Progress in Artificial Intelligence (2021) 10:309–320
https://doi.org/10.1007/s13748-021-00241-7

REGULAR PAPER

Semantic role labeling for knowledge graph extraction from text

Mehwish Alam[1] · Aldo Gangemi[2,3] · Valentina Presutti[2,3] · Diego Reforgiato Recupero[4]

Received: 16 July 2019 / Accepted: 15 March 2021 / Published online: 5 April 2021

Abstract

This paper introduces TakeFive, a new semantic role labeling method that transforms a text into a frame-oriented knowledge graph. It performs dependency parsing, identifies the words that evoke lexical frames, locates the roles and fillers for each frame, runs coercion techniques, and formalizes the results as a knowledge graph. This formal representation complies with the frame semantics used in Framester, a factual-linguistic linked data resource. We tested our method on the WSJ section of the Peen Treebank annotated with VerbNet and PropBank labels and on the Brown corpus. The evaluation has been performed according to the CoNLL Shared Task on Joint Parsing of Syntactic and Semantic Dependencies. The obtained precision, recall, and F1 values indicate that TakeFive is competitive with other existing methods such as SEMAFOR, Pikes, PathLSTM, and FRED. We finally discuss how to combine TakeFive and FRED, obtaining higher values of precision, recall, and F1 measure.

Keywords Semantic role labeling · Frame semantics · Framester · Dependency parsing · Role oriented knowledge graphs

1 Introduction

Most knowledge in linked data and knowledge graphs is of relational nature: people participating in events, products having prices, artifacts with parts, works of art produced by artists, beers sold at a bar, etc. For that reason, a good part of integration and interoperability ends up consisting of aligning relations among heterogeneous schemas and data.

Less known is the fact that the relations holding between entities are usually part of a larger context or situation: beers can be found at a bar because there is a selling/purchase situation; artists produce works because there is a creative process involved; artifacts are assembled through craftsmanship or industrial procedures; products are assigned prices in the market; people are assigned roles in events, etc.

Regardless of the representation language used and its serialization, existing knowledge graphs share a common limit. For example, two of the most important linguistic resources which are part of the Linked Open Data are WordNet[1] and FrameNet[2]. They have been formalized in OntoWordNet [1], WordNet RDF [2], FrameNet DAML [3], FrameNet RDF [4], etc. The limited coverage of FrameNet reduces its usability and, to create a wide coverage (including contextual and situational information) and multilingual extensions, a solution would be to create valid links between FrameNet and other lexical resources such as WordNet, VerbNet[3], and BabelNet[4]. They express facts that typically lack contextual and situational information. This limit makes interoperability difficult because when two different datasets need to be integrated, implicit situations need to be reconstructed. This happens quite smoothly in humans, but not in knowledge-based systems. A possible solution would be to enrich linked data with contextual and situational knowledge, for example from FrameNet. A method to contextualize knowledge graphs is to express the facts that they capture as

✉ Diego Reforgiato Recupero
diego.reforgiato@unica.it

Mehwish Alam
mehwish.alam@fiz-karlsruhe.de

Aldo Gangemi
aldo.gangemi@istc.cnr.it

Valentina Presutti
valentina.presutti@istc.cnr.it

[1] FIZ-Karlsruhe Leibniz Institute for Information Infrastructure, Karlsruhe, Germany

[2] ISTC, CNR, Rome, Italy

[3] University of Bologna, Bologna, Italy

[4] Università degli Studi di Cagliari, Cagliari, Italy

[1] https://wordnet.princeton.edu/.

[2] https://framenet.icsi.berkeley.edu/fndrupal/.

[3] https://verbs.colorado.edu/verbnet/.

[4] https://babelnet.org/.

projections of frames. *Frames* are cognitive structures that are used by humans for organizing their knowledge, as well as for interpreting, processing, or anticipating information (cf. [5] for a discussion encompassing both linguistic and knowledge-based approaches to frames). In linguistics, a reference model for frames is Fillmore's Frame Semantics [6], where a frame is introduced intuitively as "a kind of outline figure with not necessarily all of the details filled in". More precisely, a frame is a structure that reifies an n-ary relation with multi-varied arguments, denotes a situation, event, state, or configuration, and is supposed to bear representational similarity to the knowledge encoded in cognitive systems. Any binary projection of a frame is called a *semantic role*. For example, in the sentence *I bought a pair of shoes*, the word "bought" identifies an occurrence of a commercial event, where "I" and "pair of shoes" are objects that play the roles of "buyer" and "'goods", respectively, in the `Commerce_buy` frame. Fillmore's Frame Semantics has been substantiated by FrameNet [7]: a long-standing, manually developed resource of (English) frames represented in a structured format by a group of linguists in Berkeley.

Recently, two resources have been introduced which support semantic interoperability by using frames: FrameBase [8] and Framester [9]. The idea is simple in principle: since situations are frame occurrences, let us align any schema to a set of frames from a stable ontology, and make data interoperate along that path (if a schema fragment 1 and a schema fragment 2 align to the same frame, the respective data can be jointly queried modulo ontology-based data access, where the ontology of frames is uniform across resources). This is apparently good news, but while an initial ontology of frames can be found in FrameNet, the methods through which we can actually align any existing schema to frames are much less obvious. And the main reason is that the relations defined in schemas, ontologies, and knowledge graphs cannot be trivially aligned to semantic roles.

For example, we need to assign the relation `foaf:knows` to a semantic role within frames such as `framenet:Personal_relationship` or `framenet:Familiarity`. However, there is no semantic role corresponding to `foaf:knows`. The alignment would work only as a result of a path internal to the frame, e.g., an OWL property chain on roles, such as `isPartnerIn` or `hasPartner`.

Current approaches are still struggling with this problem: FrameBase manually aligns relations to semantic roles, leading to scalability issues, while Framester provides an extensive amount of linguistic mappings that help a semi-automated alignment, but a previous linguistic parsing of the relations and their context is required, which is still non-standard, especially considering that only a few ontologies explicitly encode the competency questions that led to the form of their relations. In practice, the integration of a knowledge extraction approach from competency questions or other textual material, its alignment to Framester, and the usage of dereferencing methods as proposed by FrameBase collectively seem a viable automated integration solution in the future.

In order to foster the research, in this paper we propose a knowledge-graph-based algorithm for labeling semantic roles from an arbitrary text, thus accommodating the linguistic parsing needed to perform frame alignment prior to interoperability. It works by combining and verifying syntactical information extracted with NLP tools (e.g., CoreNLP) with semantic information extracted with Framester, FrameNet, and VerbNet. It is based on the two following steps:

1. given an input sentence preprocess it to identify and extract syntactic and semantic information;
2. detect syntactic CoreNLP-derived roles, semantic VerbNet-based roles for a certain frame and check the compatibility between the syntactic and semantic roles.

Our algorithm, called TakeFive, is evaluated, and compared to alternative approaches, with respect to metrics widely applied in two NLP tasks: *frame detection* and *semantic role labeling* (SRL). The first refers to the ability to automatically detect occurrences of frames in natural language text. The second refers to identifying the fragments of text denoting the entities that play specific roles in a frame occurrence. In this paper, we extend our preliminary work [10] and use Framester [9], a frame-based knowledge graph, to address frame detection and SRL with TakeFive[5]. Note that Framester has already been successfully applied within the Sentiment Analysis domain [11,12]. TakeFive uses NLP resources and software components but integrates them in a semantic web pipeline that produces knowledge graphs ready to be used for interoperability across data and schemas. We intend to verify if a knowledge-based hybrid method is comparable to purely statistical methods while retaining the ability to extract a properly linked knowledge graph from a sentence. As an example of the output of SRL, let us consider the following sentence:

Despite recent declines in yields,
investors continue to pour cash into money funds. (1)

By performing frame detection, we recognize that *to pour* evokes the frame `Pour.v` from Framester, subsumed by the frame `Cause_motion` from FrameNet, meaning that the sentence expresses an occurrence of this frame. By performing SRL, TakeFive, through the two steps mentioned above, then labels *investors* and *cash*, respectively, with the `Agent.cause_motion` role, and the `Theme.cause_`

[5] https://github.com/TakeFiveSRL/TakeFiveSRL.

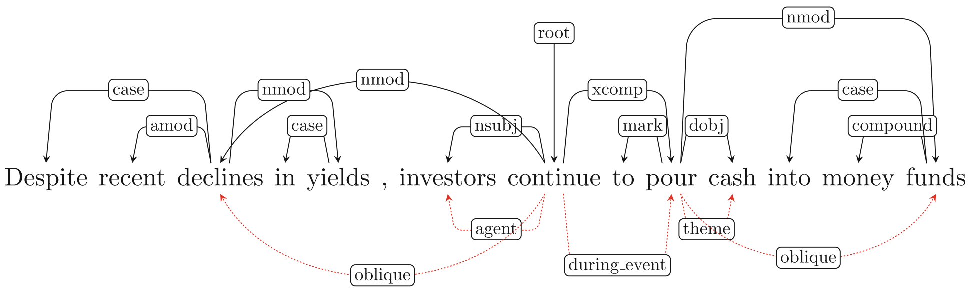

Fig. 1 Dependency tree obtained from CoreNLP. Dotted lines represent the role labeling by TakeFive

motion role, as both involved in the Cause_motion situation occurrence (cf. Fig. 1). The annotations use entities from reference ontologies for frames and semantic roles.

TakeFive has been compared to state-of-the-art SRL tools including SEMAFOR, Pikes, PathLSTM, and FRED, by using the same evaluation process from the CoNLL Shared Task on Joint Parsing of Syntactic and Semantic Dependencies [13,14]. Some of them (e.g., FRED and Pikes) are knowledge graph extractors employed to make sense out of text documents.

To sum up, the contributions of our paper are the following:

- We employ Framester by running queries on its knowledge graph to return verb senses, semantic frames, and VerbNet roles.
- We combine the output of Framester with CoreNLP to come up with TakeFive, an approach to address frame detection and semantic role labeling;
- We compare TakeFive against state-of-the art SRL tools (SEMAFOR, Pikes, PathLSTM, FRED) and show that TakeFive outperforms the competitors on reference datasets;
- The code of TakeFive, Framester, and all the other used resources are open-source and publicly available for download.

The remainder of the paper is organized as follows: Sect. 2 describes related work for SRL and SRL-based knowledge extraction. Section 3 briefly introduces the data, resources, and components used by TakeFive. Section 4 presents the TakeFive algorithm. Section 5 details the evaluation settings, the performance measures, and the comparative results for frame detection and semantic role labeling. Finally, Sect. 6 ends the paper with conclusions and future directions where we are headed.

2 Related works

Semantic technologies usually leverage syntactic resources to improve their accuracy. Stanford CoreNLP[6] [15], which we use in TakeFive, is one of the most used full-fledged NLP tools and has been used in the Semantic Web context. However, it has not been extensively employed for SRL (an exception is [16]).

After the development of PropBank [17], where semantic information has been added to the Penn English Treebank data set, and the CoNLL shared tasks on semantic role labeling [18,19], there has been a lot of research in this domain, typically using PropBank as the reference ontology for roles. PropBank is a data set consisting of the phrase-structure syntactic trees from the Wall Street Journal (WSJ) section of the Penn Treebank. Its annotations include predicate-argument structures for verbs and define a small number of roles: core roles are ARG0 through ARG5, which can be interpreted differently for different predicates. Further modifier roles ARGM* include, e.g., ARGM-TMP (temporal) and ARGM-DIR (directional).

The semantics of the core roles ARG0-ARG5 is not straightforwardly clear. The study described in [20] shows that the roles ARG2-ARG5 serve many different purposes for different verbs, and points out that they are inconsistent and highly overloaded. To improve the performance for the SRL task, the arguments were mapped to VerbNet thematic roles. Others, e.g., [21], revised the syntactic subcategorization patterns for FrameNet lexical units, using VerbNet. While PropBank labels the roles of verbs with a limited number of tags, frame-semantic parsing labels frame arguments with frame-specific roles, making it clearer what those arguments may mean. Therefore, for frame-semantic parsing, sentences may contain multiple frames that need to be detected along with their arguments. SemEval 2007 task 19 [22] addressed this problem. The task leveraged FrameNet 1.3 and released a small corpus containing more than 2000 sentences with full text annotations.

The work described in [23] projects predicate-argument structures from seed examples to unlabeled sentences using linear program formulation to find the best alignment related to the projection. The projected information and the seeds are both used to train statistical models for SRL. Besides, the authors introduce a method for finding examples for unseen verbs using a graph alignment tool, which was used to project annotations from seed examples to unlabeled sentences.

[6] https://stanfordnlp.github.io/CoreNLP/.

In [24,25] the authors use an unsupervised approach for SRL that aims at inducing semantic roles automatically from unannotated data. Although this can be useful to discover new semantic frames and roles, in this paper, we focus on the concrete representation provided in FrameNet and VerbNet, without expanding their inventory of semantic types.

Authors in [26] introduce a semantic parser that uses a broad knowledge base created by interconnecting FrameNet, VerbNet, and PropBank. SEMAFOR [27][7] is a well-known system for frame-semantic parsing, based on the combination of knowledge from FrameNet, two probabilistic models trained on full-text annotations released along with the FrameNet lexicon, and expedient heuristics. At SemEval 2007, it outperformed existing approaches.

FRED[8] [28,29] is the state-of-the-art tool for producing framed knowledge graphs for the Semantic Web. It consists of a complex pipeline of NLP and Semantic Web components for parsing text, representing it to a neo-Davidsonian logical form, extracting entities, disambiguating predicates, linking them to public resources, and creating a well-connected, formal and queryable knowledge graph out of that. FRED uses a "greedy" approach for SRL, i.e., it labels roles with reference labels (from either VerbNet or FrameNet) when the confidence of its categorical parser is high; otherwise, it uses other heuristics to provide meaningful local labels that make sense in that textual context. However, the current study majorly focuses on improving the SRL approach using roles from Framester and then using this for generating KGs. FRED can further be extended with the proposed approach for better SRL.

PIKES [30][9] is another knowledge graph extractor that automatically extracts things of interest and facts about them from the text. It applies a number of NLP tools to annotate a text and leverages a linked-data-oriented approach to generate RDF graphs.

PathLSTM is a SRL system introduced in [31], which builds on top of the mate-tools semantic role labeler[10]. It leverages neural sequence modeling techniques: the authors model semantic relationships between a predicate and its arguments by analyzing the dependency path between a predicate word, and each argument headword. Lexicalized paths are considered, which are decomposed into sequences of individual items, namely the words and dependency relations on a path. Long short-term memory networks are then applied to find a recurrent composition function that can reconstruct an appropriate representation of the full path from its individual parts.

Recently authors have employed a deep neural network architecture known as Positional Attention-based Frame Identification with BERT (PAFIBERT) [32] as a solution to the frame identification subtask in frame-semantic parsing. Their method combines the language representation power of BERT [33] with a position-based attention mechanism to disambiguate targets and associate them with the most suitable semantic frames. The difference with our approach is that theirs is limited to frame identification and yet does not extract or link semantic roles.

3 Material

In this section, we describe the lexical resources that we have employed for the design of TakeFive.

3.1 VerbNet

VerbNet [34] is a broad coverage verb lexicon in English, with links to other data sources such as WordNet [35] and FrameNet [7]. It contains semantic roles and verb classes corresponding to Levin's classes [36] and including multiple verb senses. Verb classes can therefore be considered akin to word synsets. They generalize the verbs based on their shared syntactic behavior. These verb classes feature a simple two-layer hierarchy. For example, the verb *conquer* is a member of the class `subjugate-42.3`, and hence, a sense `Conquer_42030000` is created (the sense of conquer in that class).

VerbNet further contains semantic roles, which correspond to the relations between a verb sense and its arguments. Each class has multiple *frames* (either syntactic- or semantic-oriented), which define a list of *predicates* associated with their *arguments*. There is a (partial) morphism between syntactic and semantic frames, so that semantic roles ("arguments") are also associated with patterns that characterize the syntactic behavior of a verb in that class. For example, the roles defined for the class *subjugate-42.3* are *Agent*, *Patient* and *Instrument* meaning that an agent subjugates a patient with some instrument. Here, *Agent* and *Patient* are necessary roles, and *Instrument* is an optional role. Verb senses help in determining if a particular verb instance conforms to the underlying semantics of the class. For the case of the verb *conquer*, its only sense is included in the class `subjugate-42.3`. VerbNet maps verbs to FrameNet frames, e.g., the verb sense `Conquer_42030000` is mapped to the frame `Conquering`. The version of VerbNet used in TakeFive evaluation is 3.1, and the data come from the RDF porting of VerbNet 3.1 that is included in Framester[11] [9].

[7] http://www.ark.cs.cmu.edu/SEMAFOR.

[8] http://wit.istc.cnr.it/stlab-tools/fred.

[9] http://pikes.fbk.eu.

[10] http://code.google.com/p/mate-tools/.

[11] http://github.com/framester/Framester.

3.2 FrameNet

FrameNet [7] contains *frames*, which describe a situation, state or action. Each frame has semantic roles ("frame elements") that are much more semantically detailed than VerbNet ones. FrameNet also defines a subsumption relation between either frames or roles. The subsumption relation can be used to create a hierarchy of classes, as shown in [37]. Each frame can be evoked by *lexical Units (LUs)* belonging to different parts of speech. In version 1.5, FrameNet covers about 10,000 lexical units and 1024 frames. Let us consider the following sentence:

$$[\textit{The Spaniards}]_{\textit{Conqueror}}\ [\textit{conquered}]_{\textit{Lexical Unit}}\ [\textit{the Incas}]_{\textit{Theme}}. \qquad (2)$$

In the above example, *The Spaniards* is the argument (we will also refer to it as filler) of the role *Conqueror* and *the Incas* is the argument (or filler) of the role *Theme* and *conquered* is the lexical unit evoking the frame.

3.3 Framester

Framester [9] is a large RDF[12] knowledge graph (currently including about 50 million triples), acting as a hub between several predicate oriented linguistic resources such as FrameNet, WordNet, VerbNet [34], BabelNet [38], Predicate Matrix [39], as well as many other linguistic, factual, and foundational knowledge graphs. It leverages this wealth of links to create an interoperable and homogeneous *predicate space* represented in a formal rendering of frame semantics [6] and semiotics [40]. Framester uses a novel mapping between WordNet, BabelNet, VerbNet, and FrameNet at its core, expands it to other linguistic resources transitively, and represents all of this formally. It further links these resources to other important ontological and linked data resources such as DBpedia [41], YAGO [42], DOLCE-Zero [43], schema.org, [44], NELL [45], etc.

Framester is accessible through its SPARQL endpoint[13]. Framester also features a subsumption hierarchy of semantic roles (i.e., frame elements) and adds generic roles on top of frame-specific roles.

Framester also offers a Word Frame Disambiguation (WFD) service based on the mappings defined within the resource. It is available as a frame detection API. First, it employs the word sense disambiguation algorithms UKB[14] and Babelfy[15]. Then, it uses the mappings between WordNet/BabelNet synsets and FrameNet frames. The associated REST API is available online[16].

4 Semantic role labeling algorithm

TakeFive extracts information from Framester by leveraging its interoperability. To do that we exploited the rigorous formal treatment for Fillmore's frame semantics we designed in Framester. That is what makes Framester acting as an exceptional hub between FrameNet, WordNet, VerbNet, BabelNet, DBpedia, Yago, DOLCE-Zero, as well as other resources. The information extracted with TakeFive is different than those provided by others: the effective interoperability of Framester is the glue of all the included knowledge bases and this allows solving different tasks, i.e., the semantic role labeling.

TakeFive generates role-oriented knowledge graphs given an input sentence. The algorithm begins by detecting the verb (lemma and VerbNet verb class), along with its arguments, and then it relates it to their corresponding VerbNet roles. According to our running example 2, TakeFive detects the verb *conquered* and then extracts the VerbNet roles of this verb, i.e., *Conqueror* and *Theme*. Finally, it assigns the role fillers, i.e., *The Spaniards* as a filler of *Conqueror* and *the Incas* as the filler of the VerbNet role *Theme*. The backbone of TakeFive follows a step-wise approach:

1. preprocessing step for extracting dependencies and frame annotations using existing tools (i.e., CoreNLP and Word Frame Disambiguation respectively),
2. detecting (CoreNLP-derived, mainly syntactic) interface roles,
3. detecting VerbNet specific roles (mainly semantic) for a certain frame,
4. checking the compatibility between interface and semantically specific roles.

In other words, we aim at using background knowledge and formal reasoning to associate semantic roles with syntactic dependencies. In the rest of the paper, we use the following terminologies:

- *CoreNLP interface roles* for the roles generalizing CoreNLP dependencies as well as resource-specific semantic roles;
- *VerbNet specific roles* for the VerbNet roles related to a certain verb sense;
- *VerbNet interface roles* for the roles that subsume VerbNet specific roles in Framester, and are subsumed by an interface role.

[12] https://www.w3.org/TR/rdf11-primer/.

[13] http://etna.istc.cnr.it/framester2/sparql.

[14] https://ixa2.si.ehu.es/ukb/.

[15] http://babelfy.org/.

[16] https://lipn.univ-paris13.fr/framester/en/wfd_html.

det, Spaniards-2, The-1,
nsubj, conquered-3, Spaniards-2,
root, ROOT-0, conquered-3,
det, Incas-5, the-4,
dobj, conquered-3, Incas-5,

Fig. 2 Dependency tree triples representation of CoreNLP

4.1 Preprocessing step

Framester and CoreNLP For a given input sentence, frame detection using Word Frame Disambiguation (WFD) is performed. It uses Babelfy as a WSD algorithm and then uses the mappings between BabelNet Synsets and FrameNet frames as given in Framester. The dependency tree associated with a given input sentence is extracted using CoreNLP. Figure 2 shows a dependency tree returned by CoreNLP for the running example.

Assigning Interface roles to CoreNLP dependencies TakeFive is based on 23 simple heuristics for mapping CoreNLP dependency triples to CoreNLP interface roles. For the running example, we have a dependency `nsubj, conquered-3, Spaniards-2` related to the verb `conquered`, and its argument `Spaniards`. Dependency types such as `nsubj, dobj,...` are generalized to *CoreNLP interface roles* through a set of heuristics, e.g., by applying the rule $nsubj \rightarrow Agent$, i.e., the role `Agent` is assigned to the argument *Spaniards*. The set of CoreNLP interface roles include *{Agent, Undergoer, Recipient, Eventuality, and Oblique}*.

4.2 TakeFive: a semantic role labeling algorithm

This section discusses the two algorithms proposed for labeling a given sentence with the VerbNet specific and VerbNet interface as well as a way to check the compatibility between the CoreNLP interface roles (as assigned previously) and VerbNet interface roles. Algorithm 1 computes VerbNet interface and specific roles of extracted verbs from an input sentence.

4.2.1 Computing VN interface and specific roles

Algorithm 1 takes the preprocessed information as input, i.e., (a) the sentence, (b) the dependency tree obtained by CoreNLP, and (c) the output of frame annotations obtained using WFD. It then returns the input sentence labeled with VerbNet specific as well as VerbNet Interface roles. If the verb is polysemic, then it uses frame detection for extracting VerbNet roles (line 2–4, see Algorithm 2); otherwise, it gets the verb sense using the SPARQL query in Fig. 4. If it returns more than one verb sense, it selects the one which is most frequent (see the query in Fig. 7) and extracts the VerbNet specific roles along with VN interface roles, if any (line 6–10, see Fig. 6). If the result is empty, it uses frame detection for obtaining the VerbNet roles described in Algorithm 2.

Algorithm 2 is used in the case of polysemous verbs. The algorithm takes as an input a sentence and annotates it with frames (line 1). If there are no frame annotations, it takes the most frequent verb sense using SPARQL query in Fig. 7 and then the VN specific and interface roles associated to this verb sense through SPARQL query in Fig. 6 (line 2–4). If the Word Frame Disambiguation API returns multiple frames and there is a relation between these frames, the most specific frame is chosen (line 6-15). Then, given the *verb* and the chosen frame, VerbNet senses are extracted using Fig. 5 (line 16). If no verb senses are returned then the most frequent verb senses are extracted for getting the VerbNet specific roles (line 18-19). However, if there is more than one verb sense then the most frequent verb sense is chosen. An intersection of both the queried and returned sets of verb senses is taken and the verb sense with the highest ranking based on the frequency of verb senses in WordNet is selected and the corresponding VN role is returned (line 22–28). If both the above cases are false, the VN role associated with the verb sense is selected using the query in Fig. 6 (line 31).

Algorithm 1: Computation of pairs (VerbNet interface roles, VerbNet specific roles) for each verb in a given sentence.

```
Data: CoreNLP and Framester preprocessed information of an
      input sentence; an input sentence
Result: Pairs (VerbNet interface roles, VerbNet specific roles) for
        each verb senses of the extracted verbs in the input
        sentence
1  foreach verb do
2      if verb is polysemic then
3          see Algorithm 2
4      end
5      else
6          verbSenses ← get verb sense using the query in Fig. 4;
7          if verbSenses! = ∅ then
8              verbSense ← obtain the most frequent verb senses
               using the SPARQL query in Fig. 7 ;
9              vnrole ← given the verbSense extract the VN roles
               using the SPARQL query in Fig. 6 ;
10         end
11         else
12             see Algorithm 2
13         end
14     end
15 end
```

4.2.2 Checking compatibility of CoreNLP interface roles

The objective here is to return all roles and fillers for each argument of verbs of the input sentence if the inter-

Algorithm 2: Obtaining VerbNet Roles using Frame Detection if no frames are obtained.

```
Input: sentence, verb
Output: 3-tuple containing verb, its VerbNet specific role,
        VerbNet Interface role
$frames \leftarrow getFrames(sentence)$;
if $frames$ is $\emptyset$ then
    $verbSense \leftarrow$ get the most frequent verb sense using the query 7 ;
    $vnrole \leftarrow$ given the verbSense get the VN roles using the query 6 ;
end
for $f_1 \in frames$ do
    for $f_2 \in frames$ do
        if $f_1 \sqsubseteq f_2$ then
            $frameSpecific \leftarrow f_1$
        end
        else if $f_2 \sqsubseteq f_1$ then
            $frameSpecific \leftarrow f_2$
        end
    end
end
$verbSenses \leftarrow$ given $verb$ and $frameSpecific$ using the query 5;
if $verbSenses == \emptyset$ then
    $verbSense \leftarrow$ get the most frequent verb sense using the query 7 ;
    $vnrole \leftarrow$ given the verbSense get the VN roles using the query 6
end
else if $len(verbSenses) > 1$ then
    $verbSenseQueried \leftarrow$ obtain the most frequent verb senses using the query 7 ;
    if $verbSenseQueried \in verbSenses$ then
        $vnrole \leftarrow$ given the verbSense extract the VN roles using the query 6
    end
    else
        $verbSense \leftarrow$ get the next most frequent verb sense obtained by the query 7 until the verbSense matches with $verbSenseQueried$.
    end
end
else
    $vnrole \leftarrow$ given the verbSense extract the VN roles using the query 6
end
```

```
vsprepositionselection:Conquer_42030000-with-Instrument
a  vnschema:SensePrepSelection ;
vnschema:hasGenericArgument  vndata:Instrument ;
vnschema:hasPreposition vndataprep:with ;
vnschema:hasVerbSense vndata:Conquer_42030000  .
```

Fig. 3 Triples for obtaining VerbNet arguments

```
SELECT DISTINCT ?verbsense
   WHERE {
     ?verbsense rdfs:label ?verblemma ;
           a fschema:VerbSense .
    filter regex(?verblemma, "conquer", "i") }
```

Fig. 4 Query for extracting Verb Senses given a verb lemma

```
SELECT DISTINCT ?verbsense
WHERE {
?verbsense rdfs:label ?verblemma ;
                a vn31schema:VerbSense;
                skos:closeMatch ?frame }
```

Fig. 5 Query for mapping frames to verb senses when the verb is polysemous in VerbNet

face roles assigned using the two methods (i.e., heuristics and Algorithm 1) are compatible. Let O be equal to $\{Agent, Undergoer, Recipient, Eventuality\}$, C be the CoreNLP interface roles (assigned using heuristics such as $nsubj \rightarrow Agent$), V be the VerbNet interface roles, and R be the VerbNet specific roles, where V and R are returned by Algorithm 1 for a given sentence. For $v_1 \in V$, $c_1 \in C$ and $r_1 \in R$, if $v_1 = \emptyset$ and $r_1 = \emptyset$, then c_1 is assigned. However, if $v_1 \neq \emptyset$ and $r_1 \neq \emptyset$, then the following algorithm is defined:

- The algorithm starts by choosing $verb$ having at least one VN sense and takes c_1 and v_1. If $c_1 \in O$ and $v_1 \in O$ then the pair (c_1, v_1) is marked compatible and r_1 is returned such that $r_1 \in R$ and v_1 `associated_to` r_1 (returned by Algorithm 1).
- If c_1 and v_1 associated with $verb$ are *oblique* check for a preposition along with CoreNLP dependencies triples having the modifier *nmod* and then return r_1 such that r_1 is compatible with the preposition according to VerbNet verb arguments. The association between the VerbNet arguments and the prepositions is already defined in VerbNet and now standardized in RDF in Framester linguistic linked data hub as shown in Fig. 3 (we also make use of the Preposition Project dataset, which is another linked dataset in Framester).
- If $c_1 = Agent$ or $c_1 = Undergoer$ and $c_1 \neq v_1$ then select the top role of the subsumption hierarchy associated to VerbNet interface role (defined by the predicate `fschema`[17] `:subsumedUnder`). If the top role is *Theme* then select v_1.
- If all the previously defined rules are false or if there is no mapping between c_1 and v_1 then return c_1.

5 Evaluation

This section details the experimental setting, and the two evaluation procedures for measuring the performance of the TakeFive SRL algorithm. It also describes a comparison between TakeFive and other SRL tools.

[17] https://w3id.org/framester/schema/.

Fig. 6 Query to retrieve the VerbNet role for a certain (optional) interface role and a certain verb sense

```
SELECT DISTINCT ?interfacerole ?verbnetrole
WHERE {
   ?verbnetrole a vn31schema:Argument;
   vn31schema:inVerbSense <VERBNET SENSE>
   OPTIONAL
   {
    ?verbnetrole fschema:subsumedUnder+ ?interfacerole.
?interfacerole a fschema:InterfaceRole } }
```

```
SELECT DISTINCT ?verbsense
WHERE {
?verbsense rdfs:label <verblemma> ;
a vn31schema:VerbSense .
?verbsense skos:closeMatch ?wnsense .
?wnsense schema:tagCount ?freq .
FILTER ((datatype(?freq)) = xsd:int) }
ORDER BY DESC(?freq) LIMIT 1
```

Fig. 7 Query to retrieve most frequent verb senses given a verb lemma

5.1 Implementation details

The algorithm has been developed in Python and uses REST-APIs for Framester and Stanford CoreNLP. It also employs Py4J[18] as a bridge between Python and Java. A Java class was developed that can directly be called from the main Python code through Py4J. It can be faster if a cache mechanism is used to store Framester results and SPARQL queries to the Framester endpoint. The TakeFive SRL tool is open source and freely available online[19].

5.2 Evaluation setting

The performance evaluation was conducted for verifying if the chosen VerbNet roles associated with fillers are correct or not. We used two different datasets. The first one was the WSJ section of the Penn Treebank annotated with VerbNet and PropBank labels[20]. These annotations include the VerbNet and PropBank roles associated with each verb of each sentence of the dataset and related to each filler. As an example, consider the following sentence contained in the WSJ annotated dataset:

The Canadian pig herd totaled 10,674,500 at Oct. 1,
down 3 from a year earlier,
said Statistics Canada, a federal agency. (3)

The two verbs *totaled* and *said* are indicated in the annotations together with their VerbNet verb classes, as well as their VerbNet and PropBank roles and fillers. In particular, Table 1 shows the annotations for sentence 3.

In order to test our algorithm in out-of-domain data, as the second dataset, we used the PropBanked Brown corpus [46] as it is also mapped into VerbNet thematic roles in the SemLink resource. The Brown Corpus is a standard corpus of American English that consists of about one million words (about 500 samples of 2000+ words each) of English text divided into fifteen sections. The reason to create such a corpus was to provide a heterogeneous sample of English text useful for comparative language studies.

The performance evaluation was conducted by computing precision, recall and F1 score using the official CoNLL-2009 scorer[21] [14]. The CoNLL-2009 scorer evaluates the semantic frames by reducing them to semantic dependencies. A semantic dependency from every predicate to all its arguments is created. These dependencies are labeled according to their corresponding arguments. Additionally, a semantic dependency from each predicate to a virtual ROOT node is added. The latter dependencies are labeled with the predicate senses. This approach guarantees that the semantic dependency structure forms a single-rooted, connected (but not necessarily acyclic) graph. It can be seen that the scoring strategy gains some points even though a system assigns the incorrect predicate sense. For further details refer to [13,14].

In order to use the CoNLL scorer when comparing to other methods, we formatted the output of TakeFive as well as that of the other tools as required by the CoNLL-2009 scorer. Finally, for compliance purposes, we employed SemLink[22] to map VerbNet roles to PropBank roles.

5.3 Results

The results obtained by TakeFive have been compared to other state-of-the-art methods, including SEMAFOR, Pikes, FRED, and PathLSTM (for details (see Sect. 2)). As FRED and TakeFive are two resources maintained by overlapping teams, we have also combined their results by including all the VerbNet roles extracted by FRED into the results of Take-

[18] https://www.py4j.org/.

[19] https://github.com/TakeFiveSRL/TakeFiveSRL.

[20] http://verbs.colorado.edu/semlink/.

[21] https://ufal.mff.cuni.cz/conll2009-st/scorer.html.

[22] https://verbs.colorado.edu/semlink/.

Table 1 Annotations for example sentence 3

Verb	Verb class	VerbNet role	PropBank role	Filler
Say	37.7-1	Topic	ARG1	The Canadian pig herd totaled 10,674,500 at Oct. 1, down 3 from a year earlier
Say	37.7-1	Agent	ARG0	Statistics Canada, a federal agency
Total	54.1-1	Theme	ARG1	The Canadian pig herd
Total	54.1-1	Value	ARG2	10,674,500

Table 2 Labeled and unlabeled precision, recall, and F1 values of TakeFive, TakeFive+FRED, SEMAFOR, Pikes, FRED, and PathLSTM on the WSJ corpus

Method	Lab. Prec. (%)	Lab. Recall (%)	Lab. F1 (%)	Unlab. Prec. (%)	Unlab. Recall (%)	Unlab. F1 (%)
TakeFive	80.12	76.04	78.02	85.09	80.44	82.70
SEMAFOR	81.05	77.01	78.97	87.32	82.97	85.09
TakeFive+FRED	82.55	78.48	80.46	87.60	83.18	85.33
FRED	74.02	72.36	73.18	83.11	81.65	82.37
Pikes	72.11	70.62	71.35	79.27	78.15	78.70
PathLSTM	73.66	71.65	72.64	82.64	80.63	81.62

Table 3 Labeled and unlabeled precision, recall, and F1 values of TakeFive, TakeFive+FRED, SEMAFOR, Pikes, FRED, and PathLSTM on the Brown corpus

Method	Lab. Prec. (%)	Lab. Recall (%)	Lab. F1 (%)	Unlab. Prec. (%)	Unlab. Recall (%)	Unlab. F1 (%)
TakeFive	68.86	64.20	65.29	73.13	78.97	70.02
SEMAFOR	69.11	65.34	66.17	75.96	70.07	73.10
TakeFive+FRED	70.32	66.21	68.33	75.12	71.85	73.88
FRED	62.21	60.16	61.07	71.22	69.15	70.77
Pikes	60.79	58.08	59.11	67.92	66.87	66.11
PathLSTM	61.29	59.68	60.31	70.22	68.22	69.18

Five; we named this new algorithm as TakeFive+FRED. First of all, the *onf* files (2454) of the WSJ corpus[23] were processed. Each file contains input sentences and their parse trees. The gold standard has 74977 rows. Each row corresponds to a verb in a given sentence and includes the VerbNet verb class, fillers, and VerbNet and PropBank roles associated with that verb. Different rows might refer to the same sentence (as there might be several verbs within a given sentence). As already mentioned, when labeling each sentence using TakeFive, we extracted Framester and CoreNLP information (frames, dependency triples, POS tags, etc.). To speed up the experiments, we used a cache mechanism so that the information is downloaded only once (the cache mechanism is not currently available in the on-line version of TakeFive software).

Table 2 shows the labeled and unlabeled precision, recall, and F1-measure values of TakeFive, TakeFive+FRED, and the competitors, SEMAFOR, Pikes, PathLSTM, and FRED on the WSJ corpus. Table 3 shows instead the results on the Brown corpus. *Labeled scores* are related to the correct identification of labeled dependency, whereas *Unlabeled scores* do not take into account labels. For example, for the correct proposition: *verb.01: ARG0, ARG1, ARGM-TMP*, the system that generates the following output for the same argument tokens *verb.02: ARG0, ARG1, ARGM-LOC* receives a labeled precision score of 2/4 because two out of four semantic dependencies are incorrect: the ROOT dependency is labeled “02” instead of “01” and the dependency to the “ARGM-TMP” is incorrectly labeled “ARGM-LOC”. On the other hand, the same example would receive an unlabeled precision score of 4/4. SEMAFOR is the method with the highest accuracy. Our proposed approach is the second best; however, the numbers are very close to that of SEMAFOR, especially for the labeled case. We have done some investigation and noticed that Framester does not return any elements if they are not semantically linked to each other. This is related to the intrinsic nature of how Framester has been created and consists of: several ontologies linked to each other where VerbNet roles have been matched to FrameNet semantic frames. Moreover, the way we have combined CoreNLP

[23] The dataset was made available by the Linguistic Data Consortium: https://www.ldc.upenn.edu.

with Framester is straightforward and does not exploit any machine learning technique yet. The situation improves when we augment its results with those of FRED, not necessarily matched against certain ontologies and whose roles and frames are created using several heuristics. Basically, FRED can generate a higher number of output elements but not necessarily correct. Besides, we noticed that FRED has some internal issues with the offset extraction of words of the sentence and this affects the output representation of FRED and TakeFive+FRED. Nevertheless, the combined approach TakeFive+FRED can slightly outperform SEMAFOR. FRED captures complementary roles that TakeFive is unable to detect. We are certain that fixing FRED offset extraction of words of the sentences will allow us to keep improving the results of FRED and TakeFive+FRED (i.e., thus improving the 0.24 difference with SEMAFOR for the unbalanced F1). Besides, the employment of machine learning techniques when combining CoreNLP and Framester will certainly produce further benefits for both the labeled and unlabeled cases of FRED and TakeFive+FRED. For all these observations, it results in difficulty to find a generalization effect observable from our approach. It basically depends on the coverage of the queries shown in Figs. 5, 6, and 7. If we improve the coverage and mapping of the included resources (it is a current ongoing work we are already carrying out to upgrade Framester), we would definitely include better results. As far as Pikes is concerned we have observed that it misses some important roles for tokens. Its SRL engine is based on mate-tools[24], further developed in [47] in 2009. Similar conclusions, and similar performances with Pikes, can be drafted for PathLSTM that is also based on mate-tools. Based on the intuition that each of the methods above can capture complementary information, we believe that future experiments on an optimal combination of multiple SRL approaches might yield even better results.

A different strategy? While the results with CoNLL indicate that TakeFive performs as well as state-of-the-art methods, and better in an ensemble, in a semantic web context the evaluation strategy may be too lightweight. In order to test this, a different evaluation strategy has been conducted (not detailed here for space reasons, see [10]), which follows more closely the kind of SRL extraction that is supposed to be represented in a knowledge graph. The results with this second evaluation strategy show a lower accuracy (more than half than the one obtained with the CoNLL scorer) because i) in strategy 2 we defined the score so that matching is verified only when the role filler contains all the exact words of the gold standard, and ii) in strategy 2 we took into account VerbNet roles. The first evaluation method is based on CoNLL2009 score which takes into account the headwords only, and PropBank roles. The latter are much lower in number with respect to their corresponding VerbNet elements. Moreover, matching the headword only for a certain filler probably oversimplifies the matches.

The main lesson learned is that NLP evaluation settings may be inadequate when measuring the absolute performance of a semantic task as complex as SRL. Since the contrastive results show that the differences in method performance are consistent, even if at different accuracy levels, the accuracy seems to entirely depend on the "resolution" or sensibility of the setting. We recommend defining knowledge-graph-oriented benchmarks and scorers, and, in the particular case of SRL, revisiting the way role ontologies are designed.

6 Conclusions and future work

In this paper, we have addressed the problem of semantic role labeling jointly using Framester and Stanford CoreNLP in a novel implemented algorithm, TakeFive. In particular, we aimed at detecting verb frames and their labeled arguments (semantic roles and fillers). To assess the quality of our approach, we have carried out a comparative performance evaluation between TakeFive, and other SRL tools including SEMAFOR, FRED, Pikes, and PathLSTM. TakeFive, the only one using a hybrid knowledge-based approach, is close to the best with the Wall Street Journal corpus from the Penn Treebank as well as the Brown corpus. We have also observed that a simple ensemble of TakeFive, and the FRED machine reader, produces the best overall results.

We have noticed that natural texts (even in the reasonably controlled production of the Wall Street Journal) contain many more linguistic phenomena than expected in existing manually developed resources such as VerbNet. (For this reason, FRED uses a greedy algorithm for SRL instead of one that is closed under one specific resource.)

As ongoing and future work, we aim at designing new role ontologies that respond to best practices in knowledge graph design. We also want to use ensemble learning approaches by combining multiple methods and feeding/controlling the ensemble pipeline by using existing linguistic resources for SRL, and heuristical methods.

As far as the new findings related to word embeddings are concerned, given the recent adoption of BERT for frame identification, one more direction that we would like to head regards the employment of BERT for finding semantic roles and frames and combine this new approach with TakeFive and existing lexical resources (e.g., CoreNLP). Employing new semantic textual similarity measures could also bring benefits for finding semantic roles and frames [48]. As future perspectives, a supervised modification of TakeFive will be explored which would then be compared with other

[24] http://code.google.com/p/mate-tools/.

supervised approaches exploiting deep neural networks (e.g., transformer-based deep neural network).

Last but not least, the employment of machine learning techniques related to the combination of Framester and lexical resources such as CoreNLP to come up with a new pair frame, roles will be investigated as well to further improve the promising results we obtained so far.

Funding Open access funding provided by Universitá degli Studi di Cagliari within the CRUI-CARE Agreement.

References

1. Gangemi, A., Navigli, R., Velardi, P.: The ontowordnet project: extension and axiomatization of conceptual relations in wordnet. In: Meersman, R., Tari, Z., Schmidt, D.C. (eds.) On the Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE, pp. 820–838. Springer, Berlin (2003)
2. van Assem, M., Gangemi, A., Schreiber, G.: Conversion of WordNet to a standard RDF/OWL representation. In: Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06), European Language Resources Association (ELRA), Genoa, Italy (2006). http://www.lrec-conf.org/proceedings/lrec2006/pdf/165_pdf.pdf
3. Narayanan, S., Fillmore, C.J., Baker, C.F., Petruck, M.R.: FrameNet meets the semantic web: a DAML+OIL frame representation. In: Proceedings of the 18th National Conference on Artificial Intelligence, AAAI, AAAI, Edmonton, Alberta (2002)
4. Nuzzolese, A.G., Gangemi, A., Presutti, V.: Gathering lexical linked data and knowledge patterns from framenet. In: Proceedings of the Sixth International Conference on Knowledge Capture, K-CAP '11, Association for Computing Machinery, New York, NY, USA, p. 4148. https://doi.org/10.1145/1999676.1999685 (2011)
5. Gangemi, A., Presutti, V.: A pattern science for the semantic web. Semant. Web **1**, 61–68 (2010)
6. Fillmore, C.J.: Frame semantics and the nature of language. Ann. N. Y. Acad. Sci. **280**(1), 20–32 (1976)
7. Baker, C.F., Fillmore, C.J., Lowe, J.B.: The berkeley framenet project. In: Proceedings of the 17th International Conference on Computational Linguistics, vol. 1 (1998)
8. Rouces, J., de Melo, G., Hose, K.: Framebase: Representing n-ary relations using semantic frames. In: European Semantic Web Conference, Springer, pp. 505–521 (2015)
9. Gangemi, A., Alam, M., Asprino, L., Presutti, V., Recupero, D.R.: Framester: A Wide Coverage Linguistic Linked Data Hub. In: EKAW (2016)
10. Recupero, D.R., Alam, M., Gangemi, A., Presutti, V.: Frame semantic parsing using framester knowledge graphs. In: Posters & Demonstrations and Industry Tracks Co-located with ISWC (2017)
11. Atzeni, M., Dridi, A., Recupero, D.: Reforgiato: using frame-based resources for sentiment analysis within the financial domain. Prog. Art. Intell. **7**(4), 273–294 (2018). https://doi.org/10.1007/s13748-018-0162-8
12. Recupero, D.R., Alam, M., Buscaldi, D., Grezka, A., Tavazoee, F.: Frame-based detection of figurative language in tweets [application notes]. IEEE Comput. Intell. Mag. **14**(4), 77–88 (2019). https://doi.org/10.1109/MCI.2019.2937614
13. Surdeanu, M., Johansson, R., Meyers, A., Màrquez, L., Nivre, J.: The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In: Conference on Computational Natural Language Learning (2008)
14. Hajič, J., Ciaramita, M., Johansson, R., Kawahara, D., Martí, M. A., Màrquez, L., Meyers, A., Nivre, J., Padó, S., Štěpánek, J., Straňák, P., Surdeanu, M., Xue, N., Zhang, Y.: The CoNLL-2009 Shared Task: Syntactic and Semantic Dependencies in Multiple Languages. In: Conference on Computational Natural Language Learning (2009)
15. Manning, C.D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S.J., McClosky, D.: The Stanford CoreNLP natural language processing toolkit. In: ACL, System Demonstrations (2014)
16. Kilgarriff, A.: The sketch engine: ten years on. Lexicography **1**, 1–30 (2014)
17. Kingsbury, P., Palmer, M.: From treebank to propbank. In: In Language Resources and Evaluation (2002)
18. Carreras, X., Màrquez, L.: Introduction to the CoNLL-2004 shared task: semantic role labeling. In: Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004, Boston, Massachusetts, USA, Association for Computational Linguistics. pp. 89–97. https://www.aclweb.org/anthology/W04-2412 (2004)
19. Carreras, X., Màrquez, L.: Introduction to the CoNLL-2005 shared task: semantic role labeling. In: Proceedings of the Ninth Conference on Computational Natural Language Learning, CONLL'05 (2005)
20. Yi, S.T., Loper, E., Palmer, M.: Can semantic roles generalize across genres. In: Proceedings of the Human Language Technology Conference/North American chapter of the Association for Computational Linguistics annual meeting (HLTNAACL2007) (2007)
21. Bauer, D., Rambow, O.: Increasing coverage of syntactic subcategorization patterns in FrameNet using VerbNet. In: IEEE Fifth International Conference on Semantic Computing (2011)
22. Baker, C., Ellsworth, M., Erk, K.: Semeval task 19: frame semantic structure extraction. In: 4th International Workshop on Semantic Evaluations (2007)
23. Fürstenau, H., Lapata, M.: Semi-supervised semantic role labeling via structural alignment. Comput. Linguist. **38**(1), 135–171 (2012)
24. Titov, I., Klementiev, A.: A Bayesian approach to unsupervised semantic role induction. In: 13th Conference of the European Chapter of the Association for Computational Linguistics (2012)
25. Lang, J., Lapata, M.: Unsupervised semantic role induction with graph partitioning. In: International Conference on Empirical Methods in Natural Language Processing (2011)
26. Giuglea, A.-M., Moschitti, A.: Semantic role labeling via FrameNet, VerbNet and PropBank. In: International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics, ACL-44 (2006)
27. Das, D., Schneider, N., Chen, D., Smith, N. A.: SEMAFOR 1.0: A probabilistic frame-semantic parser, Technical Report, Carnegie Mellon University (2010)
28. Presutti, V., Draicchio, F., Gangemi, A.: Knowledge extraction based on discourse representation theory and linguistic frames. In:

International Conference on Knowledge Engineering and Knowledge Management (2012)
29. Gangemi, A., Presutti, V., Recupero, D.R., Nuzzolese, A.G., Draicchio, F., Mongiovì, M.: Semantic Web. In: Semantic Web Machine Reading with FRED. IOS Press, pp. 873–893. https://doi.org/10.3233/SW-160240
30. Corcoglioniti, F., Rospocher, M., Aprosio, A.P.: Extracting knowledge from text with PIKES. In: ISWC 2015 Posters & Demonstrations Track (2015)
31. Roth, M., Lapata, M.: Neural semantic role labeling with dependency path embeddings, CoRR arXiv:1605.07515
32. Tan, S., Na, J.: Positional attention-based frame identification with BERT: A deep learning approach to target disambiguation and semantic frame selection, CoRR arXiv:1910.14549
33. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, vol. 1, pp. 4171–4186 (2019). https://doi.org/10.18653/v1/N19-1423
34. Schuler, K.K.: VerbNet: a broad-coverage, comprehensive verb lexicon, Ph.D. Thesis, Philadelphia, PA, USA, aAI3179808 (2005)
35. Fellbaum, C. (ed.): WordNet: An Electronic Lexical Database. MIT Press, Cambridge (1998)
36. Levin, B.: English Cerb Classes and Alternations: A Preliminary Investigation. University of Chicago Press, Chicago (1993)
37. Nuzzolese, A.G., Gangemi, A., Presutti, V.: Gathering lexical linked data and knowledge patterns from FrameNet. In: International Conference on Knowledge Capture (K-CAP) (2011)
38. Navigli, R., Ponzetto, S.P.: Babelnet: the automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artif. Intell. **193**, 217–250 (2012)
39. de Lacalle, M.L., Laparra, E., Rigau, G.: Predicate Matrix: extending SemLink through WordNet mappings. In: International Conference on Language Resources and Evaluation (2014)
40. Gangemi, A.: What's in a Schema?, pp. 144–182. Cambridge University Press, Cambridge (2010)
41. Lehmann, J., Bizer, C., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia—a crystallization point for the web of data. J. Web Semant. **7**(3), 154–165 (2009)
42. Hoffart, J., Suchanek, F.M., Berberich, K., Weikum, G.: Yago2: a spatially and temporally enhanced knowledge base from wikipedia. Artif. Intell. **194**, 28–61 (2013)
43. Nuzzolese, A.G., Gangemi, A., Presutti, V., Ciancarini, P., Musetti, A.: Automatic Typing of DBpedia Entities. In: Proceedings of the International Semantic Web Conference (ISWC), Boston, MA, US (2012)
44. Speer, R., Havasi, C.: Representing general relational knowledge in conceptnet 5. In: LREC, pp. 3679–3686 (2012)
45. Carlson, A., Betteridge, J., Kisiel, B., Settles, B., Hruschka Jr, E.R., Mitchell, T.M.: Toward an architecture for never-ending language learning. In: AAAI, vol. 5, p. 3 (2010)
46. Marcus, M., Kim, G., Marcinkiewicz, M.A., MacIntyre, R., Bies, A., Ferguson, M., Katz, K., Schasberger, B.: The Penn Treebank: Annotating predicate argument structure. In: Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8–11, 1994 (1994). https://www.aclweb.org/anthology/H94-1020
47. Björkelund, A., Hafdell, L., Nugues, P.: Multilingual semantic role labeling. In: 13th Conference on Computational Natural Language Learning: Shared Task, CoNLL (2009)
48. Shajalal, M., Aono, M.: Semantic textual similarity between sentences using bilingual word semantics. Prog. Artif. Intell. **8**(2), 263–272 (2019). https://doi.org/10.1007/s13748-019-00180-4

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.