Leveraging Predicate-Argument Structures for Knowledge Extraction and Searchable Representation Using RDF

Tomas Vileiniskis and Rita Butkiene

***Abstract*—Predicate-argument structures are best known as means to represent shallow semantics behind natural language sentences by employing semantic role labeling (SRL) technique. The latter serves as foundation for complex tasks like question answering, text summarization, plagiarism detection and others. In this paper we show how SRL and semantic web technology can be used to build a knowledge graph from open-domain natural language texts with the main goal of enabling semantically-flavored information retrieval on top of the resulting knowledge base. In particular, we propose a domain-agnostic ontology schema capable of capturing event-oriented knowledge and a modification of breadth-first search graph traversal algorithm for serving users information needs. Finally, we evaluate behavior of the whole framework by annotating part of WikiQA dataset and use the constructed knowledge graph to judge information retrieval effectiveness which shows promising results.**

***Index Terms*—Semantic search, knowledge graphs, RDF, semantic role labeling.**

I. Introduction

The evolution of natural language processing techniques leads to major advancements in different application domains ranging from text classification and categorization to complex knowledge extraction and question answering. Understanding and formalizing the knowledge behind free form text can be approached at multiple abstraction levels, starting from simple named entity resolution and going to shallow semantic representations obtainable by mapping semantic roles to syntactic constituents of a sentence. Semantic role labeling (SRL) helps to achieve the latter by extracting information into predicate-argument structure mostly following linguistic features only. E.g. having a sentence S_1: "*ICKE 2020 was held by Okayama University in Japan*", its predicate-argument representation would look like the following:

S_1: [A0: by Okayama University] [V: hold.01] [A1: ICKE 2020] [AM-LOC: in Japan]

The core concepts of SRL parse are verbs (predicates), main and adjunctive arguments. The main subject and object arguments (A0, A1) are essential companions of a predicate that determine its meaning, while adjuncts (AM-TMP, AM-LOC, AM-MN) are meant to provide additional information like temporal or locative aspects of an expressed event. SRL is a perfect fit for any information extraction task where the final goal is to end up with a fine-grained event-centric knowledge base. In particular, it's due to the fact that argument roles are determined in a domain-agnostic way, relying mostly on syntactical language features. Therefore, even different syntactic variations of the same expressed information can be determined as representations of the same knowledge bit. Having another sample sentence S_2: "*Okayama University held ICKE 2020 in Japan*" its predicate-argument representation looks like the following:

S_2: [A0: Okayama University] [V: hold.01] [A1: ICKE 2020] [AM-LOC: in Japan]

SRL parse on the above sentence leads to the very same predicate-argument structure as in sentence S_1 even being of different voice constructions.

In general, automatic SRL is approached by machine learning techniques where the annotation algorithms are trained on corpora such as FrameNet [1] or PropBank [2]. The task consists of two main parts: argument boundary detection and argument classification based on the roles played in a sentence.

In this paper we don't focus on the automatic SRL annotation peculiarities but usage of the resulting parse data structures for an open-domain knowledge base construction instead. We show how SRL can be utilized for shallow semantic text preprocessing which is then followed by deep semantic annotation to obtain fine-grained ontological types of subject and object arguments.

For knowledge representation we rely on Semantic Web technology. In particular, Resource Description Framework (RDF) is used to serialize extracted knowledge while a dedicated OWL (Web Ontology Language) ontology serves the purpose of capturing the extracted event semantics in a domain-agnostic way.

Last but not least, we propose a modification of breadth-first search algorithm to show how the resulting knowledge graph can be used to serve free word order user queries.

The rest of the paper is organized as follows. Section II overviews related work in our research domain. In Section III we present the overall architecture of our framework. Section IV explains the core concepts behind our proposed domain-agnostic, ontology-based knowledge graph and how the resulting RDF representation can be queried by a graph traversal algorithm. Finally, we discuss initial experimental findings in Section V before drawing conclusions in the last section of the paper.

II. Related Work

Applicability of SRL for open-domain information extraction highly depends on precision of the automatic SRL annotation methods. Over the years state-of-the-art has

Manuscript received December 15, 2019; revised April 23, 2020.

Tomas Vileiniskis and Rita Butkiene are with Kaunas University of Technology, Department of Information Systems, Studentu st. 50, Kaunas, Lithuania (e-mail: tomas.vileiniskis@ktu.edu; rita.butkiene@ktu.edu).

doi: 10.18178/ijke.2020.6.1.128

reached ~85% for F1 scores [3] on PropBank corpora [2]. Even being trained mainly on financial domain data, PropBank SRL annotators have been used to process open-domain texts in the past as well [4].

Relation identification and subsequent event extraction is a core prerequisite technique in multiple research areas. One of such is abstractive news event summarization [5], [6] where the main goal is to capture and model events in a way (e.g. semantic triples) that would allow clustering them by similarity while dealing with language style diversity and fact duplication at the same time. Since shallow semantic representations alone are usually not enough to solve such tasks effectively, additional methods like named entity linking and disambiguation [7], [8] are employed for deeper semantic analysis within predicate arguments. Fine-graining subjective/objective attributes of an event allows for more precise extracted knowledge normalization [9], [10].

The other research field where predicate-argument structures are taken advantage of is question answering (QA) [11]-[13]. In general, the main principle here is to apply SRL both for the sentences in target corpora and questions used as an input by the system. Finally, different kind of mapping rules and heuristics are applied to the corresponding semantic roles to determine best answer candidates.

Our work presented in this paper falls somewhere in-between of the overviewed research. As a main contribution we propose a framework to build a unique event-centric knowledge base leveraging SRL and semantic web technology that also enables free word querying on top of the resulting RDF graph.

III. Knowledge Extraction Framework

In this section we present the main idea and concepts of our proposed knowledge extraction framework. The goal here is to process natural language text in order to end up with unique event-specific "who did what" knowledge bit extractions. We'll refer to those as SRL triples throughout the paper. Each atomic SRL triple is expected to have:

- Predicate – carries the main event information typically expressed by a verb which has a corresponding entry in PropBank lexicon.
- Subject – identifies the "who" role played in the event. In PropBank, subjective arguments are usually marked as A0/A1 roles.
- Object – identifies the "what" role played in the same event. PropBank distinguishes those objective roles by marking them A1/A2 depending on the specific verb semantics.

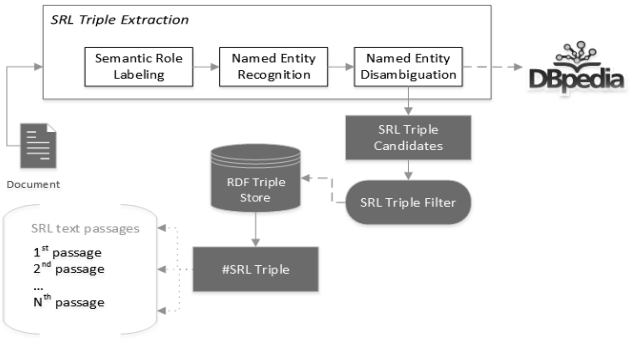

Fig. 1. SRL-based knowledge extraction framework.

Adjunctive arguments are optional and are treated as additional information for the expressed event however they do not determine the uniqueness of a SRL triple.

The conceptual schema of our knowledge extraction framework is depicted in Fig. 1. It can be decomposed into the building blocks out listed below.

SRL Triple Extraction Pipeline

The pipeline consists of multiple NLP components: SRL (includes tokenizer, part-of-speech tagger and dependency parser), named entity (NE) tagger and named entity disambiguator (NED). The predicate-argument structures extracted by SRL annotator undergo disambiguation or so called NE linking. In particular, A0, A1, A2 and AM-LOC arguments are further analyzed for mentions of entities that are looked up on external knowledge base DBpedia (https://wiki.dbpedia.org/) by employing a dedicated NED tool [8].

SRL Triple Filtering Component

This component takes as an input all of the annotations produced by the pipeline above and applies unique knowledge bit filtering rules. The rules require for the newly constructed triple <A0/A1; PREDICATE; A1/A2> to be unique in the RDF store. Uniqueness of the subject and object arguments are judged by their matches with DBpedia entity URIs while the predicate is normalized to its PropBank verb sense form, e.g. *hold.01*.

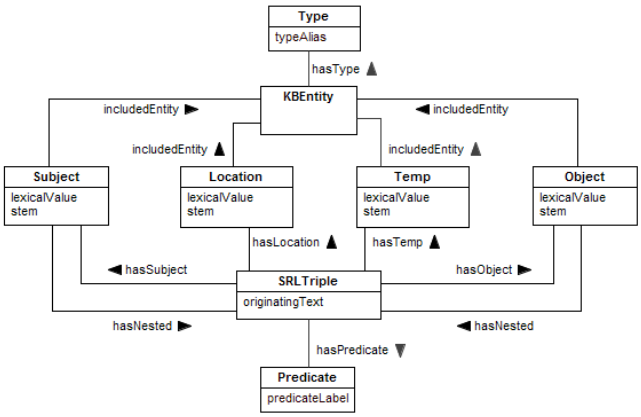

Fig. 2. SRL Triple ontology.

Given that a previously unseen knowledge bit is identified while parsing a sentence it gets asserted as an RDF triple in the local RDF triple store according to an ontology schema created for capturing both shallow and deep semantics of an event. The schema is depicted in Fig. 2 and further described in Section IV. However, if the knowledge bit appears to be a duplicate one of an already existing SRL triple assertion, its textual representation is considered to be included into the text passages list of the SRL triple in question. The motivation here is to maintain a set of textual proofs for each of the unique knowledge bits identified that can be later on utilized for semantically-aware information retrieval and result representation (see Section IV.C).

While still work in progress under our framework, multiple heuristics can be employed to determine whether the text passage is worthy the inclusion into the SRL triples' proof list.

E.g., the overall contextual quality of the originating triple's document could be judged or the quality criteria could be limited to the actual sentence or even the predicate-argument structure in question. Each distinct mention of the same event not only has different syntactic variations but can carry additional information expressed in adjunctive manner. Locative, temporal arguments define supplementary event characteristics, hence such textual proofs of mixed flavor are perfect candidates to enrich the passage list.

IV. Event-Centric Knowledge Graph

In this section we present an event-centric OWL ontology developed to serve our framework needs. While our knowledge extraction efforts are aimed at being domain-agnostic, the extracted events still carry semantics that need to be captured for multiple reasons. First, we utilize SPARQL ASK queries for repeated knowledge detection where triple patterns in the query reflect the SRL triple's uniqueness constraints. Second, ontological concept typing information is used at query time to enforce constraints on the graph traversal algorithm presented in Section IV B.

A. Ontology Schema

The ontology schema depicted in Fig. 2. shows the core concepts and their properties used to capture semantics of a predicate-argument structure at two levels.

Shallow semantics get represented following the concepts at the bottom part of the schema. They reflect the SRL annotations produced by the triple extraction pipeline. "SRLTriple", "Predicate", "Subject" and "Object" classes get instantiated whenever a new knowledge bit is identified during text processing. While the aforementioned class instantiation is mandatory to assert a new SRL triple, "Location" ant "Temp" classes are optional and get filled with T-box data only when the extracted event expression carries locative and/or temporal information.

Deep semantics are captured by utilizing the two concepts at the top part of the schema. "KBEntity" stands for a knowledge base entity mentioned in either of A0, A1, A2, AM-LOC or AM-TMP arguments. In our case, instances of "KBEntity" class represent DBpedia entity URIs against which the textual expression of an argument gets disambiguated. An object property "includedEntity" ties the entity URIs to their corresponding argument instances. Additionally, once there's a match to a DBpedia entity, we bring in its notable types following "rdf:type" predicate. This information is captured in "Type" class and later on used for semantic query expansion purposes.

An example T-box assertion of knowledge extracted from the S_1 sample sentence is shown in Fig. 3 (AM-LOC argument skipped for simplicity reasons).

B. Checking for Duplicate Knowledge

As mentioned earlier, mapping the extracted knowledge to an OWL ontology which is represented as an RDF graph allows to effectively check whether the newly identified knowledge bit is a duplicate of an already existing SRL triple assertion. In particular, we employ SPARQL ASK queries for this purpose. They get generated dynamically by analyzing the annotations produced by the knowledge extraction pipeline. Let's say that for a specific predicate-argument structure we have a set DE_{sub} of disambiguated entities in subject position and a set DE_{obj} of disambiguated entities in object position. Then, the SPARQL query triple patterns get produced by following the rule:

$$for\ e_{sub} \in DE_{sub} \longrightarrow ?sub\ :includedEntity\ e_{sub}$$

$$for\ e_{obj} \in DE_{obj} \longrightarrow ?sub\ :includedEntity\ e_{obj}$$

Such logic handles complex duplication checking cases when there are multiple entities in either subjective or objective arguments.

A sample SPARQL ASK query for a single entity mention check is provided down below:

```
<http://semantika.srl/srl/srltriple/04dcbe36-1e61-11ea-978f-2e728ce88125> a <srl:SRLTriple>;
  <srl:hasObject> <http://semantika.srl/srl/object/21987808-1e61-11ea-978f-2e728ce88125>;
  <srl:hasPred> <http://semantika.srl/srl/predicate/hold.01>;
  <srl:hasSubject> <http://semantika.srl/srl/subject/33c53494-1e61-11ea-978f-2e728ce88125>;
  <srl:originatingText> "ICKE 2020 was held by Okayama University in Japan".

<http://semantika.srl/srl/object/21987808-1e61-11ea-978f-2e728ce88125> a <srl:Object>;
  <srl:includedEntity> <http://dbpedia.org/resource/ICKE>;
  <srl:lexicalValue> "ICKE 2020";
  <srl:stem> "ICKE 2020".

<http://semantika.srl/srl/subject/33c53494-1e61-11ea-978f-2e728ce88125> a <srl:Subject>;
  <srl:includedEntity> <http://dbpedia.org/resource/Okayama>;
  <srl:lexicalValue> "by Okayama University";
  <srl:stem> "by Okayama Univers".

<http://semantika.srl/srl/predicate/hold.01> a <srl:Predicate>;
```

Fig. 3. Sample representation of partial T-box event assertion from S_1 in RDF turtle.

```
ASK WHERE {
  ?SRL a :SRLTriple.
  ?SRL :hasPredicate <PRED#SENSE>.
  ?SRL :hasSubject ?sub.
  ?sub :includedEntity <A0#KB_ENTITY_URI>.
  ?SRL :hasObject ?obj.
  ?obj :includedEntity <A1#KB_ENTITY_URI>.
}
```

Depending on the query result (ASK type queries produce TRUE or FALSE as Boolean values) the decision is made whether to process the extracted knowledge bit as a new SRL triple or go the other branch and consider adding its textual expression as an additional proof of already existing knowledge.

C. Searching the Knowledge Graph

Knowledge graph construction principles are oriented towards fulfilling one specific use case in this paper – serving users information needs. We propose a combination of breadth-first search (BFS) and spreading activation graph traversal algorithms to power full-text like querying capabilities taking use of the underlying captured event semantics.

The logic behind the querying algorithm is depicted in Fig. 4 down below.

Algorithm 1 Constrained spreading activation

INPUT: N - intial root node set
OUTPUT: R - result set of top-k SRLTriple nodes

```
srlTripleNodeReached ← FALSE
NeighborNodes ← ∅
procedure SpreadngActivation(N)
    for each node n ∈ N do
        SetInitialActivationScore(n)
        NeighborNodes ←SpreadInBFSManner(n)
        for each node neighbour ∈ NeighbourNodes do
            while srlTripleNodeReached ≠ TRUE do
                if neighbour is of type :SRLTriple then
                    srlTripleNodeReached ← TRUE
                    add neighbour to R
                else
                    continue SpreadInBFSManner(neighbour)
                end if
            end while
            srlTripleNodeReached ← FALSE
        end for
        NeighborNodes ← ∅
        sort R by activation score
    end for
end procedure
```

Fig. 4. Graph traversal algorithm flow.

The goal of the algorithm is to identify a specific knowledge bit within the knowledge graph that the user has in mind when expressing his information needs in a free text form.

Given a set of query keywords initial nodes get selected as a starting point for the traversing task. The selection is made by looking up the nodes that have keyword mentions in their lexical values or DBpedia entity matches. With the initial nodes identified, spreading through the graph edges starts in a BFS manner. It continues till a node of type “SRLTriple” is reached on the way, signifying of a possible answer-bearing knowledge bit. This is a different strategy from the usual spreading activation constraints (travel distance, max activation score etc.) Since multiple “SRLTriple” nodes eventually get reached, quality measures need to be applied to select the best resulting answer candidate. For this, activation scores are applied to the nodes on the pathway, e.g., an initial node matching a DBpedia entity will score higher than the one having a plain keyword match only. As a final step, “SRLTriple” node that has the highest activation score is picked and his originally extracted textual passages are emitted as proofs for the carried knowledge.

V. Experimental Evaluation

An early experimental evaluation was carried out to determine the effectiveness of both building the knowledge graph and querying it later on. In this section we provide observations on current algorithm behaviors and reasons behind that.

We chose WikiQA [14] dataset as it provides question-answer pairs making the experimental evaluation implementation more convenient. Also, the dataset covers a wide variety of domains with different semantics behind each of the question-answer tuple. We limited our experiment to a subset of WikiQA dataset leaving only sentences with NE mentions which is a prerequisite by our knowledge extraction rules.

TABLE I: Annotation Results

Target sentences	Sentences annotated	SRL triples extracted
1025	84	102

As shown in Table I. annotation recall is quite low (8.2%). The reasons are two-fold. First, SRL annotator makes quite a bit of glitches when working with open-domain texts, since it’s trained on domain-specific data. Second, it turned out there are not that many sentences in WikiQA dataset that would pass our knowledge extraction rules of <A0/A1; PREDICATE; A1/A2> where subject and object arguments are additionally expected to have a DBpedia entity mention.

TABLE II: Querying Results

Target queries	Served queries	Correctly answered queries
91	62	40

The query set for graph traversal task was also adjusted to reflect successful SRL annotations. As presented in Table II. we ended up with 92 target queries out of which 62 were successfully served, meaning that our algorithm emitted at least one answer node as response. Out of that, 40 queries returned the nodes that reached the highest activation scores out of all the resulting nodes. It gives us a precision of 64.5% while recall is lower at 44%. Precision seems to suffer mostly from full-text like matching of keywords to initial graph nodes when queries do not contain any NEs. For better recall values, current keyword expansion approach would need to be extended to cover predicates and wider semantic entity types as well.

VI. Conclusion

In this paper we presented a novel knowledge extraction framework that leverages SRL and semantic web technology capabilities to build and query a domain-agnostic, event-specific RDF knowledge graph. The main strength of our solution is the ability to cope with duplicate event mentions that helps to avoid unnecessary graph growth bypassing repeated knowledge assertions.

Our experimental observations show that this approach can be used to tackle knowledge extraction challenges in an open-domain environment, especially if the text processing pipeline is improved with better performing SRL annotation components. In future, we plan to relax the knowledge extraction rules to be capable of handling subjective and objective arguments not necessarily having entity mentions within them in order to further increase recall values.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

A came up with the research topic and proposed the conceptual idea behind the framework. A, B conducted the research by fine-graining knowledge extraction rules and designing the ontology schema. A implemented the framework under supervision of B. The experimental evaluation was led by B. A, B wrote the paper with A's focus on conceptual framework presentation and B's on result validation. Finally, both authors had approved the manuscript as a final version of presented research.

REFERENCES

[1] C. F. Baker, C. J. Fillmore, and J. B. Lowe, "The berkeley framenet projec," in *Proc. the 17th International Conf. on Computational Linguistics*, 1998, pp. 86-90. Association for Computational Linguistics.

[2] M. Palmer, D. Gildea, and P. Kingsbury, "The proposition bank: An annotated corpus of semantic roles," *Computational Linguistics*, vol. 31, no. 1, pp. 71-106, 2005.

[3] L. He, K. Lee, O. Levy, and L. Zettlemoyer, "Jointly predicting predicates and arguments in neural semantic role labeling," *arXiv preprint arXiv:1805.04787*, 2018.

[4] J. Christensen, S. Soderland, and O. Etzioni, "An analysis of open information extraction based on semantic role labeling," in *Proc. the 6th International Conf. on Knowledge capture*, 2011, pp. 113-120.

[5] M. Rospocher, M. van Erp, P. Vossen, A. Fokkens, I. Aldabe, G. Rigau, A. Soroa, T. Ploeger, and T. Bogaard, "Building event-centric knowledge graphs from news," *Journal of Web Semantics*, vol. 37, pp.132-151, 2016.

[6] R. E. Prasojo, M. Kacimi, and W. Nutt, "Modeling and summarizing news events using semantic triples," in *European Semantic Web Conference*, 2018, pp. 512-527, Springer, Cham.

[7] P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer, "DBpedia spotlight: Shedding light on the web of documents," in *Proc. the 7th International Conf. on Semantic Systems*, 2011, pp. 1-8, ACM.

[8] R. Usbeck, A. C. N. Ngomo, M. Röder, D. Gerber, S. A. Coelho, S. Auer, and A. Both, "AGDISTIS-graph-based disambiguation of named entities using linked data," in *Proc. International Semantic Web Conf.*, 2014, pp. 457-471, Springer, Cham.

[9] B. Haarmann, L. Sikorski, and U. Schade, "Text analysis beyond keyword spotting," in *Proc. the Military Communications & Information Systems Conf. (MCC)*, 2011.

[10] P. Exner and P. Nugues, "Using semantic role labeling to extract events from wikipedia," *DeRiVE@ ISWC*, pp. 38-47, 2011.

[11] D. Shen and M. Lapata, "Using semantic roles to improve question answering," in *Proc. the 2007 Joint Conf. on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, 2007, pp. 12-21.

[12] P. Moreda, H. Llorens, E. Saquete, and M. Palomar, "Combining semantic information in question answering systems," *Information Processing & Management*, vol. 47, no. 6, pp. 870-885, 2011.

[13] L. R. Pillai, G. Veena, and D. Gupta, "A combined approach using semantic role labelling and word sense disambiguation for question generation and answer extraction," in *Proc. 2018 Second International Conf. on Advances in Electronics, Computers and Communications (ICAECC)*, 2018, pp. 1-6, IEEE.

[14] Y. Yang, W. T. Yih, and C. Meek, "Wikiqa: A challenge dataset for open-domain question answering," in *Proc. the 2015 Conf. on Empirical Methods in Natural Language Processing*, 2015, pp. 2013-2018.

Tomas Vileiniskis is a Ph.D student at Kaunas University of Technology, Faculty of Informatics, Department of Information Systems. He has bachelor's and master's degrees in information systems engineering. His research interests cover semantic web technology like RDF, SPARQL, ontologies and its application in information extraction and information retrieval domains.

Rita Butkiene is an associated professor at the Kaunas University of Technology. She has defended her PhD in 2002 on the topic "Information system functional requirements specification method". Her research interests include information system engineering, ontologies, semantic technologies, and databases. She now is working on the development of semantic search framework, which includes information extraction from unstructured text and information retrieval from an ontology and semantically annotated text.