sentences minimize disagreement, as corroborated by the average score we obtained in the gold standard (cf. Section 10.1, Table 3 and 4);
(c) since we aim at populating a KB, we prioritize precise statements instead of recall, for the sake of data quality. As a result, we focus on atomic factual information to reduce the risk of noise;
(d) on the light of the above points, Entity Linking acts as a surrogate of syntactic parsing, thus complying with our initial claim.

[17] `http://thewikimachine.fbk.eu/`

We still foresee further investigation of the other strategies for scaling besides the use case. Specifically, we believe that the refinement of the chunker grammar would be the most beneficial approach: POS tagging is already involved into the system architecture, thus allowing to concentrate the engineering costs on the grammar only.

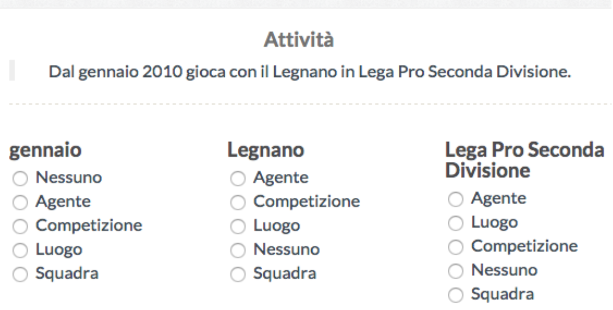
6.2. *Training Set Creation*

We apply a one-step, bottom-up approach to let the crowd perform a full frame annotation over a set of training sentences. In Frame Semantics, lexical ambiguity is represented by the number of frames that a LU may trigger. For instance, `vincere` (to win) conveys TROFEO (trophy) and VITTORIA (victory), thus having an ambiguity value of 2. The idea is to directly elicit the detection of *core* FEs, which are the essential items allowing to discriminate between frames. In this way, we are able to both annotate the FEs and let the correct frame emerge, thus also disambiguating the LU. The objective is achieved as follows: given a sentence s holding a LU with frame set F and set cardinality (i.e., ambiguity value) n, we solicit n annotations of s, and associate each one to the core FEs of each frame $f \in F$. We allow workers to select the `None` answer, and infer the correct frame based on the amount of `None`.

The training set is randomly sampled from the input corpus and contains $3,055$ items. The outcome is the same amount of frame examples and $55,385$ FE examples. The task is sent to the CrowdFlower platform.

6.2.1. *Crowdsourcing Caveats*

Swindles represent a widespread pitfall of crowdsourcing services: workers are usually rewarded a very low monetary amount (i.e., a few cents) for jobs that can be finalized with a single mouse click. Therefore, the results are likely to be excessively contaminated by random answers. CrowdFlower tackles the problem via *test questions*,[18] namely data units which are premarked with the correct response. If a worker fails to meet a given minimum accuracy threshold,[19] he or she will be labeled as *untrusted* and his or her contribution will be automatically rejected.



Attività

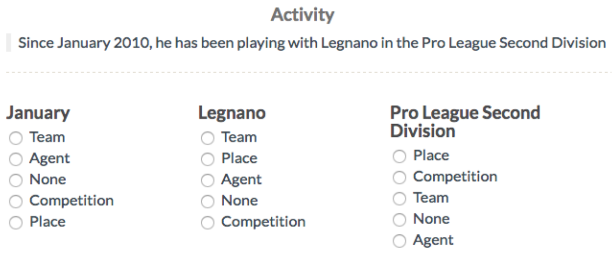
Dal gennaio 2010 gioca con il Legnano in Lega Pro Seconda Divisione.

gennaio
○ Nessuno
○ Agente
○ Competizione
○ Luogo
○ Squadra

Legnano
○ Agente
○ Competizione
○ Luogo
○ Nessuno
○ Squadra

Lega Pro Seconda Divisione
○ Agente
○ Luogo
○ Competizione
○ Nessuno
○ Squadra

Figure 3. Worker interface example



Activity

Since January 2010, he has been playing with Legnano in the Pro League Second Division

January
○ Team
○ Agent
○ None
○ Competition
○ Place

Legnano
○ Team
○ Place
○ Agent
○ None
○ Competition

Pro League Second Division
○ Place
○ Competition
○ Team
○ None
○ Agent

Figure 4. Worker interface example translated in English

6.2.2. *Task Design*

We ask the crowd to (a) read the given sentence, (b) focus on the "topic" (i.e., the potential frame that disambiguates the LU) written above it, and (c) assign the correct "label" (i.e., the FE) to each "word" (i.e., unigram) or "group of words" (i.e., n-grams) from the multiple choices provided below each n-gram. Figure 3 displays the front-end interface of a sample sentence, with Figure 4 being its English translation.

During the preparation phase of the task input data, the main challenge is to automatically provide the crowd with relevant candidate FE text chunks, while minimizing the production of noisy ones. To tackle this, we experimented with the following chunking strategies:

- third-party full-stack NLP pipeline, namely TEXTPRO [46] for Italian, by extracting nominal chunks with the CHUNKPRO module;[20]
- custom noun phrase chunker via a context-free grammar;
- EL surface forms;

We surprisingly observed that the full-stack pipeline outputs a large amount of noisy chunks, besides being the slowest strategy. On the other hand, the custom

[18] `https://success.crowdflower.com/hc/en-us/articles/202703305-Getting-Started-Glossary-of-Terms#test_question`

[19] `https://success.crowdflower.com/hc/en-us/articles/202702975-Job-Settings-Guide-To-Test-Question-Settings-Quality-Control`

[20] `http://textpro.fbk.eu/`

Table 2

Training set crowdsourcing task outcomes. Cf. Section 6.2.1 for explanations of CrowdFlower-specific terms

Sentences	3,111
Test questions	56
Trusted judgments	9,198
Untrusted judgments	972
Total cost	152.46 $

chunker was the fastest one, but still too noisy to be crowdsourced. EL resulted in the best trade-off, and we adopted it for the final task.

The task parameters are as follows:

- we set 3 judgments per sentence to enable the computation of an agreement based on majority vote;
- the pay sums to 5 $ cents per page, where one page contains 5 sentences;
- we limit the task to Italian native speakers only by targeting the Italian country and setting the required language skills to Italian;
- the minimum worker accuracy is set to 70% in quiz mode (i.e., the warm-up phase where workers are only shown gold units and are recruited according to their accuracy) and relaxed to 65% in work mode (i.e., the actual annotation phase) to avoid extra cost in terms of time and expenses to collect judgments;
- on account of a personal calibration, the minimum time per page threshold is set to 30 seconds, which allows to automatically discard a contributor when triggered;
- we set the maximum number of judgments per contributor to 280, in order to prevent each contributor from answering more than once on a given sentence, while avoiding to remove proficient contributors from the task.

The outcomes are resumed in Table 2.

Finally, the crowdsourced annotation results are processed and translated into a suitable format to serve as input training data for the classifier.

6.3. Frame Classification: Features

We train our classifiers with the following linguistic features, in the form of bag-of-features vectors:

1. *both classifiers*: for each input word token, both the token itself (bag of terms) and the lemma (bag of lemmas);
2. *FE classifier*: contextual sliding window of width 5 (i.e., 5-gram, for each token, consider the 2 previous and the 2 following ones);
3. *frame classifier*: we implement our bottom-up frame annotation approach, thus including the set of FE labels (bag of roles) to help this classifier induce the frame;
4. *gazetteer*: defined as a map of key-value pairs, where each key is a feature and its value is a list of n-grams, we automatically build a wide-coverage gazetteer with relevant DBpedia ontology (DBPO) classes as keys (e.g., SoccerClub) and instances as values (e.g., Juventus), by way of a query to the target KB.

7. Numerical Expressions Normalization

During the pilot crowdsourcing annotation experiments, we noticed a low agreement on numerical FEs. This is likely to stem from the FE labels interpretation: workers got particularly confused by TIME and DURATION, which explains the low agreement. Moreover, asking the crowd to label such frequently occurring FEs would represent a considerable overhead, resulting in a higher temporal cost (i.e., more annotations per sentence) and lower overall annotation accuracy. Hence, we opted for the implementation of a rule-based system to detect and normalize numerical expressions. The normalization process takes as input a numerical expression such as a date, a duration, or a score, and outputs a transformation into a standard format suitable for later inclusion into the target KB.

The task is not formulated as a classification one, but we argue it is relevant for the completeness of the extracted facts: rather, it is carried out via matching and transformation rule pairs. Given for instance the input expression tra il 1920 e il 1925 (between 1920 and 1925), our normalizer first matches it through a regular expression rule, then applies a transformation rule complying to the XML Schema Datatypes[21] (typically dates and times) standard, and finally produces the following output:[22]

```
duration: "P5Y"^^xsd:duration
start: "1920"^^xsd:gYear
end: "1925"^^xsd:gYear
```

[21] http://www.w3.org/TR/xmlschema-2/

[22] We use the xsd prefix as a short form for the full URI http://www.w3.org/2001/XMLSchema#

All rule pairs are defined with the programming language-agnostic YAML[23] syntax. The pair for the above example is as follows. Regular Expression:

```
tra il (?P<y1>\ d{{2,4}}) e il (?P<y2>\ d{{2,4}})
```

Transformation:

```
{
  ‘duration’:
‘"P{}Y"^^<{}>’.format(
int(match.group(‘y2’)) - int(match.group(‘y1’)),
schema[‘duration’]
),
  ‘start’:
‘"{}"^^<{}>’.format(
abs_year(match.group(‘y1’)), schema[‘year’]
),
  ‘end’:
‘"{}"^^<{}>’.format(
abs_year(match.group(‘y2’)), schema[‘year’]
)
}
```

In total, we have identified 21 rules, which are publicly available for consultation.[24]

8. Dataset Production

The integration of the extraction results into DBpedia requires their conversion to a suitable data model, i.e., RDF. Frames intrinsically bear N-ary relations through FEs, while RDF naturally represents binary relations. Hence, we need a method to express FEs relations in RDF, namely *reification*. This can be achieved in multiple ways:

- standard reification;[25]
- N-ary relations,[26] an application of Neo-Davidsonian representations [52,51], with similar efforts [20, 31];
- named graphs.[27]

A recent overview [30] highlighted that all the mentioned strategies are similar with respect to query performance. Given as input n frames and m FEs, we argue that:

- standard reification is too verbose, since it would require $3(n + m)$ triples;
- applying Pattern 1 of the aforementioned W3C Working Group note to N-ary relations would allow us to build $n + m$ triples;
- named graphs can be used to encode provenance or context metadata, e.g., the article URI from where a fact was extracted. In our case however, the fourth element of the quad would be the frame (which represents the context), thus boiling down to minting $n + m$ quads instead of triples;

We opted for the less verbose strategy, namely N-ary relations. Given the running example sentence `In Euro 1992, Germany reached the final, but lost 0–2 to Denmark`, classified as a DEFEAT frame and embedding the FEs WINNER, LOSER, COMPETITION, SCORE, we generate RDF as per the following Turtle serialization:

```
:Germany :defeat :Defeat_01 .
:Defeat_01
   :winner :Denmark ;
   :loser :Germany ;
   :competition :Euro_1992 ;
   :score "0-2" .
```

We add an extra instance type triple to assign an ontology class to the reified frame, as well as a provenance triple to indicate the original sentence:

```
:Defeat_01
   a :Defeat ;
   :extractedFrom "In Euro 1992,
      Germany reached the final,
      but lost 0-2 to Denmark"@it .
```

In this way, the generated statements amount to $n + m + 2$.

It is not trivial to decide on the subject of the main frame statement, since not all frames are meant to have exactly one core FE that would serve as a plausible logical subject candidate: most have many, e.g., FINISH_-COMPETITION has COMPETITION, COMPETITOR and OPPONENT as core FEs in FrameNet. Therefore, we tackle this as per the following assumption: given the encyclopedic nature of our input corpus, both the logical and the topical subjects correspond in each document. Hence, each candidate sentence inherits the document subject. We acknowledge that such assumption strongly depends on the corpus: it applies to entity-centric documents, but will not perform well for general-purpose ones such as news articles. However, we believe it is still a valid in-scope solution fitting our scenario.

[23] `http://www.yaml.org/spec/1.2/spec.html`

[24] `https://github.com/dbpedia/fact-extractor/blob/master/date_normalizer/regexes.yml`

[25] `http://www.w3.org/TR/2004/REC-rdf-primer-20040210/#reification`

[26] `http://www.w3.org/TR/swbp-n-aryRelations/`

[27] `http://www.w3.org/TR/rdf11-concepts/`

8.1. Confidence Scores

Besides the fact datasets, we also keep track of confidence scores and generate additional datasets accordingly. Therefore, it is possible to filter facts that are not considered as confident by setting a suitable threshold. When processing a sentence, our pipeline outputs two different scores for each FE, stemming from the entity linker and the supervised classifier. We merge both signals by calculating the F-score between them, as if they were representing precision and recall, in a fashion similar to the standard classification metrics. The global fact score can be then produced via an aggregation of the single FE scores in multiple ways, namely: (a) arithmetic mean; (b) weighted mean based on core FEs (i.e., they have a higher weight than extra ones); (c) harmonic mean, weighted on core FEs as well.

The reader may refer to Section 11.5 for a distributional analysis of these scores over the output dataset.

9. Baseline Classifier

To enable a performance evaluation comparison with the supervised method, we developed a rule-based algorithm that handles the full frame and FEs annotation. The main intuition is to map FEs defined in the frame repository to ontology classes of the target KB: such mapping serves as a set of rule pairs $(FE, class)$, e.g., (Winner, `SoccerClub`). In the FrameNet terminology, this is homologous to the assignment of *semantic types* to FEs: for instance, in the Activity frame, the Agent is typed with the generic class `Sentient`. The idea would allow the implementation of the bottom-up one-step annotation flow described in [25]: to achieve this, we run EL over the input sentences and check whether the attached ontology class metadata appear in the frame repository, thus fulfilling the FE classification task.

Besides that, we exploit the notion of core FEs: this would cater for the frame disambiguation part. Since a frame may contain at least one core FE, we proceed with a *relaxed* assignment, namely we set the frame if a given input sentence contains at least one entity whose ontology class maps to a core FE of that frame. The implementation workflow is illustrated in Algorithm 1: it takes as input the set S of sentences, the frame repository F embedding frame and FEs labels, core/non-core annotations and rule pairs, and the set L of trigger LU tokens.

Algorithm 1 Rule-based baseline classifier

```
Input: $S$;  $F$;  $L$
Output: $C$
$C \leftarrow \emptyset$
for all $s \in S$ do
    $E \leftarrow entityLinking(s)$
    $T \leftarrow tokenize(s)$
    for all $t \in T$ do
        if $t \in L$ then #Check whether a sentence
          token matches a LU token
            for all $f \in F$ do
                $core \leftarrow$ false
                $O \leftarrow getLinkedEntityClasses(E)$
                for all $o \in O$ do
                    $fe \leftarrow lookup(f)$ #Get the FE that
                      maps to the current linked entity class
                    $core \leftarrow checkIsCore(fe)$
                end for
                if $core$ then #Relaxed classification
                    $c \leftarrow [s, f, fe]$
                    $C \leftarrow C \cup \{c\}$
                else
                    continue #Skip to the next frame
                end if
            end for
        end if
    end for
end for
return $C$
```

It is expected that the relaxed assignment strategy will not handle the overlap of FEs across competing frames that are evoked by a single LU. Therefore, if at least one core FE is detected in multiple frames, the baseline makes a random assignment for the frame. Furthermore, the method is not able to perform FE classification in case different FEs share the ontology class (e.g., both Winner and Loser map to `SoccerClub`): we opt for a FE random guess as well.

10. Evaluation

We assess our main research contributions through the analysis of the following aspects:

- Classification performance;
- T-Box property coverage extension;
- A-Box statements addition;
- final fact correctness.

Table 3

Frame Elements (FEs) classification performance evaluation over a gold standard of 500 random sentences from the Italian Wikipedia corpus. The average crowd agreement score on the gold standard amounts to .916

Approach	**Lenient**			**Strict**		
	P	**R**	**F1**	**P**	**R**	**F1**
Baseline	73.48	65.83	69.45	67.68	63.79	65.68
Supervised	83.33	75.00	78.94	73.59	66.66	69.96

10.1. Classification Performance

We assess the overall performance of the baseline and the supervised systems over a gold standard dataset. We randomly sampled 500 sentences containing at least one occurrence of our use case LU set from the input corpus. We first outsourced the annotation to the crowd as per the training set construction and the results were further manually validated twice by the authors. CrowdFlower provides a report including an agreement score for each answer, computed via majority vote weighted by worker trust: we calculated the average among the whole evaluation set, obtaining a value of .916.

With respect to the FEs classification task, we proceed with 2 evaluation settings, depending on how FE text chunks are treated, namely:

- **lenient**, where the predicted ones at least *partially* match the expected ones;
- **strict**, where the predicted ones must *perfectly* match the expected ones.

Table 3 illustrates the outcomes. FE measures are computed as follows: (1) a true positive is triggered if the predicted label is correct and the predicted text chunk matches the expected one (according to each setting); chunks that should not be labeled are marked with a “O” and (2) not counted as true positives if the predicted ones are correct, but (3) indeed counted as false positives in the opposite case. The high frequency of “O” occurrences (circa 80% of the total) in the gold standard actually penalizes the system, thus providing a more challenging evaluation playground.

On the other hand, the frame classification task does not need to undergo chunk assessment, since it copes with the whole input sentence. Therefore, the lenient and strict settings are not applicable, and we proceed with a standard evaluation. The results are reported in Table 4.

10.1.1. Supervised Classification Performance Breakdown

Figure 5 and Figure 7 respectively display the FE and frame classification confusion matrices: they are normalized such that the sum of elements in the same row is 1. Since we highlight the cells through a color scale, the normalization is needed to avoid too similar color nuances that would originate from absolute results.

FEs. We observe that COMPETIZIONE is frequently mistaken for PREMIO and ENTITÀ, while rarely for TEMPO and DURATA, or just missed. On the other hand, TEMPO is mistaken for COMPETIZIONE: our hypothesis is that competition mentions, such as `World Cup 2014`, are disambiguated as a whole entity by the linker, since a specific target Wikipedia article exists. However, it overlaps with a temporal expression, thus confusing the classifier. AGENTE is often mistaken for ENTITÀ, due to their equivalent semantic type, which is always a person.

Frames. We note that ATTIVITÀ is often mistaken for STATO or not classified at all: in fact, the difference between these two frames is quite subtle with respect to their sense. The former is more generic and could also be labeled as CAREER: if we viewed it in a frame hierarchy, it would serve as a super-frame of the latter. The latter instead encodes the development modality of a soccer player’s career, e.g., when he remains unbound from some team due to contracting issues. Hence, we may conclude that distinguishing between these frames is a challenge even for humans.

Furthermore, frames with no FEs are classified as “O”, thus considered wrong despite the correct prediction. VITTORIA is almost never mistaken for TROFEO: this is positively surprising, since the FE COMPETIZIONE (frame VITTORIA) is often mistaken for PREMIO (frame TROFEO), but those FEs do not seem to affect the frame classification. Again, such FE distinction must take into account a delicate sense nuance, which is hard for humans as well.

Table 4

Frame classification performance evaluation over a gold standard of 500 random sentences from the Italian Wikipedia corpus. The average crowd agreement score on the gold standard amounts to .916

Approach	**P**	**R**	**F1**
Baseline	74.25	62.50	67.87
Supervised	84.35	82.86	83.60

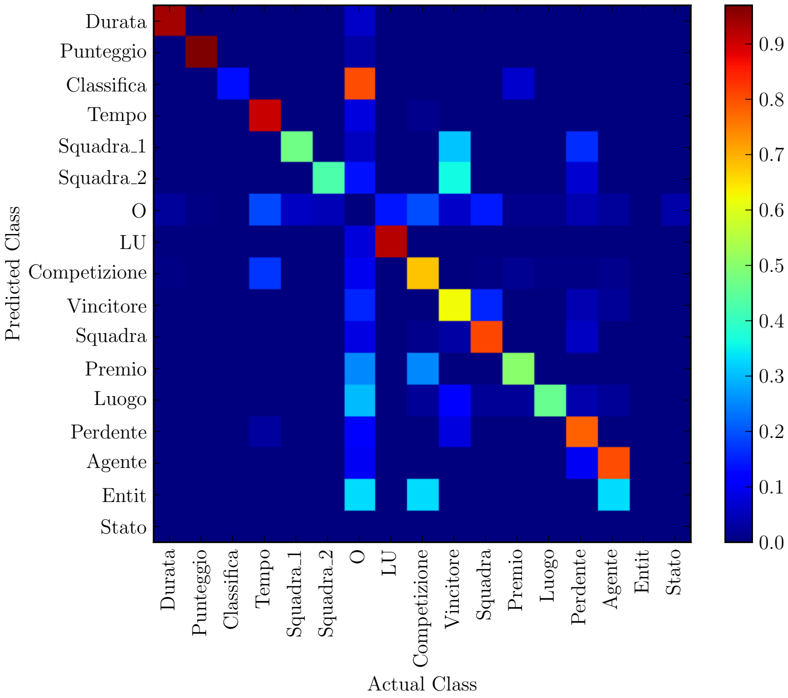


Figure 5. Supervised FE classification normalized confusion matrix, lenient evaluation setting. The color scale corresponds to the ratio of predicted versus actual classes. Normalization means that the sum of elements in the same row must be 1.0

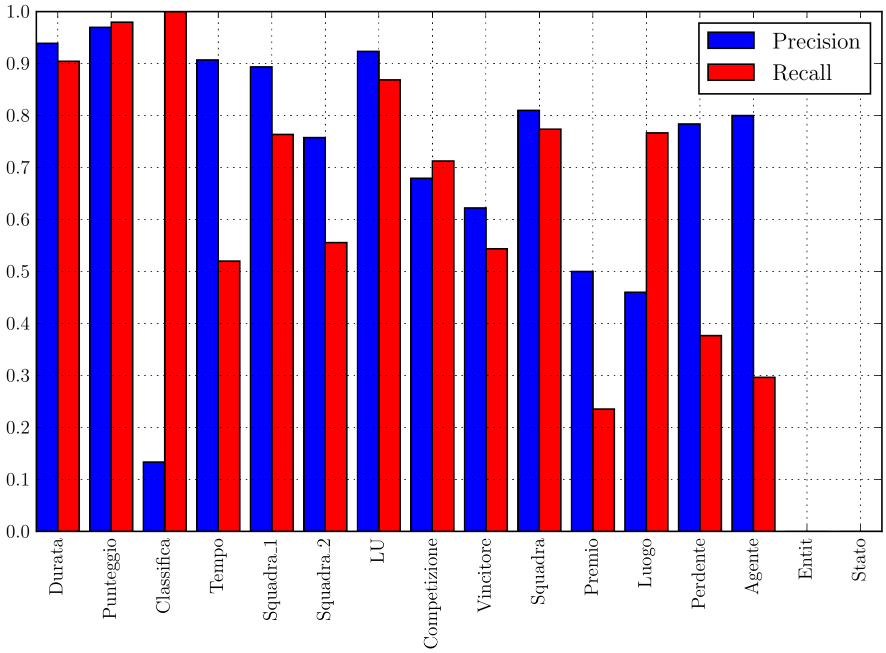


Figure 6. Supervised FE classification precision and recall breakdown, lenient evaluation setting

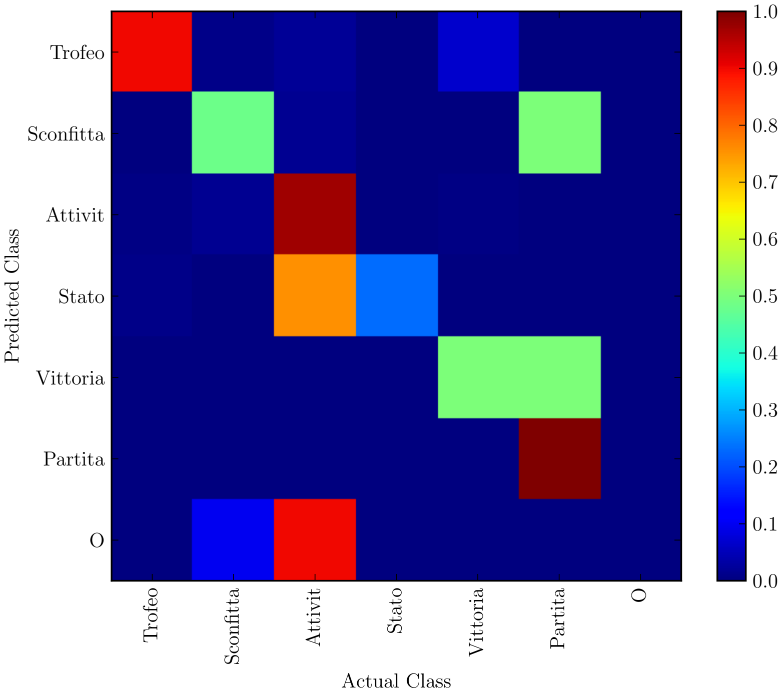


Figure 7. Supervised frame classification normalized confusion matrix. The color scale corresponds to the ratio of predicted versus actual classes. Normalization means that the sum of elements in the same row must be 1.0

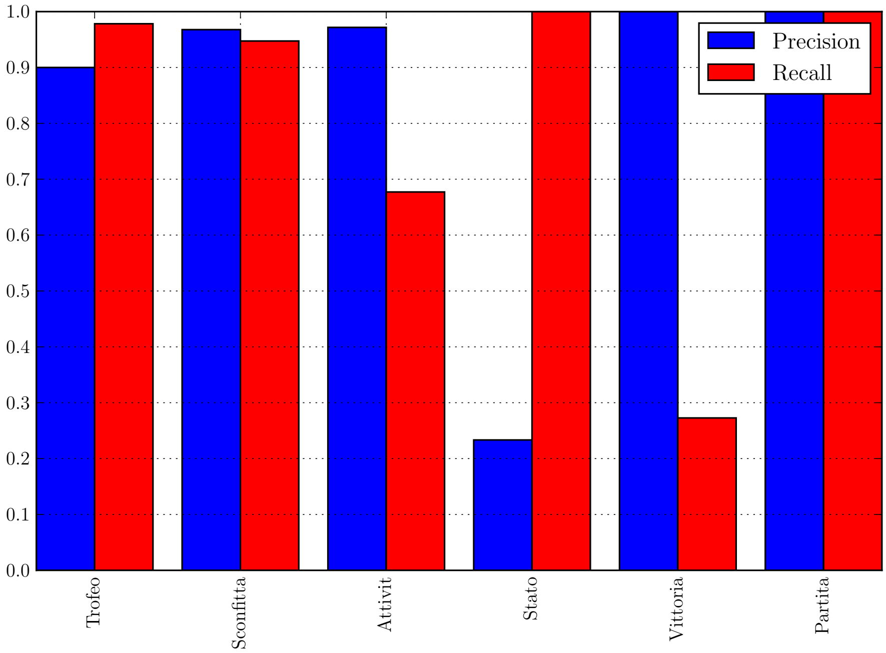


Figure 8. Supervised frame classification precision and recall breakdown

Table 5

Lexicographical analysis of the Italian Wikipedia soccer player sub-corpus

Stems (frequency %)	**Candidate frames (FrameNet)**
gioc (47), partit (39), campionat (34), stagion (36), presen (30), disput (20), serie (14), nazional (13), titolar (13), competizion (5), scend (5), torne (5)	COMPETITION
pass (24), trasfer (19), prest (15), contratt (11)	ACTIVITY_START, EMPLOYMENT_START
termin (12), contratt, ced (10), lasc (6), vend (2)	ACTIVITY_FINISH, EMPLOYMENT_END
gioc, disput (20), scend	FINISH_GAME
campionat, stagion, serie, nazional, competizion, torne	FINISH_COMPETITION
vins/vinc (18), pers/perd (11), sconfi (8)	BEAT_OPPONENT, FINISH_GAME
vins/vinc, conquis (8), otten (7), raggiun (6), aggiud (2)	WIN_PRIZE, PERSONAL_SUCCESS

Figure 6 and Figure 8 respectively plot the FE and frame classification performance, broken down to each label.

10.2. T-Box Enrichment

One of our main objectives is to extend the target KB ontology with new properties on existing classes. We focus on the use case and argue that our approach will have a remarkable impact if we manage to identify non-existing properties. This would serve as a proof of concept which can ideally scale up to all kinds of input. In order to assess such potential impact in discovering new relations, we need to address the following question: *"which extractable relations are not already mapped in DBPO or do not even exist in the raw infobox properties datasets?"*. Table 5 illustrates an empirical lexicographical study gathered from the Italian Wikipedia soccer player sub-corpus (circa $52,000$ articles). It contains occurrence frequency percentages of word stems (in descending order) that are likely to trigger domain-relevant frames, thus providing a rough overview of the extraction potential.

The corpus analysis phase (cf. Section 4) yielded a ranking of LUs evoking the frames ACTIVITY, DEFEAT, MATCH, TROPHY, STATUS, and VICTORY: these frames would serve as ontology property candidates, together with their embedded FEs. DBPO already has most of the classes that are needed to represent the main entities involved in the use case: SoccerPlayer, SoccerClub, SoccerManager, SoccerLeague, SoccerTournament, SoccerClubSeason, SoccerLeagueSeason, although some of them lack an exhaustive description (cf. SoccerClubSeason[28] and SoccerLeagueSeason).[29]

For each of the 7 aforementioned DBPO classes, we computed the amount and frequency of ontology and raw infobox properties by querying the Italian DBpedia endpoint. Results (in ascending order of frequency) are publicly available,[30] and Figure 9 illustrates their distribution. The horizontal axis stands for the normalized (log scale) frequency, encoding the current usage of properties in the target KB; the vertical axis represents the ratio (which we call coverage) between the position of the property in the ordered result set of the query and the total amount of distinct properties (i.e., the size of the result set). Properties with a null frequency are ignored.

First, we observe a lack of ontology property usage in 4 out of 7 DBPO classes, probably due to missing mappings between Wikipedia template attributes and DBPO. On the other hand, the ontology properties have a more homogenous distribution compared to the raw ones: this serves as an expected proof of concept, since the main purpose of DBPO and the ontology mappings is to merge heterogenous and multilingual Wikipedia template attributes into a unique representation. On average, most raw properties are concentrated below coverage and frequency threshold values of 0.8 and 4 respectively: this means that roughly 80% are rarely used, and the log scale further highlights the evidence. While ontology properties are better distributed, most still do not reach a high coverage/frequency trade-off,

[28] http://mappings.dbpedia.org/server/ontology/classes/SoccerClubSeason

[29] http://mappings.dbpedia.org/server/ontology/classes/SoccerLeagueSeason

[30] http://it.dbpedia.org/downloads/fact-extraction/soccer_statistics/

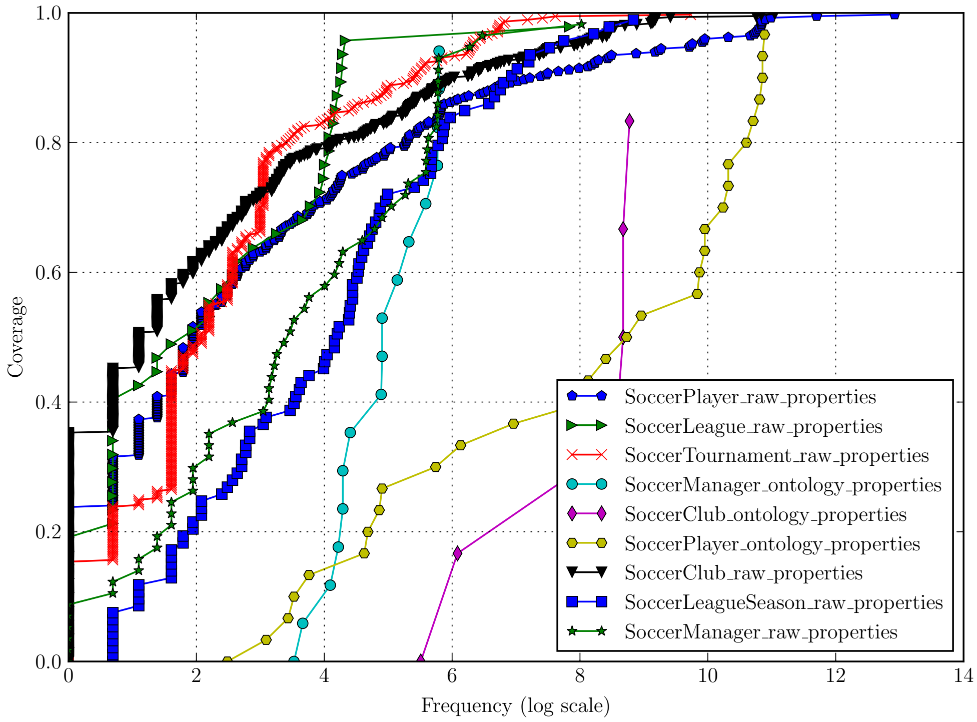


Figure 9. Italian DBpedia soccer property statistics

except for SoccerPlayer, which benefits from both rich data (cf. Section 2) and mappings.[31]

On the light of the two analyses discussed above, it is clear that our approach would result in a larger variety and finer granularity of facts than those encoded into Wikipedia infoboxes and DBPO classes. Moreover, we believe the lack of dependence on infoboxes would enable more flexibility for future generalization to sources beyond Wikipedia.

Subsequent to the use case implementation, we manually identified the following mappings from frames and FEs to DBPO properties:

- Frames: (ACTIVITY, careerStation), (AWARD, award), (STATUS, playerStatus);
- FEs: (TEAM, team), (SCORE, score), (DURATION, [duration, startYear, endYear]).

Our system would undeniably benefit from a property matching facility to discover more potential mappings, although a research contribution in ontology alignment is out of scope for this work. In conclusion, we claim that 3 out 6 frames and 12 out of 15 FEs represent novel T-Box properties.

10.3. A-Box Population

Our methodology enables a simultaneous T-Box and A-Box augmentation: while frames and FEs serve as T-Box properties, the extracted facts feed the A-Box part. Out of $49,063$ input sentences, we generated a total of $213,479$ and $216,451$ triples (i.e., with a 4.35 and 4.41 ratio per sentence) from the supervised and the baseline classifiers respectively. 52% and 55% circa are considered *confident*, namely facts with confidence scores (cf. Section 8.1) above the dataset average threshold.

To assess the domain coverage gain, we can exploit two signals: (a) the amount of produced novel data with respect to pre-existing T-Box properties and (b) the overlap with already extracted assertions, regardless of their origin (i.e., whether they stem from the raw infobox or the ontology-based extractors). Given the same Italian Wikipedia dump input dating 21 January 2015, we ran both the baseline and the supervised fact extraction, as well as the DBpedia extraction framework to produce an Italian DBpedia chapter release, thus enabling the coverage comparison.

Table 6 describes the analysis of signal (a) over the 3 frames that are mapped to DBPO properties. For each property and dataset, we computed the amount of avail-

[31] http://mappings.dbpedia.org/index.php/Mapping_it:Sportivo

Table 6

Relative A-Box population gain compared to pre-existing T-Box property assertions in the Italian DBpedia chapter

Property	Dataset	Assertions (#)	Gain (%)
`careerStation`	DBpedia	2,073	N.A.
	Baseline all	20,430	89.8
	Supervised all	26,316	92.12
`award`	DBpedia	7,755	N.A.
	Baseline all	4,953	-56.57
	Supervised all	10,433	25.66
`playerStatus`	DBpedia	0	N.A.
	Baseline all	0	0
	Supervised all	26	100

able assertions and reported the gain relative to the fact extraction datasets. Although we considered the whole Italian DBpedia KB in these calculations, we observe that it has a generally low coverage with respect to the analyzed properties, probably due to missing ontology mappings. For instance, the amount of assertions is always zero if we analyze the use case subset only, as no specific relevant mappings (e.g., `Carriera_sportivo`[32] to `careerStation`) currently exist. We view this as a major achievement, since our automatic approach also serves as a substitute for the manual mapping procedure.

Table 7 shows the results for signal (b). To obtain them, we proceeded as follows.

1. slice the use case DBpedia subset;
2. gather the subject-object patterns from all datasets. Properties are not included, as they are not comparable;
3. compute the patterns overlap between DBpedia and each of the fact extraction datasets (including the confident subsets);
4. compute the gain in terms of novel assertions relative to the fact extraction datasets.

The A-Box enrichment is clearly visible from the results, given the low overlap and high gain in all approaches, despite the rather large size of the DBpedia use case subset, namely $6,167,678$ assertions.

10.4. Final Fact Correctness

We estimate the overall correctness of the generated statements via an empirical evaluation over a sample of the output dataset. In this way, we are able to conduct a more comprehensive error analysis, thus isolating the performance of those components that play a key role in the extraction of facts: the Frame Semantics classifier, the numerical expression normalizer, and an external yet crucial element, i.e., the entity linker.

To achieve so, we randomly selected 10 instances for each frame from the supervised dataset and retrieve all the related triples. We excluded instance type triples (cf. Section 8), which are directly derived from the reified frame ones. Then, we manually assessed the validity of each triple element and assigned it to the component responsible for its generation. Finally, we checked the correctness of the whole triple.

More formally, given the evaluation set of triples E, the frame predicates set F, the non-numerical FE predicates set $\bar{N}$, and the numerical FE predicates set N (cf. Section 5), relevant triple elements are added to the classifier C, the normalizer N, the linker L, and to the set of all facts A as follows.

$$\begin{array}{cc} \multicolumn{2}{c}{E \subseteq S \times P \times O;} \\ P = F \cup \bar{N} \cup N; & F \cap \bar{N} \cap N = \emptyset; \\ p_c \in F \cup \bar{N}; & p_n \in N; \\ O = O_c \cup O_n; & O_c \cap O_n = \emptyset; \\ o_c \in O_c; & o_n \in O_n; \end{array}$$

$$\begin{array}{c} \forall (s,p,o) \in E \text{ let} \\ C \leftarrow C \cup \{(p_c, o_c)\};\ N \leftarrow N \cup \{(p_n, o_n)\}; \\ L \leftarrow L \cup \{o_c\}; \qquad A \leftarrow A \cup \{(s,p,o)\} \end{array}$$

Table 8 summarizes the outcomes.

10.4.1. Discussion

First, we observe that all the results but the linker are in line with our classification performance assessments detailed in Section 10.1.1. Accordingly, we notice that most of the errors involve the linker. More specifically, we summarize below an informal error analysis:

- generic dates appearing without years (as in `the 13th of August`) are resolved to their Wikipedia

Table 7

Overlap with pre-existing assertions in the Italian DBpedia chapter and relative gain in A-Box population

Dataset	Overlap (#)	Gain (%)
Baseline all	3,341	98.2
Supervised all	4,546	97.4
Baseline confident	2,387	97.6
Supervised confident	2,841	96.8

[32] `https://it.wikipedia.org/wiki/Template:Carriera_sportivo`

Table 8

Fact correctness evaluation over 132 triples randomly sampled from the supervised output dataset. Results indicate the ratio of correct data for the whole fact (**All**) and for triple elements produced by the main components of the system, namely: **Classifier**, as per Figure 2, part 2(c), and Section 6; **Normalizer**, as per Figure 2, part 2(d), and Section **??**; **Linker**, external component, as per Section 6.

Classifier	Normalizer	Linker	All
.763	.820	.430	.727

page.[33] These occurrences are then wrongly classified as COMPETIZIONE, consistently with what we remarked in Section 10.1.1;
- country names, e.g., `Sweden` are often linked to their national soccer team or to the major national soccer competition. This seems to mislead the classifier, which assigns a wrong role to the entity, instead of PLACE;
- the generic adjective `Nazionale` (national) is always linked to the Italian national soccer team, even though the sentence often contains enough elements to understand the correct country;
- some yearly intervals, e.g., `2010-2011` are linked to the corresponding season of the major Italian national soccer competition.

Unfortunately, the linker tends to assign a fairly high confidence to these matches and so does the classifier, which assumes correct linking of entities. This leads to many assertions with undeserved high scores and underlines how important Entity Linking is in our pipeline.

11. Observations

We pinpoint and discuss here a list of notable aspects of this work.

11.1. LU Ambiguity

We acknowledge that the number of frames per LU in our use case repository may not be exhaustive to cover the potentially higher LU ambiguity. For instance, `giocare` (to play) may trigger an additional frame depending on the context (as in the sentence `to play as a defender`); `esordire` (to start out) may also trigger the frame PARTITA (match). Nevertheless, our one-step annotation approach is agnostic to the frame repository. Consequently, we expect that the LU ambiguity would not be an issue. Of course, the more a LU is ambiguous, the more expensive becomes the crowdsourcing job (cf. Section 6.2).

11.2. Manual Intervention Costs

Despite its low cost, we admit that crowdsourcing does not conceptually bypass the manual effort needed to create the training set: workers are indeed human annotators. However, we argue that the price can decrease even further by virtue of an automatic communication with the CrowdFlower API. This is already accomplished in the ongoing STREPHIT project, where we programmatically create jobs, post them, and pull their results. Hence, we may regard crowdsourcing as an activity that does not imply any direct manual intervention by whoever runs the pipeline, if we exclude a minor quantity of test annotations, which are essential to reject cheaters.

Even though we recognize that the use case frame repository is hand-curated, we would like to emphasize that (a) it is intended as a test bed to assess the validity of our approach, and (b) its generalization should instead maximize the reuse of available resources. This is currently implemented in the STREPHIT project, where we fully leverage FrameNet to look up relevant frames given a set of LUs.

11.3. NLP Pipeline Design

On account of our initial claim on the use of a shallow NLP machinery, we motivate below the choice of stopping to the grammatical layer. The decision essentially emanates from (1) the sentence selection phase, where we investigated several strategies, and (2) the construction of the crowdsourcing jobs, where we concurrently (2a) maximized the simplicity to smooth the way for the laymen workers, and (2b) automatically generated the candidate annotation chunks.

- *Chunking* is substituted by Entity Linking, as explored in Section 6.2.2;
- *Syntactic parsing* dramatically affects the computational costs, as shown in Table 9 and discussed in Section 6.1. Yet, we suppose that it could probably improve the performance in terms of recall. Given the KB population task, we still argue that precision should be made a priority, in order to produce high quality datasets;
- *Semantic Role Labeling* is not a requirement, since our system replaces this layer, as described in Section 6.

[33] `https://en.wikipedia.org/wiki/August_13`

Table 9
Comparative results of the *Syntactic* sentence extraction strategy against the Sentence *Splitter* one, over a uniform sample of a corpus gathered from 53 Web sources, with estimates over the full corpus.

Strategy	# Documents	# Extracted	Cost
Splitter	7,929	13,846	1m 13s
Syntactic		41,205	6h 15m 49s
Splitter	504,189	899,159	1h 19m
Syntactic		2,675,853	16d 22h 45m 32s

11.4. Simultaneous T-Box and A-Box Augmentation

The Fact Extractor is conceived to extract factual information from text: as such, its primary output is a set of assertions that naturally feed the target KB A-Box. The T-Box enrichment is an intrinsic consequence of the A-Box one, since the latter provides evidence of new properties for the former. In other words, we adopt a data-driven method, which implies a bottom-up direction for populating the target KB. It is the duty of the corpus analysis module (Section 4) to understand the most meaningful relations between entities from the very bottom, i.e., the corpus. After that, the system proceeds upwards and translates the classification results into A-Box statements. These are already structured to ultimately carry the properties into the top layer of the KB, i.e., the T-Box.

11.5. Confidence Scores Distribution

Table 10 presents the cumulative (i.e., all FEs and frames aggregated) statistical distribution of confidence scores as observed in the gold standard. If we dig into single scores, we notice that the classifier usually outputs very high values for O and LU chunks, while average scores for other FEs range from .821 for COMPETITION to .594 for WINNER, down to .488 for LOSER. On the other hand, EL scores have a relatively high average and a standard deviation of 0.273. In other words, the EL component is prone to set rather optimistic values, which are likely to have an impact on the global score.

Overall, due to the high presence of O chunks (circa 80% of the total), the EL and the classifier scores roughly match for each FE, and so do the final ones computed with the strategies introduced in Section 8.1. Assigning different weights to core and extra FEs has little impact on the global scores as well, varying their value by only 1 or 2% in both the weighted and the harmonic means. The arithmetic and weighted means yield the most optimistic global scores, averaging at .83 over the output dataset, while the harmonic mean settles at .75.

11.6. Scaling Up

Our approach has been tested on the Italian language, a specific domain, and with a small frame repository. Hence, we may consider the use case implementation as a monolingual closed-domain information extraction system. We outline below the points that need to be addressed for scaling up to multilingual open information extraction:

1. *Language*: training data availability for POS tagging and lemmatization. The LUs automatically extracted through the corpus analysis phase should be projected to a suitable frame repository;
2. *Domain*:
   - Baseline: mapping between FEs and target KB ontology classes;
   - Supervised:
     - financial resources for the crowdsourced training set construction, on average 4.79 $ cents per annotated sentence;
     - adapt the query to generate the gazetteer.

11.7. Crowdsourcing Generalization

With the Wikidata commitment in mind (Section 1), we aim at expanding our approach towards a corpus of non-Wikimedia Web sources and a broader domain. This entails the generalization of the crowdsourcing step. Overall, it has been proven that the laymen execute natural language tasks with reasonable performances [54]. Specifically, crowdsourcing Frame Semantics annotation has been recently shown to be feasible by [33]. Furthermore, [5] stressed the importance of eliciting non-expert annotators to avoid the high recruitment cost of linguistics experts. In [25], we further validated the results obtained by [33], and reported satisfactory accuracy as well. Finally, [13] proposed an approach to successfully scale up frame disambiguation.

Table 10
Cumulative confidence scores distribution over the gold standard

Type	Min	Max	Avg	Stdev
Classifier FEs	.181	.999	.945	.124
Classifier Frames	.412	.999	.954	.093
Links	.202	1.0	.697	.273
Global	.227	1.0	.838	.151

On the light of the above references, we argue that the requirement can be indeed satisfied: as a proof of concept, we are working in this direction with StrepHit, where we have switched to a more extensive and heterogeneous input corpus. Here, we focus on a larger set L of LUs, thus $|L| \times n$ frames, where n is the average LU ambiguity. At the time of writing this paper, we are in the process of building the training set.

11.8. Miscellanea

First, if a sentence is not in the gold standard, the supervised classifier should discard it (abstention). Second, the baseline approach may contain rules that are more harmful than beneficial, depending on the target KB reliability: for instance, the `SportsEvent` DBPO class leads to wrongly typed instances, due to the misuse of the template by Wikipedia editors. Finally, both the input corpus and the target KB originate from a relatively small Wikipedia chapter (i.e., Italian, with 1.23 million articles) if compared to the largest one (i.e., English, with almost 5 million articles). Therefore, we recognize that the T-Box and A-Box evaluation results may be proportionally different if obtained with English data.

11.9. Technical Future Work

We report below a list of technical improvements left for planned implementation:

- LUs are handled as unigrams, but n-grams should be considered too;
- tagging n-grams with ontology classes retrieved at the EL step may be an impactful additional feature;
- the gazetteer is currently being matched at the token level, but it may be more useful if run over the whole input (sentence);
- in order to reduce the noise in the training set, we foresee to leverage a sentence splitter and extract 1-sentence examples only;
- further evaluation experiments will also count EL surface forms instead of links;
- the inclusion of the frame confidence would further refine the final confidence score.

12. Related Work

We locate our effort at the intersection of the following research areas:

- Information Extraction;
- KB Construction;
- Open Information Semantification.

12.1. Information Extraction

Although the borders are blurred, nowadays we can distinguish two principal Information Extraction paradigms that focus on the discovery of relations holding between entities: Relation Extraction (RE) and Open Information Extraction (OIE). While they both share the same purpose, their difference relies in the relations set size, either fixed or potentially infinite. It is commonly argued that the main OIE drawback is the generation of noisy data [18,57], while RE is usually more accurate, but requires expensive supervision in terms of language resources [3,55,57].

12.1.1. Relation Extraction

RE traditionally takes as input a finite set R of relations and a document d, and induces assertions in the form $rel(subj, obj)$, where rel represent binary relations between a subject entity $subj$ and an object entity obj mentioned in d. Hence, it may be viewed as a closed-domain paradigm. Recent efforts [4,3,55] have focused on alleviating the cost of full supervision via distant supervision. Distant supervision leverages available KBs to automatically annotate training data in the input documents. This is in contrast to our work, since we aim at enriching the target KB with external data, rather than using it as a source. Furthermore, our relatively cheap crowdsourcing technique serves as a substitute to distant supervision, while ensuring full supervision. Other approaches such as [9,58] instead leverage text that is not covered by the target KB, like we do.

12.1.2. Open Information Extraction

OIE is defined as a function $f(d)$ over a document d, yielding a set of triples (np_1, rel, np_2), where nps are noun phrases and rel is a relation between them. Known complete systems include Ollie [39], ReVerb [21], and NELL [12]. Recently, it has been discussed that cross-utterance processing can improve the performance through logical entailments [2]. This paradigm is called “open” since it is not constrained by any schemata, but rather attempts to learn them from unstructured data. In addition, it takes as input heterogeneous sources of information, typically from the Web.

In general, most efforts have focused on English, due to the high availability of language resources. Approaches such as [22] explore multilingual directions, by leveraging English as a source and applying statistical machine translation (SMT) for scaling up to target languages. Although the authors claim that their system does not directly depend on language resources, we argue that SMT still heavily relies on them. Furthermore, all the above efforts concentrate on binary relations, while we generate n-ary ones: under this perspective, EXEMPLAR [16] is a rule-based system which is closely related to ours.

12.2. Knowledge Base Construction

DBPEDIA [37], FREEBASE [11] and YAGO [32] represent the most mature approaches for automatically building KBs from Wikipedia. Despite its crowd-sourced nature (i.e., mostly manual), WIKIDATA [56] benefits from a rapidly growing community of active users, who have developed several robots for automatic imports of Wikipedia and third-party data. The KNOWLEDGE VAULT [18] is an example of KB construction combining Web-scale textual corpora, as well as additional semi-structured Web data such as HTML tables. Although our system may potentially create a KB from scratch from an input corpus, we prefer to improve the quality of existing resources and integrate into them, rather than developing a standalone one.

Under a different perspective, [42] builds on [14] and illustrate a general-purpose methodology to translate FrameNet into a fully compliant Linked Open Data KB via the SEMION tool [43]. The scope of such work diverges from ours, since we do not target a complete conversion of the frame repository we leverage. On the other hand, we share some transformation patterns in the dataset generation step (Section 8), namely we both link FEs to their frame by means of RDF predicates.

Likewise, FRAMEBASE [52,51] is a data integration effort, proposing a single model based on Frame Semantics to assemble heterogenous KB schemata. This would overcome the knowledge soup issue [26], i.e., the blend of disparate ways in which structured datasets are published. Similarly to us, it utilizes Neo-Davidsonian representations to encode n-ary relations in RDF. Further options are reviewed but discarded by the authors, including singleton properties [41] and schema.org roles.[34] In contrast to our work, FrameBase also provides automatic facilities which bring back the n-ary relations to binary ones for easier queries. The key purpose is to amalgamate different datasets in a unified fashion, thus essentially differing from our KB augmentation objective.

12.3. Open Information Semantification

OIE output can indeed be considered structured data compared to free text, but it still lacks of a disambiguation facility: extracted facts generally do not employ unique identifiers (i.e., URIs), thus suffering from intrinsic natural language polysemy (e.g., `Jaguar` may correspond to the animal or a known car brand).

To tackle the issue, [19] propose a framework that clusters OIE facts and maps them to elements of a target KB. Similarly to us, they leverage EL techniques for disambiguation and choose DBpedia as the target KB. Nevertheless, the authors focus on A-Box population, while we also cater for the T-Box part. Moreover, OIE systems are used as a black boxes, in contrast to our full implementation of the extraction pipeline. Finally, relations are still binary, instead of our n-ary ones.

The main intuition behind LEGALO [49,47] resides in the exploitation of hyperlinks, serving as pragmatic traces of relations between entities, which are finally induced via NLP. The first version [47] focuses on Wikipedia articles, like we do. In addition, it leverages page links that are manually curated by editors, while we consume Entity Linking output. Ultimately, its property matcher module can be leveraged for KB enrichment purposes. Most recently, a new release [49] expands the approach by (a) taking into account hyperlinks from Entity Linking tools, and (b) handling generic free text input. On account of such features, both Legalo and the Fact Extractor are proceeding towards closely related directions. This paves the way to a novel paradigm called *Open Knowledge Extraction* by the authors, which is naturally bound to the Open Information Semantification one introduced in [19]. The only difference again relies on the binary nature of Legalo's extracted relations, which are generated upon FRED [27,48].

FRED is a machine reader that harnesses several NLP techniques to produce RDF graphs out of free text input. It is conceived as a domain-independent middleware enabling the implementation of specific applications. As such, its scope diverges from ours: we instead deliver datasets that are directly integrated into a target KB. In a fashion similar to our work, it encodes knowledge based on Frame Semantics and employs Entity Linking to mint unambiguous URIs for entities and properties. Furthermore, it relies on the same design pattern for expressing n-ary relations in RDF [31]. As opposed

[34] https://www.w3.org/wiki/WebSchemas/RolesPattern

to us, it also encodes NLP tools output via standard formats, i.e., EARMARK [45] and NIF [29]. Additionally, it uses a different natural language representation (i.e., Discourse Representation Structures), which requires a deeper layer of NLP technology, namely syntactic parsing, while we stop to shallow processing via grammatical analysis.

12.4. Further Approaches

12.4.1. Distributional Methods

An additional strand of research encompasses distributional methods: these originate from Lexical Semantics and can be put to use for Information Extraction tasks. Prominent efforts, e.g., [1,8,44] aim at processing corpora to infer features for terms based on their distribution in the text. In a nutshell, similarities between terms can be computed on account of their co-occurrences. This is strictly connected to our supervised classifier, which is modeled in a vector space and takes into account both bag of terms and contextual windows (cf. Section 6.3), in a fashion similar to [1].

12.4.2. Matrix Factorization

Matrix factorization strategies applied to text categorization, e.g., [59], are shown to increase the performance of SVM classifiers, which we exploit: the key idea involves the construction of latent feature spaces, thus being closely related to Latent Semantic Indexing [17] techniques. While this line of work differs from ours, we believe it could be useful to optimize the features we use in the supervised classification setting.

12.4.3. Semantic Role Labeling

In broad terms, the Semantic Role Labeling (SRL) NLP task targets the identification of arguments attached to a given predicate in natural language utterances. From a Frame Semantics perspective, such activity translates into the assignment of FEs. This applies to efforts such as [35], and tools like MATE [10], while we perform full frame classification. On the other hand, systems like SEMAFOR [36,15] also serve the frame disambiguation part, uniformly to our method. Hence, SEMAFOR could be regarded as a baseline system. Nonetheless, it was not possible to actually perform a comparative evaluation of our use case in Italian, since the parser exclusively supports the English language.

All the work mentioned above (and SRL in general) builds upon preceding layers of NLP machinery, i.e., POS-tagging and syntactic parsing: the importance of the latter is especially stressed in [50], thus being in strong contrast to our approach, where we propose a full bypass of the expensive syntactic step.

13. Conclusion

In a Web where the profusion of unstructured data limits its automatic interpretation, the necessity of *Intelligent Web-reading Agents* turns more and more evident. These agents should preferably be conceived to browse an extensive and variegated amount of Web sources corpora, harvest structured assertions out of them, and finally cater for target KBs, which can attenuate the problem of information overload. As a support to such vision, we have outlined two real-world scenarios involving general-purpose KBs:

(a) WIKIDATA would benefit from a system that reads reliable third-party resources, extracts statements complying to the KB data model, and leverages them to validate existing data with reference URLs, or to recommend new items for inclusion. This would both improve the overall data quality and, most importantly, underpin the costly manual data insertion and curation flow;

(b) DBPEDIA would naturally evolve towards the extraction of unstructured Wikipedia content. Since Wikidata is designed to be the hub for serving structured data across Wikimedia projects, it will let DBpedia focus on content besides infoboxes, categories and links.

In this article, we presented a system that puts into practice our fourfold research contribution: first, we perform (1) *N-ary relation extraction* thanks to the implementation of Frame Semantics, in contrast to traditional binary approaches; second, we (2) *simultaneously enrich both the T-Box and the A-Box* parts of our target KB, through the discovery of candidate relations and the extraction of facts respectively. We achieve this with a (3) *shallow layer of NLP* technology, namely grammatical analysis, instead of more sophisticated ones, such as syntactic parsing. Finally, we ensure a (4) *fully supervised* learning paradigm via an affordable *crowdsourcing* methodology.

Our work concurrently bears the advantages and leaves out the weaknesses of RE and OIE: although we assess it in a closed-domain fashion via a use case (Section 2), the corpus analysis module (Section 4) allows to discover an exhaustive set of relations in an open-domain way. In addition, we overcome the supervision cost bottleneck trough crowdsourcing. Therefore, we believe our approach can represent a trade-off between open-domain high noise and closed-domain high cost.

The FACT EXTRACTOR is a full-fledged Information Extraction NLP pipeline that analyses a natural language textual corpus and generates structured machine-readable assertions. Such assertions are disambiguated by linking text fragments to entity URIs of the target KB, namely DBpedia, and are assigned a confidence score. For instance, given the sentence `Buffon plays for Serie A club Juventus since 2001`, our system produces the following dataset:

```
@prefix dbpedia: <http://it.dbpedia.org/resource/> .
@prefix dbpo: <http://dbpedia.org/ontology/> .
@prefix fact: <http://fact.extraction.org/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

dbpedia:Gianluigi_Buffon
   dbpo:careerStation dbpedia:CareerStation_01 .

dbpedia:CareerStation_01
   dbpo:team dbpedia:Juventus_Football_Club ;
   fact:competition dbpedia:Serie_A ;
   dbpo:startYear "2001"^^xsd:gYear ;
   fact:confidence "0.906549"^^xsd:float .
```

We estimate the validity of our approach by means of a use case in a specific domain and language, i.e., soccer and Italian. Out of roughly $52,000$ Italian Wikipedia articles describing soccer players, we output more than $213,000$ triples with an estimated average 81.27% F_1. Since our focus is the improvement of existing resources rather than the development of a standalone one, we integrated these results into the ITALIAN DBPEDIA CHAPTER[35] and made them accessible through its SPARQL endpoint. Moreover, the codebase is publicly available as part of the DBPEDIA ASSOCIATION repository.[36]

We have started to expand our approach under the Wikidata umbrella, where we feed the *primary sources* tool. The community is currently concerned by the trustworthiness of Wikidata assertions: in order to authenticate them, they should be validated against references to external Web sources. Under this perspective, we are leading the STREPHIT Wikimedia IEG project[37] builds upon the FACT EXTRACTOR and aims at serving as a reference suggestion mechanism for statement validation. To achieve this, we have successfully managed to switch the input corpus from Wikipedia to third-party corpora and translated our output to fit the Wikidata data model. The soccer use case has already been partially implemented: we have ran the baseline classifier and generated a small demonstrative dataset, named FBK-STREPHIT-SOCCER, which has been uploaded to the primary sources tool back-end. We invite the reader to play with it, by following the instructions in the project page.[38] At the time of writing this article, we are scaling up to (a) a larger input in (b) the English language, with (c) a bigger set of relations, and (d) a different domain. The *Web Sources* corpus contains more than $500,000$ English documents gathered from 53 sources; the corpus analysis yielded 50 relations, which are connected to an already available frame repository, i.e., FrameNet.

For future work, we foresee to progress towards multilingual open information extraction, thus paving the way to (a) its full deployment into the DBpedia Extraction Framework, and to (b) a thorough referencing system for Wikidata.

Acknowledgments. The FACT EXTRACTOR has been developed within the DBPEDIA ASSOCIATION and was partially funded by GOOGLE under the SUMMER OF CODE 2015 program. The STREPHIT project is undergoing active development and is fully funded by the WIKIMEDIA FOUNDATION via the INDIVIDUAL ENGAGEMENT GRANTS program.

References

[1] Eneko Agirre, Enrique Alfonseca, Keith B. Hall, Jana Kravalova, Marius Pasca, and Aitor Soroa. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, May 31 - June 5, 2009, Boulder, Colorado, USA*, pages 19–27. The Association for Computational Linguistics, 2009.

[2] Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D. Manning. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 344–354. The Association for Computer Linguistics, 2015.

[3] Gabor Angeli, Julie Tibshirani, Jean Wu, and Christopher D. Manning. Combining distant and partial supervision for relation extraction. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP*

[35] `http://it.dbpedia.org/2015/09/meno-chiacchiere-piu-fatti-una-marea-di-nuovi-dati-estratti-dal-testo-di-wikipedia/?lang=en`

[36] `https://github.com/dbpedia/fact-extractor`

[37] `https://meta.wikimedia.org/wiki/Grants:IEG/StrepHit:_Wikidata_Statements_Validation_via_References`

[38] `https://www.wikidata.org/wiki/Wikidata:Primary_sources_tool#How_to_use`

2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 1556–1567. ACL, 2014.

[4] Isabelle Augenstein, Diana Maynard, and Fabio Ciravegna. Relation extraction from the web using distant supervision. In Krzysztof Janowicz, Stefan Schlobach, Patrick Lambrix, and Eero Hyvönen, editors, *Knowledge Engineering and Knowledge Management - 19th International Conference, EKAW 2014, Linköping, Sweden, November 24-28, 2014. Proceedings*, volume 8876 of *Lecture Notes in Computer Science*, pages 26–41. Springer, 2014.

[5] Collin F. Baker. Framenet, current collaborations and future goals. *Language Resources and Evaluation*, 46(2):269–286, 2012.

[6] Collin F Baker. Framenet: A knowledge base for natural language processing. *ACL 2014*, 1929:1–5, 2014.

[7] Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The berkeley framenet project. In Christian Boitet and Pete Whitelock, editors, *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, COLING-ACL '98, August 10-14, 1998, Université de Montréal, Montréal, Quebec, Canada. Proceedings of the Conference.*, pages 86–90. Morgan Kaufmann Publishers / ACL, 1998.

[8] L. Douglas Baker and Andrew Kachites McCallum. Distributional clustering of words for text classification. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 96–103. The Association for Computing Machinery, 1998.

[9] Jonathan Berant and Percy Liang. Semantic parsing via paraphrasing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 1415–1425. The Association for Computer Linguistics, 2014.

[10] Anders Björkelund, Love Hafdell, and Pierre Nugues. Multilingual semantic role labeling. In Jan Hajic, editor, *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task, CoNLL 2009, Boulder, Colorado, USA, June 4, 2009*, pages 43–48. ACL, 2009.

[11] Kurt D. Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In Jason Tsong-Li Wang, editor, *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, pages 1247–1250. ACM, 2008.

[12] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell. Toward an architecture for never-ending language learning. In Maria Fox and David Poole, editors, *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010.* AAAI Press, 2010.

[13] Nancy Chang, Praveen Paritosh, David Huynh, and Collin Baker. Scaling semantic frame annotation. In *Proceedings of The 9th Linguistic Annotation Workshop LAW IX, June 5, 2015, Denver, Colorado, USA*, pages 1–10. ACL, 2015.

[14] Bonaventura Coppola, Aldo Gangemi, Alfio Massimiliano Gliozzo, Davide Picca, and Valentina Presutti. Frame detection over the semantic web. In Lora Aroyo, Paolo Traverso, Fabio Ciravegna, Philipp Cimiano, Tom Heath, Eero Hyvönen, Riichiro Mizoguchi, Eyal Oren, Marta Sabou, and Elena Paslaru Bontas Simperl, editors, *The Semantic Web: Research and Applications, 6th European Semantic Web Conference, ESWC 2009, Heraklion, Crete, Greece, May 31-June 4, 2009, Proceedings*, volume 5554 of *Lecture Notes in Computer Science*, pages 126–142. Springer, 2009.

[15] Dipanjan Das, Desai Chen, André F. T. Martins, Nathan Schneider, and Noah A. Smith. Frame-semantic parsing. *Computational Linguistics*, 40(1):9–56, 2014.

[16] Filipe de Sá Mesquita, Jordan Schmidek, and Denilson Barbosa. Effectiveness and efficiency of open relation extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP 2013, 18-21 October 2013, Grand Hyatt Seattle, Seattle, Washington, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 447–457. ACL, 2013.

[17] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *JASIS*, 41(6):391–407, 1990.

[18] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge vault: a web-scale approach to probabilistic knowledge fusion. In Sofus A. Macskassy, Claudia Perlich, Jure Leskovec, Wei Wang, and Rayid Ghani, editors, *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 601–610. ACM, 2014.

[19] Arnab Dutta, Christian Meilicke, and Heiner Stuckenschmidt. Enriching structured knowledge with open information. In *Proceedings of the 24th International Conference on World Wide Web, WWW 2015, Florence, Italy, May 18-22, 2015*, pages 267–277, 2015.

[20] Fredo Erxleben, Michael Günther, Markus Krötzsch, Julian Mendez, and Denny Vrandecic. Introducing wikidata to the linked data web. In *The Semantic Web - ISWC 2014 - 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I*, pages 50–65, 2014.

[21] Anthony Fader, Stephen Soderland, and Oren Etzioni. Identifying relations for open information extraction. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1535–1545. ACL, 2011.

[22] Manaal Faruqui and Shankar Kumar. Multilingual open relation extraction using cross-lingual projection. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 1351–1356, 2015.

[23] Charles Fillmore. Frame semantics. *Linguistics in the morning calm*, pages 111–137, 1982.

[24] Charles J. Fillmore. Frame Semantics and the nature of language. In *Annals of the New York Academy of Sciences: Conference on the Origin and Development of Language*, pages 20–32. Blackwell Publishing, 1976.

[25] Marco Fossati, Claudio Giuliano, and Sara Tonelli. Outsourcing framenet to the crowd. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 2: Short Papers*, pages 742–747. The Association for Computer Linguistics, 2013.

[26] Aldo Gangemi and Valentina Presutti. Towards a pattern science for the semantic web. *Semantic Web*, 1(1-2):61–68, 2010.

[27] Aldo Gangemi, Valentina Presutti, Diego Reforgiato Recupero, Andrea Giovanni Nuzzolese, Francesco Draicchio, and Misael Mongiovìb. Semantic web machine reading with fred. *Semantic Web*, 2016. Under Review.

[28] Claudio Giuliano, Alfio Massimiliano Gliozzo, and Carlo Strapparava. Kernel methods for minimally supervised WSD. *Computational Linguistics*, 35(4):513–528, 2009.

[29] Sebastian Hellmann, Jens Lehmann, Sören Auer, and Martin Brümmer. Integrating NLP using linked data. In *The Semantic Web - ISWC 2013 - 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II*, pages 98–113, 2013.

[30] Daniel Hernández, Aidan Hogan, and Markus Krötzsch. Reifying RDF: what works well with wikidata? In *Proceedings of the 11th International Workshop on Scalable Semantic Web Knowledge Base Systems co-located with 14th International Semantic Web Conference (ISWC 2015), Bethlehem, PA, USA, October 11, 2015.*, pages 32–47, 2015.

[31] Rinke Hoekstra. *Ontology Representation - Design Patterns and Ontologies that Make Sense*, volume 197 of *Frontiers in Artificial Intelligence and Applications*. IOS Press, 2009.

[32] Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, and Gerhard Weikum. YAGO2: A spatially and temporally enhanced knowledge base from wikipedia. *Artif. Intell.*, 194:28–61, 2013.

[33] Jisup Hong and Collin F Baker. How Good is the Crowd at "real" WSD? In *Proceedings of the 5th Linguistic Annotation Workshop*, pages 30–37, 2011.

[34] Richard Johansson and Pierre Nugues. Lth: Semantic structure extraction using nonprojective dependency trees. In *Proceedings of the 4th international workshop on semantic evaluations*, pages 227–230. The Association for Computational Linguistics, 2007.

[35] Richard Johansson and Pierre Nugues. Dependency-based semantic role labeling of propbank. In *2008 Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, Proceedings of the Conference, 25-27 October 2008, Honolulu, Hawaii, USA, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 69–78. ACL, 2008.

[36] Meghana Kshirsagar, Sam Thomson, Nathan Schneider, Jaime G. Carbonell, Noah A. Smith, and Chris Dyer. Frame-semantic role labeling with heterogeneous annotations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers*, pages 218–224. The Association for Computational Linguistics, 2015.

[37] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. DBpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, 6(2):167–195, 2015.

[38] Lluís Màrquez, Xavier Carreras, Kenneth C. Litkowski, and Suzanne Stevenson. Semantic role labeling: An introduction to the special issue. *Computational Linguistics*, 34(2):145–159, 2008.

[39] Mausam, Michael Schmitz, Stephen Soderland, Robert Bart, and Oren Etzioni. Open language learning for information extraction. In Jun'ichi Tsujii, James Henderson, and Marius Pasca, editors, *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*, pages 523–534. ACL, 2012.

[40] Pablo N. Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. Dbpedia spotlight: shedding light on the web of documents. In Chiara Ghidini, Axel-Cyrille Ngonga Ngomo, Stefanie N. Lindstaedt, and Tassilo Pellegrini, editors, *Proceedings the 7th International Conference on Semantic Systems, I-SEMANTICS 2011, Graz, Austria, September 7-9, 2011*, ACM International Conference Proceeding Series, pages 1–8. ACM, 2011.

[41] Vinh Nguyen, Olivier Bodenreider, and Amit Sheth. Don't like rdf reification?: making statements about statements using singleton property. In *Proceedings of the 23rd international conference on World wide web*, pages 759–770. ACM, 2014.

[42] Andrea Giovanni Nuzzolese, Aldo Gangemi, and Valentina Presutti. Gathering lexical linked data and knowledge patterns from framenet. In Mark A. Musen and Óscar Corcho, editors, *Proceedings of the 6th International Conference on Knowledge Capture (K-CAP 2011), June 26-29, 2011, Banff, Alberta, Canada*, pages 41–48. ACM, 2011.

[43] Andrea Giovanni Nuzzolese, Aldo Gangemi, Valentina Presutti, and Paolo Ciancarini. Fine-tuning triplification with semion. In *EKAW workshop on Knowledge Injection into and Extraction from Linked Data (KIELD2010)*, pages 2–14, 2010.

[44] Fernando C. N. Pereira, Naftali Tishby, and Lillian Lee. Distributional clustering of english words. *CoRR*, abs/cmp-lg/9408011, 1994.

[45] Silvio Peroni, Aldo Gangemi, and Fabio Vitali. Dealing with markup semantics. In *Proceedings the 7th International Conference on Semantic Systems, I-SEMANTICS 2011, Graz, Austria, September 7-9, 2011*, pages 111–118, 2011.

[46] Emanuele Pianta, Christian Girardi, and Roberto Zanoli. The textpro tool suite. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2008, 26 May - 1 June 2008, Marrakech, Morocco*, 2008.

[47] Valentina Presutti, Sergio Consoli, Andrea Giovanni Nuzzolese, Diego Reforgiato Recupero, Aldo Gangemi, Ines Bannour, and Haïfa Zargayouna. Uncovering the semantics of wikipedia pagelinks. In Krzysztof Janowicz, Stefan Schlobach, Patrick Lambrix, and Eero Hyvönen, editors, *Knowledge Engineering and Knowledge Management - 19th International Conference, EKAW 2014, Linköping, Sweden, November 24-28, 2014. Proceedings*, volume 8876 of *Lecture Notes in Computer Science*, pages 413–428. Springer, 2014.

[48] Valentina Presutti, Francesco Draicchio, and Aldo Gangemi. Knowledge extraction based on discourse representation theory and linguistic frames. In Annette ten Teije, Johanna Völker, Siegfried Handschuh, Heiner Stuckenschmidt, Mathieu d'Aquin, Andriy Nikolov, Nathalie Aussenac-Gilles, and Nathalie Hernandez, editors, *Knowledge Engineering and Knowledge Management - 18th International Conference, EKAW 2012, Galway City, Ireland, October 8-12, 2012. Proceedings*, volume 7603 of *Lecture Notes in Computer Science*, pages 114–129. Springer, 2012.

[49] Valentina Presutti, Andrea Giovanni Nuzzolese, Sergio Consoli, Diego Reforgiato Recupero, and Aldo Gangemi. From hyperlinks to semantic web properties using open knowledge extraction. *Semantic Web*, Preprint(Preprint):1–28, 2016.

[50] Vasin Punyakanok, Dan Roth, and Wen-tau Yih. The importance of syntactic parsing and inference in semantic role labeling.

Computational Linguistics, 34(2):257–287, 2008.

[51] Jacobo Rouces, Gerard de Melo, and Katja Hose. Framebase: Representing n-ary relations using semantic frames. In Fabien Gandon, Marta Sabou, Harald Sack, Claudia d'Amato, Philippe Cudré-Mauroux, and Antoine Zimmermann, editors, *The Semantic Web. Latest Advances and New Domains - 12th European Semantic Web Conference, ESWC 2015, Portoroz, Slovenia, May 31 - June 4, 2015. Proceedings*, volume 9088 of *Lecture Notes in Computer Science*, pages 505–521. Springer, 2015.

[52] Jacobo Rouces, Gerard de Melo, and Katja Hose. Integrating heterogeneous knowledge with framebase. *Semantic Web*, 2016. Under Review.

[53] Thomas Schmidt. The kicktionary revisited. In Angelika Storrer, Alexander Geyken, Alexander Siebert, and Kay-Michael Würzner, editors, *Text Resources and Lexical Knowledge. Selected Papers from the 9th Conference on Natural Language Processing, KONVENS 2008, Berlin, Germany*, pages 239–251. Mouton de Gruyter, 2008.

[54] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 254–263. Association for Computational Linguistics, 2008.

[55] Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. Multi-instance multi-label learning for relation extraction. In Jun'ichi Tsujii, James Henderson, and Marius Pasca, editors, *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL 2012, July 12-14, 2012, Jeju Island, Korea*, pages 455–465. ACL, 2012.

[56] Denny Vrandecic and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, 2014.

[57] Fei Wu and Daniel S. Weld. Open information extraction using wikipedia. In Jan Hajic, Sandra Carberry, and Stephen Clark, editors, *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden*, pages 118–127. The Association for Computer Linguistics, 2010.

[58] Xuchen Yao and Benjamin Van Durme. Information extraction over structured data: Question answering with freebase. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 956–966. The Association for Computer Linguistics, 2014.

[59] Shenghuo Zhu, Kai Yu, Yun Chi, and Yihong Gong. Combining content and link for classification using matrix factorization. In Wessel Kraaij, Arjen P. de Vries, Charles L. A. Clarke, Norbert Fuhr, and Noriko Kando, editors, *SIGIR 2007: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, July 23-27, 2007*, pages 487–494. ACM, 2007.