Semantic Answer Type Prediction by Using BERT classifier and Rule-based Ranking Strategies

Xiao Ning[1,3], Ammar Ammar[2], Arif Yilmaz[1], Shervin Mehryar[4], and Remzi Celebi[1*]

[1] Institute of Data Science, Maastricht University, Maastricht, the Netherlands
[2] Department of Bioinformatics, Maastricht University,Maastricht,the Netherlands
[3] School of Biological Science and Medical Engineering, Southeast University, China
[4] University of Toronto, Toronto, Canada
{x.ning,a.ammar,a.yilmaz,remzi.celebi}@maastrichtuniversity.nl
{shervin.mehryar}@utoronto.ca

Abstract. A key task in Question Answering (QA) is answer type prediction in which the type of the answer expected to a question expressed in natural language is predicted in order to improve overall search and retrieval performance. Answers might be of many different types as natural language is ambiguous and a question might correspond to different relevant queries. The task of predicting granular answer types for a question from a big ontology is a greater challenge due to many possible classes. In this paper, we focus on semantic answer type prediction where the candidate types come from a general-purpose ontology. We propose a model that is trained on the datasets provided for the International Semantic Web Conference (ISWC 2021) SMART Task Challenge. We model the problem as a two-stage pipeline of sequence classification tasks (answer category prediction, answer resource type prediction), each one making use of a fine-tuned BERT classifier. To cope with the highly skewed distribution of answer types in the resource category, the BERT classifier is enhanced with a rule-based ranking strategy. On the DBpedia dataset, we obtain an accuracy score of 0.985 for the answer category prediction, 0.737 of NDCG@5 and 0.702 of NDCG@10 for the answer type prediction.

Keywords: Answer type prediction · Hierarchical classification · Question answering · Semantic type prediction.

1 Introduction

With the explosive growth in the volume of online information, finding information on the web is an increasingly greater challenge for users. The need and interest in automated question answering systems will likely continue to grow with the increasing demands of users for immediate answers. An increasing number of smart devices including Apple's Siri and Microsoft's Little Ice, have embedded question answering (QA) systems which provide efficient and interactive assistance to their users. Natural language questions present a particular challenge due to the fact that the semantics are often ambiguous and highly context dependent. While a number of approaches have been proposed to deal with natural language question ambiguity and the provision of correct answers in QA systems [1], approaches such as predicting the type of expected answer by reducing the number of relevant candidates are used in practice to improve search/retrieval quality.

Currently, a modular architecture which integrates Answer Type Modeling modules that limit the subset of possible candidate answers through the use of information retrieval techniques, is embedded in most question answering systems [2]. The answer type modeling or prediction task aims to identify the type of results in order to filter out irrelevant results, which notably increases the performance of question answering systems. Generally, answers to questions can be categorized into a few basic programming data types such as *boolean*(true/false), *numeric* and *string*. A fine-grained classification of answer types would be possible when tasks are modeled to predict the semantic types from an ontology. However, the task becomes significantly more challenging when a target ontology contains a large number of types. To address this challenge, the SMART challenge dataset, which consists of questions, categories and answer types, was released by the organizers in the International Semantic Web Conference (ISWC 2020/2021) [3, 4]. The task is to provide answer categories (Task 1) and answer types (Task 2), where the answer types for "resource" category are sub-lists of ontology classes to WikiData or the DBpedia KGs.

The SMART challenge dataset provides a training set of natural language questions alongside a single given answer category (*boolean*, *literal* or *resource*) and 1-6 given answer types. Most questions in the resource category have several answer types ranging from the specific to the general, according to the subsumption hierarchy contained in the ontology. The task is then to achieve the highest accuracy for answer category and highest NDCG [5] values for answer type prediction. In this paper, we introduce our approach for identifying the answer types of a given question utilizing fine-tuned BERT classifiers. We used the ontology hierarchy of DBpedia KG to generate the general answer type and specific answer type for each question using fine-tuned BERT classifiers. We also designed a rule-based strategy to update the probabilities of answer type candidates. Experiments confirm that our approach can achieve outstanding performance. On the DBpedia dataset, we achieved a maximum accuracy score of 0.985 for the answer category prediction, 0.737 of NDCG@5 and 0.702 of NDCG@10 for the answer type prediction. Associated data, code and learned models for this work can be accessed at Github repository [5].

[5] https://github.com/xiao-nx/ISWC2021_SMART

2 Related Work

Generally, identifying the answer type is one of the key steps in a question answering system. Therefore, the dominant approach to question answering begins with building a labeled query-type dataset and then performing answer type prediction modeling to limit the subset of possible candidates [2, 6]. Recently, various approaches have been proposed to tackle this problem. For example, Abdi et al. [7] proposed an ontology-based question answering system based on an Inferring Schema Mapping (ISM) method, which uses the combination of syntactic and semantic information, and attribute-based inference. They converted the natural language queries given by users into ontological knowledge base queries, finally successfully applied it in the physics domain. Yavuz et al. [8] proposed a bidirectional LSTM model to infer answer types in conjunction with semantic parsing, which maps a natural language question into its semantic representation logical form. This representation relates to meaning stored structurally in knowledge bases by recursively computing vector representations.

The task of answer type prediction can be also seen as an extreme multi-label text classification problem where questions need to be labeled with a relevant subset of classes (e.g., from a big KG) [9]. Traditional machine learning methods and deep learning methods are two main approaches to address multi-label text classification. One of the most common way of traditional machine learning methods is to use a one-versus-all approach where a classifier per class is learned [10]. This approach suffers from i) computational complexity and ii) class imbalance when the sample size grows to a large size. In addition, many tree-based methods [11, 12] and label-embedding based methods [13, 14] have been proposed to overcome these limitations. While tree-based methods aim to produce a balanced tree structure, label-embedding based methods map labels in low-dimensional vector (embedding) space to reduce the effective label space. Label-embedding can be contextualized with the use of KG embeddings. KG embeddings have been used in several types of applications including recommendation systems and question answering [15]. The main idea behind KG embeddings is to preserve the information of the knowledge graph while representing each entity/relation as a low-dimensional vector.

Deep learning methods have achieved outstanding results in natural language processing domain, and BERT-based deep learning methods have achieved the sate-of-the-art performance in almost downstream tasks over recent years. Bidirectional Encoder Representations from Transformers (BERT) is a popular language representation model based on the Transformer model architecture, which was published in 2018 by Google AI researchers [16]. Many research works have demonstrated that fine tuned BERT can achieve state-of-the-art performance in a wide range of nature language processing tasks, including Named Entity Recognition, Question Answering and others. Kertkeidkachorn et al. [17] presented a hierarchical contextualized-based approach, which builds on top of state-of-the-art contextualized models and the hierarchical strategy to deal with the hierarchical answer types, choosing BERT to undertake the corresponding multi-class classification and multi-label classification tasks. Vinay Setty et al. [18]

proposed a two-phase solution for SMART Task, BERT for high-level category classification, and X-BERT (a variant of BERT) to model the type prediction task as an extreme multi-label text classification (XMC) problem. Their findings suggest that X-BERT for extreme multi-label classification clearly outperform retrieval-based approaches. In addition, the authors in [20] used BERT-classifier for answer type prediction, and applied a reward function based on a class hierarchy to predict resource classes. The reward function re-ranks the top class and its children obtained from the BERT-classifier to favor more specific classes (deeper classes in the hierarchy). Their method ranked 2nd in the SMART 2020 challenge and achieved an NDCG@5 of 77.7%.

3 Datasets

We have used the dataset provided by the SMART challenge for the ISWC 2021. The challenge contains questions and answer types from two ontologies: DBpedia and Wikidata. Each dataset is structured as JSON format and labeled with classes of the target ontology (i.e. DBpedia or Wikidata ontology). Here, we focused on only the DBpedia dataset. The train dataset in DBpedia contains 43,554 questions with the categories being 36,886 *resource*, 4,530 *literal* and 2,138 *boolean* questions. Each sample question has a identifier, text in English, an answer category and several answer types. Answer category takes just one target value for one question and answer type could be a list of types where types are ordered according to the level of DBpedia ontology hierarchy. For the *boolean* category questions, the answer type is always boolean. If the category of a question is literal, then the answer type will be either *number*, *string* or *date*. The questions with *resource* categories are labeled with a list of fine-grained classes from the DBpedia ontology(∼760 classes), the relations among answer types are organized hierarchically as shown in Figure 1. Considering a question with the following list of answer types (”dbo:Location”, ”dbo:Place”, ”dbo:PopulatePlace”, ”dbo:Place”, ”dbo:Settlement”, ”dbo:City” and ”dbo:Capital”), the most general type would be ”dbo:Location” and most specific type would be ”dbo:Capital” according to the DBpedia class hierarchy. The class distribution of the DBpedia dataset, which is shown Figure 2 has a long-tail distribution.

4 Methodology

We propose a two-stage workflow to address the Semantic Answer Type Prediction Task (SMART), demonstrated in Figure 3. The workflow starts with building a classifier that predicts the category of the question (Referred as Task 1 in challenge). For prediction of the question categories, we model the problem as a multi-class classification problem to predict one of the extended categories (*boolean, number, string, date, and resource*) from question text by fine-tuning the BERT classifier. This is process is threefold: 1) If the classifier returns ”boolean” for the category then the answer type is returned as *boolean*;

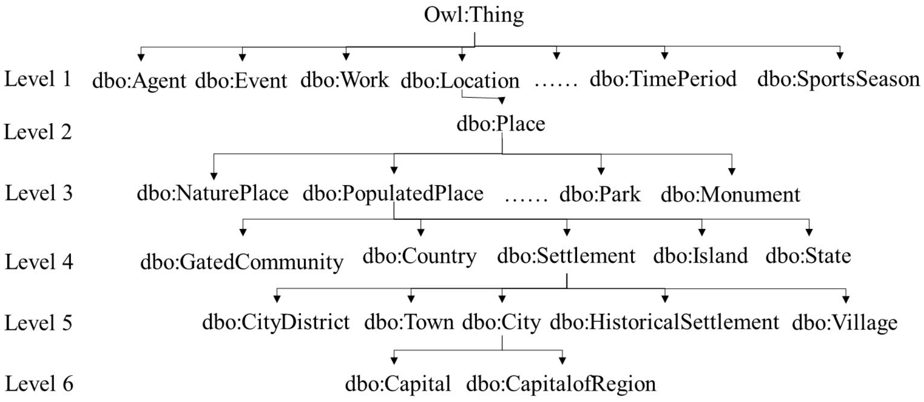

Fig. 1. Hierarchical answer types in "resource" category

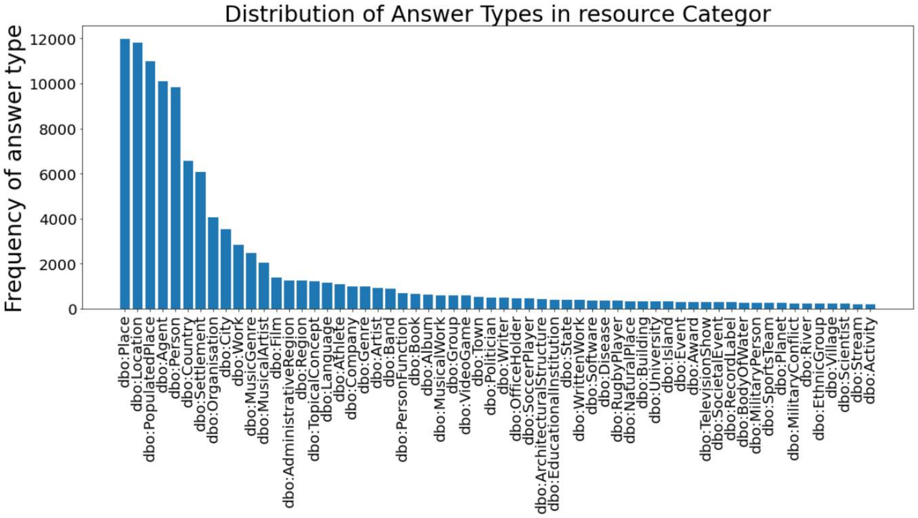

Fig. 2. Answer type distribution for DBPedia dataset. (Types more than 200 samples.)

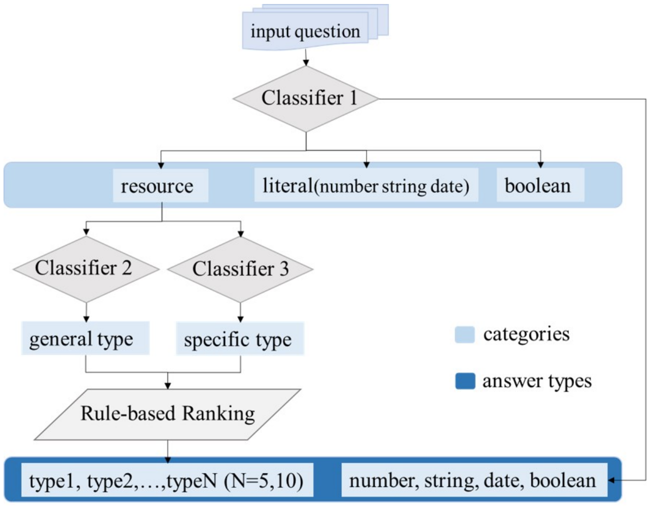

Fig. 3. Workflow for answer type prediction of our proposed method

2) if the classifier predicts "string", "date" or "numeric" , then these will be used as predicted answer types and the category will be set to *literal*; 3) finally if it returns "resource", then we apply a rule-based ranking which combines another two classifiers to predict fine-grained types (Task 2).

We use two BERT-based classifiers to predict general types and specific types for each question in the resource category, respectively. A rule-based ranking strategy that combines the predictions from both classifiers is employed. Finally, we output the top k(k=10) answer types with the highest scores from the candidate set as the final result.

4.1 Answer category prediction

In this paper, we consider following five categories as high-level categories: *boolean, number, string, date*, and *resource*. Since in the SMART challenge dataset, a question can belong to one of the three categories: *boolean, literal*, and *resource*. As *boolean* category questions are referred to as confirmation questions due to the fact that only 'yes' or 'no' is given as an answer type, so there is no further classification for this category of questions. *Literal* questions can be a *number, string* or *date* type.

To identify question categories, we fine-tune a BERT model using the Hugging Face PyTorch implementation [20]. The reason for choosing the BERT model is that BERT has shown outstanding performance on many text classification tasks.

4.2 Resource Answer Type Prediction

The prediction of the answer type of questions in the resource category is a more fine-grained (and thus more challenging) classification problem, because of the large number of types that a question can be classified to. Thus, it is not effective to train a classifier on all the ontology classes. It is well known that most questions in the resource category have several answer types ranging from the specific to the general, according to the semantic hierarchy of the ontology. Due to the large number of possible labels, we used the ontology hierarchy of DBpedia KG to reduce the number of possible types to the most general answer types and the most specific answer types for each question and to capture the hierarchical relationship between answers. We also used BERT to train the classifiers for top-level types (general sub-types) and bottom-level types (specific sub-types). There are 30 classes used in the general type classification task and 287 classes used in the specific type classification task, with corresponding accuracy scores of 0.979 and 0.889 for each classifier on the validation set.

4.3 Rule-based Ranking

We ensemble the two BERT models by combining the predictions made for each question (generic and specific answer types) and the corresponding probabilities. The goal is to increase the accuracy of the specific answer type classifier by incorporating the output of the general type classifier. To this end, we designed two ranking rules:

Rule 1: Boost the probability of the predicted type that lie below the top type. Specifically, the degree of boost to probability of each type is measured by the depth of the type in the hierarchy. The formula for updating the score of a specific type is:

$$score(s) \leftarrow p(s) + p(g) * \frac{d_c}{d_{max}}, \tag{1}$$

where s denotes the specific answer type, g denotes the generic answer type, $p(g)$ and $p(s)$ represent the probability of the predicted generic answer type and specific answer type respectively, d_c is the depth of class c in the hierarchy, while d_{max} is the maximum depth of the ontology (6 for DBpedia). This means that, after applying normalization and adding the probability on the output of the model, the top class can be a sub-class that was originally ranked below a more general class.

Rule 2: We assume that if any two predicted answer types are sub-classes of a parent type, the parent type should be included in the answer list. If two types in top N highest prediction probability share a parent type, we will consider the parent type as one of the answer type, then add this to the list of answer types and calculate the probability of the parent type with the following formula

$$score(s_p) \leftarrow \frac{p(s_1) + p(s_2)}{d_{max}}, \tag{2}$$

where s_p denotes the parent type, $p(s_1)$ and $p(s_2)$ represent probabilities of predicted answer type respectively. This relation combines every pair of answer types in the predicted specific answer type set.

5 Experiments

5.1 Settings

We implement the contextualized word embedding-based BERT model by using the Hugging Face repository. We randomly split the dataset into three parts, 80% for training model, 10% for model validation, 10% for test and error analysis. Next, we manually tune the hyper-parameters then test on the validation set to find a reasonable set of hyper-parameters. In addition, we set the hyperparameters as follows: batch size: 32, learning rate: 5e-5, optimizer: Adam, epochs: 5. Finally, according to the performance on validation dataset, we pick the best classifiers in 5 epochs and combine them as final model, and apply the final model in predicting the test dataset of the SMART challenge and submit the results.

5.2 Evaluations

We adopt the following evaluation metrics. One key performance metric is the accuracy score, which is the percentage of questions that have been classified in the correct category. To evaluate type answer classification models, Lenient Normalized Discounted Cumulative Gain (NDCG@N) metric with a Linear decay [5] is employed. Specifically, only one predicted answer type in literal category can be either correct or incorrect. For a ranked list of top-N predicted answer types in resource category, NDCG will give 0 if none of the predicted answer types are in ground truth answer types, and otherwise $1 - d(t, t_g)/h$, where h represents the maximum depth of the type hierarchy, $d(,)$ is the distance between the predicted answer types t and t_g is the closest matching ground truth answer types in the type hierarchy.

6 Results

The results in Table 1 show that the fine-tuned BERT models perform with high accuracy for category classification. We hypothesize that due to the clear patterns which the models can learn, the high-level category classification is a fairly easy task. However, most mistakes occur for the resource category, which is the majority category in both datasets.

We have analyzed the errors made by our approach. First, we look at resource types where most errors occur. In Table 2, we show anecdotal examples of the mistakes made by our approach. The table lists the types found in the gold labels for the questions and the types predicted by the classifier. Most of these errors are due to irrelevant types returned in the result list. In several cases,

Table 1. Evaluation results on final test dataset

Metrics	DBpedia
Accuracy (category prediction: boolean, literal, resource)	0.985
Lenient NDCG@5 with linear decay (literal/resource type prediction)	0.737
Lenient NDCG@10 with linear decay (literal/resource type prediction)	0.702

the predicted labels contain the ground truth labels but place them at lower ranks, which affects the NDCG scores. In some cases, the predicted labels are appropriate, even they do not exactly match the gold labels. For example, the last question in Table 2 has only one ground-truth label that is not predicted by our model, but Agent and Person types in the prediction list are more likely to be correct types.

Table 2. Example questions from DBpedia with respective ground truth and predicted labels

question	Ground Truth	Predicted Labels
What is the significance of artists of The Beatles' Story?	['dbo:Single' 'dbo:MusicalWork' 'dbo:Work']	['dbo:Single' 'dbo:Work' 'dbo:Person' 'dbo:MusicalArtist' 'dbo:Award' 'dbo:Writer' 'dbo:TelevisionShow' 'dbo:Album' 'dbo:Deity' 'dbo:RugbyPlayer']
Name the islands that belong to the archipelago whose largest city is Papeete?	['dbo:Single' 'dbo:MusicalWork' 'dbo:Work']	['dbo:Location' 'dbo:Country' 'dbo:Mountain' 'dbo:State' 'dbo:Church' 'dbo:City' 'dbo:Lake' 'dbo:Village' 'dbo:River' 'dbo:Ocean']
Who did Laszlo Papp lose to?	['dbo:Activity']	['dbo:Agent' 'dbo:Person' 'dbo:FormulaOneRacer' 'dbo:Organisation' 'dbo:AmericanFootballPlayer' 'dbo:SoccerPlayer' 'dbo:HockeyTeam' 'dbo:SoccerClub' 'dbo:BaseballTeam' 'dbo:Writer']

7 Conclusions

We proposed a novel two-stage solution for SMART challenge of ISWC 2021, in which we model the problem as a set of sequence classification tasks, each one making use of a fine-tuned BERT classifier. Our two-stage solution shows a satisfactory performance and fine-tuning BERT can achieve competitive results than other classifiers. For the more fine-grained problem of answer resource type prediction (thus more challenging as the classes can be hundreds or thousands), we have proposed the enrichment of the BERT model with rule-based ranking strategies that consider the hierarchy of the ontology classes, favoring the more specific classes that are in the bottom of the DBpedia class hierarchy. The evaluation results demonstrated that the performance of the proposed method achieves 0.985 accuracy in predicting general answer category. The method scores 0.737 of NCDG@5 and 0.702 of NCDG@10 in recommending correct answer types for questions in the resource category. Our results suggest that the proposed ensemble method can predict answer types with a high accuracy by utilizing the underlying hierarchical relationship in the target ontology.

References

1. Kodra, L.,E.K. Mece: Question Answering Systems: A Review on Present Developments, Challenges and Trends. International Journal of Advanced Computer Science and Applications,2017.8(9): p.217-224.
2. Oleksandr Kolomiyets and Marie-Francine Moens, A survey on question answering technology from an information retrieval perspective. Information Sciences, 2011.
3. Mihindukulasooriya,N., et al., SeMantic AnsweR Type prediction task (SMART) at ISWC 2020 Semantic Web Challenge. arXiv pre-print server, 2020.
4. Mihindukulasooriya, Nandana and Dubey, Mohnish, Semantic Answer Type and Relation Prediction Task (SMART 2021), arXiv, 2022
5. Balog, Krisztian and Neumayer, Robert, Hierarchical Target Type Identification for Entity-Oriented Queries. Proceedings of the 21st ACM International Conference on Information and Knowledge Management, 2012
6. Huang, Zhen and Xu, Shiyi and et al., Recent Trends in Deep Learning Based Open-Domain Textual Question Answering Systems. IEEE Access, 2020.
7. Abdi, A., N. Idris, and Z. Ahmad, QAPD: an ontology-based question answering system in the physics domain. Soft Computing, 2018. 22(1): p. 213-230.
8. Yavuz, S., et al. Improving Semantic Parsing via Answer Type Inference. Association for Computational Linguistics.
9. Fedden, S. and G.G. Corbett, Extreme classification. Cognitive Linguistics, 2019.
10. Babbar, R. and B. Schölkopf, DiSMEC: Distributed Sparse Machines for Extreme Multi-label Classification. Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, 2017.
11. Bengio, S., Label embedding trees for multi-class tasks. 2010.
12. Prabhu, Y. and M. Varma, FastXML: A fast, accurate and stable tree-classifier for extreme multi-label learning. 2014: FastXML: A fast, accurate and stable tree-classifier for extreme multi-label learning.
13. Bhatia, K., et al. Sparse Local Embeddings for Extreme Multi-label Classification. in 29th Annual Conference on Neural Information Processing Systems (NIPS). 2015. Montreal, CANADA.
14. Yu, M. and M. Dredze. Improving Lexical Embeddings with Semantic Knowledge. 2014. Association for Computational Linguistics.
15. Huang, X., et al., Knowledge Graph Embedding Based Question Answering. Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, 2019.
16. Devlin, J., et al., BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv pre-print server, 2019.
17. Kertkeidkachorn, N., et al. Hierarchical Contextualized Representation Models for Answer Type Prediction. in SMART@ISWC. 2020.
18. Setty, V. and K. Balog, Semantic Answer Type Prediction using BERT: IAI at the ISWC SMART Task 2020. arXiv pre-print server, 2021.
19. Nikas, C., P. Fafalios, and Y. Tzitzikas. Two-stage Semantic Answer Type Prediction for Question Answering using BERT and Class-Specificity Rewarding. in SMART@ ISWC. 2020.
20. Wolf, T., et al. Transformers: State-of-the-art natural language processing. in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. 2020.
21. Balog, K. and R. Neumayer, Hierarchical target type identification for entity-oriented queries. Proceedings of the 21st ACM international conference on Information and knowledge management, 2012.