KBQA Final Report

Mengshi Ma

DICE group at University Paderborn

1 Task Definition

In the "Knowledge-based Question Answering"project group, we aim to develop a question answering system, which receives a natural language question from a user, then responses with the answer of this question based on the knowledge from DBpedia.

In order to answer a question, the first step is to convert the question to a sparql query, with which the knowledge in DBpedia can be queried as a relational database. Once we get a response from the DBpedia endpoint, the answer is extracted and presented to the user.

2 Approach Description

3 Evaluation

We used gerbil for evaluation. Gerbil can evaluate QA systems on a qald data set automatically, computing micro precision, recall and F1 score, macro precision, recall and F1 score, and F1 QALD score. Additionally, the average answering time is also calculated.

We focused on qald 8 and qald 9 data sets, trained and evaluated on them. The questions in qald 9 are more complicated than in qald 8, e.g. multiple triples and more logic. Also, qald 9 data set contains more questions. Therefore, qald 9 is more challenging than qald 8 for evaluation.

Following table shows our evaluation results from approach A during the Project Group:

QALD-8					
Date	**Model**	**Precision**	**Recall**	**F1**	**F1 QALD**
06.12	NSpM	0	0	0	0
20.12	NSpM	0.0244	0.0244	0.0244	0.0476
23.01	NSpM_SL	0.1707	0.1626	0.1602	0.2777
14.06	NSpM_PSL	0.2561	0.2683	0.2602	0.4149
27.06	NSpM_LCPSL	0.3171	0.3415	**0.3252**	**0.5025**
	Tebaqa	0.4756	0.4878	0.4797	0.556

QALD-9					
Date	**Model**	**Precision**	**Recall**	**F1**	**F1 QALD**
23.01	NSpM_SL	0.1299	0.1344	0.1312	0.2362
14.06	NSpM_PSL	0.2479	0.2694	**0.2454**	**0.4127**
27.06	NSpM_LCPSL	0.2283	0.2464	0.2237	0.3845
	Tebaqa	0.2413	0.2452	0.2384	0.3741

All scores are macro scores.

At the beginning of the Project group, in order to try out the original NSpM model and gerbil evaluation, we trained with qald 8 train data set for 8 epochs. We also implemented a python script to simulate receiving natural language question, converting it to query, sending request to DBpedia endpoint, and finally generating a data set with answers in qald format, since we did not have an URL endpoint for evaluation until that time.

After figuring out the functions of each component, we trained NSpM again for more epochs. This time, the evaluation result was not zero anymore, which proved that, the NSpM is a fesible approach.

By inspecting the sparql queries in train and test data set, we noticed that many entities appear in test data set are not in train data set. Therefore, we integrated DBpedia spotlight to avoid this problem. With DBpedia spotlight and more training data, the result has been greatly improved.

For a better conversion from natural language question to sparql query, we used hugging face Pegasus model instead of the simple encoder and decoder neural network in the original NSpM.

At the end of the project group, we tried pre-training on LC-QALD data set, then fine-tuning on qald 8 or qald 9. Interestingly, the score for qald 8 increases with pre-training, but for qald 9 decreases. However, we did not enough time to figure out the reason behind it.

Compare to TeBaQa [1], our approach overperforms TeBaQa on qald 9 and performs close to TeBaQa on qald 8.

4 Learned Skills

During this two semester project group, I have learned a lot in all aspects.

We used Scrum to manage our team, therefore, I have gathered some experience with Scrum, e.g. How to be a Scrum master, how to manage weekly tasks with a board, and what to do in meetings. Also, the most important thing is how to work as a team, since a usually worked alone.

I have also improved my coding style in this project. Thanks to the linter, I wrote more readable code and never forgot to add comments on each function, which make it easier for others and me later to understand.

I have also formed good habits in git, committing frequently and with meaningful comments. I practiced git functions such as new branch, merge, rebase, and also how to handle merge conflicts. Additionally, I have learned how to set up CI/CD in Github from my teammates, which simplified our deployment.

This is my first time working with network on a virtual machine with nginx, REST and docker. I think this experience can be applied on many other projects.

I got my first insight into knowledge graph in this project group and knew how sparql query is written.

The last but not least, I learned much about natural language processing in praxis, including encoder and decoder, tokenizer, model training. Moreover, I had some experience in building our own data set for a NLP model.

5 Issues

During this project group, I have met many issues, but with the help of our teammates, they were solved quickly.

At first, I did not know what to do with our virtual machine. I asked my teammates then figured out how to deploy our software on it.

In our evaluation at beginning, we must create our data set in qald format with answer and upload it to gerbil, since the system was not deployed on VM at that time. We created a script to do it automatically.

Inspired by templates and generator in the original NSpM model, we used templates, questions and queries with placeholders, for training directly. For training on qald 8 and 9, we did not have a data set suitable for our approach. Thus, we implemented a script to replace entities in qald 8 and 9 with placeholders and convert to the format we need. In order to include more predicates in our data set, we also checked classes in DBpedia and wrote questions and queries on our own.

While training our model on felis Server, I encountered some issues with incompatible cuda and driver version. This was fixed quickly after some research.

At the beginning of the second semester, we found Convolutional Sequence-to-Sequence model (ConvS2S), which is written in tensorflow version 1. There is a big difference between Tensorflow version 1 and 2, we have spent a lot of time to rewrite it in tensorflow two and add additional features that we want. However, after the implementation we trained ConvS2S with our questions and queries, it could not predict any query. We tried a lot to fix this model, but it did not work in the end and we switch to Pegasus model.

At the end of project group, we wanted to pre-train on LC-QALD and fine-tune on qald 8 and 9, which could not be done directly. After checking parameters in Pegasus model, we found a way to do it.

6 Self-evaluation

References

1. Vollmers, D., Jalota, R., Moussallem, D., Topiwala, H., Ngomo, A.N., Usbeck, R.: Knowledge graph question answering using graph-pattern isomorphism. CoRR **abs/2103.06752** (2021), URL `https://arxiv.org/abs/2103.06752`