COPAAL – An Interface for Explaining Facts using Corroborative Paths

Zafar Habeeb Syed[1], Nikit Srivastava[1], Michael Röder[1 2], and Axel-Cyrille Ngonga Ngomo[1 2]

[1]Data Science Group, Paderborn University, Germany
zsyed|nikit@mail.uni-paderborn.de,
michael.roeder|axel.ngonga@upb.de
[2]Institute for Applied Informatics, Leipzig, Germany

Abstract. With the increasing uptake of knowledge graphs in domains as diverse as question answering, community-support systems and even personal assistants comes an increasing need for validated knowledge contained in these graphs. However, the sheer size and number of knowledge bases used in real-world applications makes manual fact checking impractical. Automated fact validation systems aim to compute the veracity of individual facts by evaluating the likelihood of these facts being true. In this demo, we present an interface for fact checking based on the COPAAL algorithm. Given triple whose veracity is to be evaluated, our interface provides (1) a score for the veracity of the triple, (2) evidence for the triple in the forms of paths, (3) explanation for the evidence in the form of verbalized RDF triples as well as (4) a graphical overview of the paths which support the input triple. We evaluate the performance of our fact checker, the quality of the verbalization we use and the usability of our user interface. The demo is available at `http://db-fact.cs.uni-paderborn.de:4200`.

1 Introduction

The Web follows a participatory paradigm, which has led to more than 150 billion RDF triples being published by thousands of independent data providers in more than 10,000 knowledge graphs (KGs).[1] The largest open KGs contains billions of triples pertaining to millions of entities. As open KGs are used in an increasing number of applications, developers and end users have an increasing need to check the veracity of triple before using them in applications, especially if these applications are mission-critical. We developed the COPAAL approach [3] to fact checking, which evaluates the veracity of RDF triples by combining RDFS semantics with path search in knowledge graphs. The approach was accepted as a full research paper at ISWC 2019. In this corresponding demo paper[2], we present (1) the user interface and REST service for fact checking based on COPAAL as well as (2) supplementary evaluation results pertaining to the verbalization of evidence and the system usability of the user interface. During the

[1] `http://lodstats.aksw.org/`

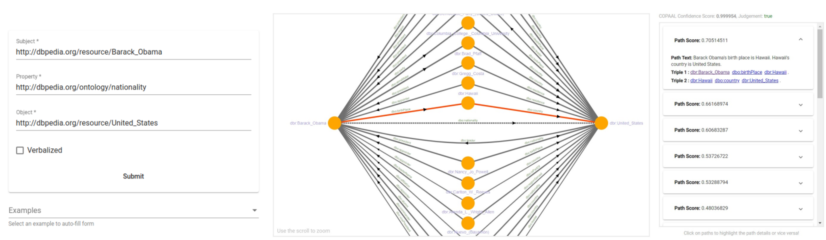
demo at the conference, we will present the strengths and weaknesses of the approach using selected examples as well as allow end users to check facts pertaining to DBpedia resources for which they would like to see evidence.

2 Summary of the Approach

With COPAAL, we address the *following problem*: Given an RDF knowledge graph $\mathcal{G}$ and a triple (s, p, o), compute the likelihood that (s, p, o) is true. E.g., `BarackObama` is a clearly a citizen of the `USA`, amongst others by virtue of having been born in `Hawaii`, which is located in the `USA`. However, this fact is not available in DBpedia 2016-10.[3] Given this particular version of DBpedia, our approach can compute paths between the resource `BarackObama` and `USA`, which corroborate the fact that `BarackObama` is a national of the `USA`. The intuition behind our approach is that certain sequences of properties (e.g., $x \xrightarrow{\texttt{birthplace}} y \xrightarrow{\texttt{country}} z$) have a high mutual information (MI) with certain predicates (e.g., `nationality`). We developed an efficient approach for computing this MI (score) and combining the MI of several paths to evaluate the veracity of particular facts. The exact computation details are given in [3].

3 System Overview

We developed a web service and a UI so that users can easily interact and perform fact validation using COPAAL. Figure 1 gives an overview of the three main components of our user interface – *input, path presentation and verbalization*.



(a) Input (b) Corroborative Paths and their explanations

Fig. 1: Overview of COPAAL

The **input** to COPAAL is a triple *(s, p, o)* whose correctness is to checked and for which evidence is to be provided. A user can enter the *subject*, *property* and the *object* directly into the interface (see Figure 1a). Note that the property must be an object property in our demo. In addition, users can choose to have the evidence for their input triple verbalized by selecting the "verbalize" option. Finally, the user can forward the input triple to the COPAAL service by clicking on the submit button.

[3] `http://downloads.dbpedia.org/2016-10/`

The COPAAL service computes **corroborative paths** for the input data and returns a set of paths and their scores, i.e., paths through the input graph which have a high MI with the input triple. In COPAAL, we visualize these paths using graphs[4] (see Figure 1b). Note that the dotted path indicates the triple to be checked and the solid paths represents the corroborative paths. One can navigate to view path explanations by clicking on a path. The explanations are either sequences of triples or (if the user chose to have **verbalized** evidence) sequences of sentences, which states the content of the corroborative paths in simple English sentences. We used the the rule-based LD2NL framework,[5] which is based on [1], to verbalize the triples in the paths.

As expected, our approach returns the path (e.g., $\texttt{BarackObama} \xrightarrow{\texttt{birthplace}} \texttt{Hawaii} \xrightarrow{\texttt{country}} \texttt{USA}$) as a main evidence for Barack Obama's nationality (score = 0.705, see Figure 1). Other paths pertaining to his alma mater, his presidency and his political affiliation further corroborate that Barack Obama is a US citizen. COPAAL computes a combined score which is also displayed to the end user. A binary (i.e., true/false) suggestion as to the truthfulness of the fact is also displayed to the user.

4 Evaluation

Corroborative paths. We evaluated the performance of COPAAL on 17 datasets. Details pertaining to the characteristics of all datasets as well as detailed insights derived from the evaluation are given in [3]. Our results on the four real-world datasets from [2] shown in Table 1 show that our approach clearly outperforms the state of the art. While our approach can perform poorly on rare predicates, the AUC-ROC results suggest that our approach is able to compute an appropriate score for most triples.

Table 1: AUC-ROC results of all approaches on *Real-World* datasets

	Birth Place	**Death Place**	**Education**	**Nationality**
COPAAL	**0.9441**	0.8997	**0.8731**	**0.9831**
PredPath	0.8997	0.8054	0.8644	0.9520
KL-REL	0.9254	**0.9095**	0.8547	0.9692
KS	0.7197	0.8002	0.8651	0.9789

Verbalization. We evaluated the verbalization underlying our demo with two groups—domain experts (66 persons) and non-experts (20 linguists). A set of triples and their verbalization were shown to the volunteers. The experts were asked to rate the verbalization regarding adequacy, fluency and *completeness*, i.e., whether all triples have been covered. The non-experts were only asked to rate the fluency. The experiment was carried out using 6 DBpedia resources.

[4] We use d3 js – A JavaScript library for generating interactive and dynamic graphs

[5] https://github.com/dice-group/ld2nl

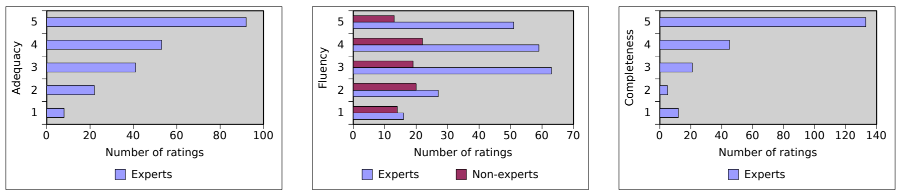


Fig. 2: Verbalization of RDF triples: adequacy (left), fluency (middle) and completeness (left) results

Our results revealed that verbalizing RDF is a difficult task. While the adequacy of the verbalization was assigned an average score of 3.92 by experts (see Fig. 2), the fluency was assigned a average score of 3.47 by experts and 3.0 by linguists (see Fig. 2). These results suggest is that (1) our framework generates sentences that are close to that which a domain expert would also generate (adequacy). However (2) while the sentences is grammatically sufficient for the experts, they are by linguists rated as being grammatically passably good but still worthy of improvement.

System usability We evaluated our user interface based on the System Usability Scale. 10 persons participated in the corresponding survey. Overall, we reached an SUS score of 79.3 (school grade: A-), which means that most end users would be willing to use the system and would recommend it to a friend if it were to be slightly improved.

5 Conclusion

In this demo, we present a first interface for fact checking using COPAAL [3]. We foresee a plethora of improvements in future works, including a natural-language interface (both spoken and written) and a natural language output channel for the evidence. Moreover, we will improve upon the verbalization of the paths.

Acknowledgements

This work has been supported by the German Federal Ministry of Transport and Digital Infrastructure (BMVI) in the projects LIMBO (no. 19F2029I) and OPAL (no. 19F2028A).

References

1. Ngonga Ngomo, A.C., Bühmann, L., Unger, C., Lehmann, J., Gerber, D.: Sorry, i don't speak sparql: translating sparql queries into natural language. In: Proceedings of the 22nd international conference on World Wide Web. pp. 977–988. ACM (2013)
2. Shiralkar, P., Flammini, A., Menczer, F., Ciampaglia, G.L.: Finding streams in knowledge graphs to support fact checking. In: 2017 IEEE International Conference on Data Mining (ICDM). pp. 859–864. IEEE (2017)
3. Syed, Z.H., Röder, M., Ngonga Ngomo, A.C.: Unsupervised discovery of corroborative paths for fact validation. In: Proceedings of the International Semantic Web Conference (2019)