Hierarchical Expected Answer Type Classification for Question Answering

Aleksandr Perevalov¹ and Andreas Both¹²

¹ Anhalt University of Applied Sciences, Köthen (Anhalt), Germany {aleksandr.perevalov,andreas.both}@hs-anhalt.de
² DATEV eG, Germany andreas.both@datev.de

Abstract. To know what a user's question is about is a crucial step in the Question Answering (QA) process. Thus, the Expected Answer Type (EAT) of a question enables to significantly narrow down the search field and improve the QA quality. In this paper, we present a Web user interface (UI) and a RESTful API for the hierarchical EAT classification over DBpedia. The provided functionality enables end-users to get the EAT predictions for 104 languages, see the confidence of the prediction, and leave feedback. In addition, the API enables researchers and developers to integrate the EAT classification into their systems.

Keywords: Expected Answer Type Classification · Target Type Identification · Knowledge Graph Question Answering · Entity Typing.

1 Introduction

The Knowledge Graph Question Answering (KGQA) systems are aimed to answer entity-oriented questions. For example, while asking a question – like "Where was Angela Merkel born?" – we expect to see an entity with the type "Place" (e.g., Hamburg). In this case, "Place" (or even better: "City") is the expected answer type (EAT). Such types are typically organized into hierarchical type ontologies [4] (e.g., DBpedia Ontology¹) depending on the particular knowledge graph used within a QA system.

Following the example question, the EAT hierarchy may look as follows: dbo:City \rightarrow dbo:Settlement \rightarrow dbo:PopulatedPlace \rightarrow dbo:Place² where the first type is the most specific one and the last – the most general one. Recently, many research papers have demonstrated that QA systems may benefit from the EAT classification [5,3,6].

In this paper, we present the Web UI and RESTful API for the hierarchical EAT classification over DBpedia³. As we extended our previously developed

Copyright C 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

¹ http://mappings.dbpedia.org/server/ontology/classes/

² dbo - is a prefix for http://dbpedia.org/ontology/

³ https://webengineering.ins.hs-anhalt.de:41009/eat-classification

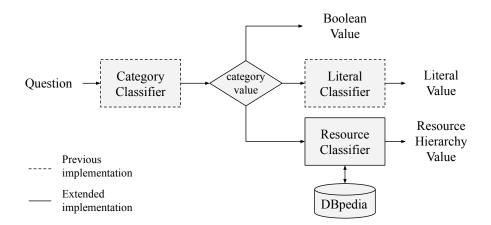


Fig. 1. The final architecture of the EAT classifier's back-end

approach [9], the predictions are available and might be compared for both the "existing" and the "improved" approach. The tool supports 104 languages, provides the prediction confidence as well as an opportunity to leave feedback for a given prediction. The RESTful interface to the functionality enables easy integration with other existing KGQA systems or future research.

2 Related Work

The expected answer type is sometimes referred to as *target type* in the context of entity-oriented search [1]. So-called Entity- and Type-Centric models were introduced in [1] to identify the target type of a question. These models are used to rank the queries given the entity- or type-related content [3]. The idea of incorporating an additional context to improve answer type predictions was proposed in work [12]. One of the ISWC 2020's Semantic Web challenge was addressing the answer type classification (SeMantic AnsweR Type prediction task, SMART) [7]. It has shown that transformer-based models demonstrate the highest results in this task [11,8]. The approach based on using external data (e.g., KGQA datasets) was introduced in paper [10]. Recently, the authors of [2] proposed a system for EAT prediction in a "distantly supervised fashion" (i.e., no manual data annotation is required), however, the evaluation results were not presented.

3 Approach and Implementation

The tool works on top of the approach previously developed by the authors [9] that is capable to identify not only resource answer types (e.g., dbo:City), but also literal (number, date, string) and boolean types. The extended approach is targeting the resource answer types by predicting the most specific EAT for a

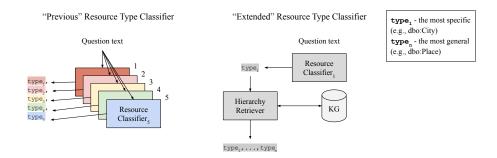


Fig. 2. The difference between resource type classifier of the previous and the extended approaches

given question. After doing so, the corresponding DBpedia hierarchy is fetched instead of an independent prediction of EAT for each granularity level (see Figure 2). Hence, the extended approach differs only in the resource classifier.

Figure 2 demonstrates that in the previous approach, no hierarchy consistency check is done. Thus, the predicted types may belong to a different hierarchy, which is unacceptable as the prediction becomes inconsistent. In addition, the hierarchy size is limited only to five types. On the other hand, the extended approach predicts the most specific resource answer type and fetches the rest of the hierarchy from a KG (e.g., DBpedia) thereafter (via hierarchy retriever). The hierarchy retriever just executes the SPARQL query and formats the final output.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?sType WHERE {
    <type> rdfs:subClassOf* ?sType .
    FILTER(CONTAINS(STR(?sType), "dbpedia.org/ontology"))
}
# the 'type' placeholder is replaced with the predicted type
```

Listing 1. Retrieving super types of a given answer type from DBpedia.

In this case, the resource answer type hierarchy is consistent and not limited to a specific size.

For training and evaluation, we used the DBpedia dataset of the SMART Task. We reuse our previously prepared multilingual extension for the dataset⁴ and fine-tune the classifier using multilingual language model⁵ that supports 104 languages.

The evaluation of the obtained EAT classifier demonstrated reasonable results: (1) category prediction – Accuracy := 0.977, (2) type ranking – NDCG@5 := 0.745; NDCG@10 := 0.710 [1]. The results are comparable to the 2020s

⁴ The multilingual dataset extension contains questions in 5 languages: https://github.com/Perevalov/iswc-classification

⁵ https://huggingface.co/bert-base-multilingual-cased

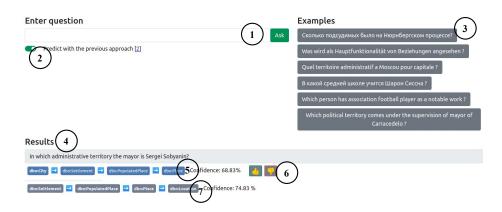


Fig. 3. The Web interface of EAT classifier

SMART winner [11]. The final architecture of the EAT classifier is shown in Figure 1.

The Web UI of the EAT classifier is presented in Figure 3. The description of the numbered elements is as follows: (1) question input field, (2) switch button that enables to get the additional prediction with the model [9], (3) section with example questions, (4) results section where the asked question is listed, (5) the prediction result and the confidence from the new model, (6) feedback buttons (only for the new model's prediction), and (7) the prediction result as well as the confidence from the model [9].

The RESTful API⁶ of the EAT classifier has GET endpoints for both currently provided models. After providing the parameter question containing the question's text, the service returns a dictionary with the following fields: category (holds on of "resource", "literal", or "boolean"), answer_type (if canse of predicting not a resource, then the primitive data is stored in the array, e.g., ["number"] or ["boolean"], else one or more elements corresponding to the resource hierarchy, e.g., ["dbo:Person", "dbo:Agent"]); and confidence – a float value $f \in [0, 1]$ corresponds to the models confidence of the prediction.

4 Conclusion

In this work, we presented the Web UI and the RESTful API for retrieving EAT predictions and validating EAT classifiers. Currently, two EAT components are integrated. Among the DBpedia Ontology types (resources), the tool is capable to distinguish between literal and boolean answer types. The EAT classifier is capable of providing predictions for questions given using up to 104 languages, and showed reasonable quality w.r.t. SMART Task evaluation over the DBpedia dataset.

⁶ https://webengineering.ins.hs-anhalt.de:41020/docs

For future work, we plan to improve the approach w.r.t. the quality and extend it to other ontologies (e.g., Wikidata) to enable comparability. We would like to flatten the architecture of the classifier (see Figure 1) s.t., only one model is used for the prediction. In addition, it is worth paying attention to the robustness of the model w.r.t. corrupted input data (e.g., spelling mistakes).

References

- Balog, K., Neumayer, R.: Hierarchical target type identification for entity-oriented queries. In: Proceedings of the 21st ACM international conference on Information and knowledge management. pp. 2391–2394. CIKM '12, ACM, New York, NY, USA (2012). https://doi.org/10.1145/2396761.2398648
- Dash, S., Mihindukulasooriya, N., Gliozzo, A., Canim, M.: Type prediction systems. CoRR (2021), https://arxiv.org/abs/2104.01207
- Garigliotti, D., Hasibi, F., Balog, K.: Target type identification for entity-bearing queries. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 845–848. SIGIR '17, ACM, New York, NY, USA (2017). https://doi.org/10.1145/3077136.3080659
- Garigliotti, D., Hasibi, F., Balog, K.: Identifying and exploiting target entity type information for ad hoc entity retrieval. Inf. Retr. 22(3–4), 285–323 (Aug 2019). https://doi.org/10.1007/s10791-018-9346-x
- Höffner, K., Walter, S., Marx, E., Usbeck, R., Lehmann, J., Ngonga Ngomo, A.C.: Survey on challenges of question answering in the semantic web. Semantic Web 8(6), 895–920 (2017)
- Kamath, S., Grau, B., Ma, Y.: Predicting and integrating expected answer types into a simple recurrent neural network model for answer sentence selection. Computación y Sistemas 23 (2019)
- Mihindukulasooriya, N., Dubey, M., Gliozzo, A., Lehmann, J., Ngomo, A.C.N., Usbeck, R.: SeMantic AnsweR Type prediction task (SMART) at ISWC 2020 Semantic Web Challenge. CoRR/arXiv (2020), https://arxiv.org/abs/2012.00555
- Nikas, C., Fafalios, P., Tzitzikas, Y.: Two-stage semantic answer type prediction for question answering using BERT and class-specificity rewarding. In: Proceedings of the SeMantic Answer Type prediction task (SMART) at ISWC 2020. CEUR Workshop Proceedings, vol. 2774, pp. 19–28. CEUR-WS.org (2020), http://ceur-ws.org/Vol-2774/paper-03.pdf
- Perevalov, A., Both, A.: Augmentation-based answer type classification of the SMART dataset. In: Proceedings of the SeMantic Answer Type prediction task (SMART) at ISWC 2020. CEUR Workshop Proceedings, vol. 2774, pp. 1–9. CEUR-WS.org (2020), http://ceur-ws.org/Vol-2774/paper-01.pdf
- Perevalov, A., Both, A.: Improving answer type classification quality through combined question answering datasets. In: Knowledge Science, Engineering and Management. pp. 191–204. Springer International Publishing, Cham (2021)
- Setty, V., Balog, K.: Semantic answer type prediction using BERT IAI at the ISWC SMART task 2020. In: Proceedings of the SeMantic Answer Type prediction task (SMART) at ISWC 2020. CEUR Workshop Proceedings, vol. 2774, pp. 10–18. CEUR-WS.org (2020), http://ceur-ws.org/Vol-2774/paper-02.pdf
- Tonon, A., Catasta, M., Prokofyev, R., Demartini, G., Aberer, K., Cudré-Mauroux, P.: Contextualized ranking of entity types based on knowledge graphs. Journal of Web Semantics **37-38**, 170–183 (2016). https://doi.org/10.1016/j.websem.2015.12.005