Frame Semantic Parsing using Framester Knowledge Graphs

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

1. University of Cagliari, Cagliari, Italy, 2. Université Paris 13, Paris, France, 3. CNR, ISTC, Rome, Italy.

Abstract. This paper introduces `TakeFive`, a new algorithm that performs frame semantic parsing using frame-oriented knowledge graph generated by Framester. `TakeFive` performs dependency parsing, identifies the words that evoke lexical frames, locates the roles and fillers for each frame, and runs coercion techniques.

1 Introduction

[1]So-called cognitive computing systems such as Google Now [3], SIRI[2], and IBM Watson[3] have provided strong evidence of what can be achieved with knowledge graphs used as background knowledge. In those cases, knowledge graphs are proprietary resources represented with proprietary formats. However, a key point of knowledge graphs, including linked data, is to represent entities and their relations with possibly additional attributes that may support temporal, spatial, causal inferences. Regardless of the format and the copyright, existing knowledge graphs share a common limit: they express facts that lack of contextual and situational information. This makes it hard if not impossible to go beyond encyclopaedic question answering or limited human-machine interaction tasks. The ability to automatically perform semantic frame parsing of natural language text is a requirement for evolving frame-oriented knowledge graphs. For example, FrameBase [4] has shown the usefulness of linguistic frames as a cognitive tool for semantic interoperability. Frame-semantic parsing refers to the combined tasks of frame detection and semantic role labeling on natural language text. Its output can greatly enrich knowledge graphs and semantic interoperability. Let us consider the following sentence from the Wall Street Journal (WSJ) dataset[4]:

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

[1] The research leading to these results has received funding from the European Union Horizon 2020 the Framework Programme for Research and Innovation (2014-2020) under grant agreement 643808 Project MARIO Managing active and healthy aging with use of caring service robots.

[2] `https://www.apple.com/ios/siri/`

[3] `https://www.ibm.com/watson/`

[4] Available from `https://catalog.ldc.upenn.edu/`

By performing frame-semantic parsing on this sentence, we recognize that the text fragment *to pour* evokes e.g. the frame `Cause_motion` from FrameNet, meaning that the sentence provides an occurrence of this frame, and that the text fragments *the investors* and *cash* respectively denote the argument of a role `Agent.cause_motion`, and the argument of a role `Theme.cause_motion`, as both involved in the `Cause_motion` situation occurrence. FrameNet, VerbNet and PropBank are three of the main resources for frames and roles which are abundantly used for Semantic Role Labeling (SRL). This paper proposes a novel method, called `TakeFive`, that relies on dependency (instead of categorial) parsing, one (or more) reference resources available from a novel linguistic linked data hub Framester [1]. We evaluate `TakeFive` with VerbNet frames and roles and compare it against existing methods for SRL-based knowledge extraction.

2 `TakeFive`, Semantic Role Labeling Algorithm

`TakeFive`[5] addresses the problem of detecting the verb (lemma and VerbNet verb class), along with its arguments, and relating them to their corresponding VerbNet roles. Consider the sentence: *The Spaniards conquered the Incas.* Here, our method should be able to detect the verb *conquered*, the fact that *The Spaniards* is the filler of the VerbNet role *Conqueror* whereas *the Incas* is the filler of the VerbNet role *Theme*. Verbs, fillers and roles are therefore the entities we are looking for and that we need to properly associate with the input sentence. The backbone of `TakeFive` is a two step approach: **(i)** preprocessing the sentence, where syntactic and semantic information are extracted and **(ii)** detecting (CoreNLP-derived, mainly syntactic) interface roles, (VerbNet-based, mainly semantic) specific roles for a certain frame, and checking the compatibility between interface and semantically specific roles.

Step 1: Framester and CoreNLP preprocessing. For a given input sentence we collect semantic information from Framester and syntactic information from Stanford CoreNLP: the usage of Word Frame Disambiguation (WFD)[6] allows detecting the frames evoked by each verb when the verb is polysemous, whereas CoreNLP provides a dependency tree along with the POS tags (see Figure 1). Here, `nsubj, conquered-3, Spaniards-2` related to the verb `conquered`, and its `Spaniards` argument. Dependency types such as `nsubj, dobj` are generalized to *interface roles* (e.g., *Agent, Undergoer, Recipient, Eventuality, Oblique*) to add a semantic layer on top of the syntactic one e.g., *nsubj* → *Agent*. By applying our heuristic *nsubj* → *Agent* to the dependency triple `nsubj, conquered-3, Spaniards-2`, we assign the role `Agent` to the argument *Spaniards*. As next step, we need to check if the CoreNLP interface role is compatible with the VerbNet interface role of the underlying verb (`conquered` in our example).

[5] Further details are available at `https://lipn.univ-paris13.fr/framester/en/srl`
[6] `http://lipn.univ-paris13.fr/framester/`

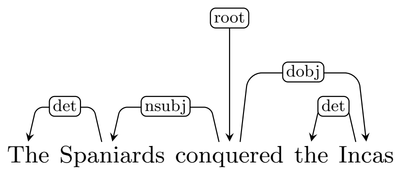

Fig. 1: Dependency graph generated by CoreNLP

Step 2: Compatibility between CoreNLP and VerbNet interface roles. `TakeFive` introduces an algorithm for checking the compatibility between the CoreNLP interface roles and VerbNet roles with respect to a verb occurring in a sentence. The first part of the algorithm takes as input a sentence, along with the CoreNLP and Framester information of the same sentence and generates a pair of VerbNet interface roles and VerbNet specific roles. Due to space constraints, we directly explain the algorithm using our example sentence. Consider two dependency triples (Listing 1 from `https://lipn.univ-paris13.fr/framester/en/srl`) {`nsubj, conquered-3, Spaniards-2`} and {`dobj, conquered-3, Incas-5`}. Using our heuristics, we assign the CoreNLP interface roles `Agent` and `Undergoer` to `Spaniards` and `Incas`, respectively. The VerbNet sense of the verb *conquered* is `Conquer_42030000` and the returned pairs (VerbNet interface role, VerbNet specific role) are: `(Agent, Agent.conquer_42030000), (Eventuality, Event.conquer_42030000)`.The second part of the algorithm checks the compatibility of CoreNLP interface roles detected using the heuristics defined in Step 1 and the VerbNet interface roles detected in the previous part of the algorithm. The objective here is to return all roles and fillers for each argument of verbs from the input sentence. For our example, it follows that the CoreNLP interface role `Agent` is equal to the VerbNet interface role and is returned. The same applies for the CoreNLP interface role *Undergoer*. `Patient.conquer_42030000` would be the VerbNet specific role that would be matched and the role *Patient* is returned. Therefore the final output would contain the role *Agent* for the argument *Spaniards* and the role *Patient* for the argument *Incas*.

3 Performance Evaluation

Several experiments were conducted for testing the performance of `TakeFive` and the results were compared with several existing tools such as SEMAFOR, FRED, Pikes and PathLSTM. Recently, we have presented FRED [2] as a machine reader to produce frame-based knowledge graphs. We combined FRED and `TakeFive` by including all the VerbNet roles and fillers extracted by FRED to the results of `TakeFive` when the latter does not extract roles information for a particular filler in general caused by the complexity of the sentence grammar. Conversely, if FRED detects a VerbNet role for a particular filler which has not been detected by `TakeFive`, it is likely to be a correct pair thanks to the Combinatory Categorial Grammar theory which FRED is built upon. The data set used for this purpose was the WSJ section of the Penn Treebank PropBank annotated with VerbNet and PropBank annotations[7]. These annotations indicate the VerbNet and PropBank roles associated to each verb of each sentence contained in the dataset and related to each filler. An evaluation analysis was conducted as follows: for each pair (role, filler) that was returned using our approach, it was verified against the gold standard annotations related to the same sentence and same verb. For each pair, the produced output contains $(role_{OUT}, filler_{OUT})$. This output was compared with the annotated pairs $(role_{ANN}, filler_{ANN})$ and a weighted score defined as follows: if $role_{OUT} == role_{ANN}$ and $filler_{OUT} == filler_{ANN}$ we assign 1; if $role_{ANN} \neq role_{OUT}$ but either there exists a subsumption relation between them or they are siblings, and $filler_{OUT} == filler_{ANN}$, then we assign a score of either 0.5 or 0.25. Otherwise, the weighted score has a value of 0. We performed a precision-recall analysis as follows: (i) true positives are counted when the weighted score for a pair is greater than 0, (ii) false positives are counted when the weighted score for the pair is equal to 0, (iii) false negatives are counted for all the annotation pairs that were not successfully retrieved by a given method, (iv) true negatives are represented by all the pairs (role, fillers) not retrieved by the algorithm for which there is no annotation. Table 1 shows the comparisons between our approach and the other competitors.

Method	Weighted Score	Precision	Recall	F1
TakeFive	0.174	0.156	0.22	0.185
TakeFive +FRED	0.193	0.176	0.201	0.191
SEMAFOR	0.050	0.038	0.031	0.034
Pikes	0.181	0.155	0.122	0.137
FRED	0.066	0.052	0.080	0.063
PathLSTM	0.101	0.095	0.094	0.094

Table 1: Results of TakeFive, TakeFive +FRED and the competitors.

4 Conclusions

This paper introduces a new algorithm for semantic role labeling, `TakeFive`, which aims at detecting verbs and their associated arguments. Several experiments show that the proposed approach outperforms the state of the art algorithms for semantic role labelling. Ongoing work focuses on defining a strategy to combine the existing methods for performance improvements.

References

1. Gangemi, A., Alam, M., Asprino, L., Presutti, V., Recupero, D.R.: Framester: A wide coverage linguistic linked data hub. In: EKAW, 2016. pp. 239–254 (2016)
2. Gangemi, A., Presutti, V., Recupero, D.R., Nuzzolese, A.G., Draicchio, F., Mongiovì, M.: Semantic web machine reading with FRED. Semantic Web 8(6), 873–893 (2017), https://doi.org/10.3233/SW-160240
3. Guha, R., Gupta, V., Raghunathan, V., Srikant, R.: User modeling for a personal assistant. WSDM '15 (2015)
4. Rouces, J., de Melo, G., Hose, K.: Framebase: Representing n-ary relations using semantic frames. In: European Semantic Web Conference (2015)

[7] https://github.com/ibeltagy/pl-semantics/blob/master/resources/semlink-1.2.2c/1.2.2c.okay