Machine Translation: English - SPARQL

Anand Panchbhai, anandp@iitbhilai.ac.in

I. Abstract

This document contains all the results obtained till now during the span of this project.

II. Introduction

The document is divided into 3 parts, first the data generation methodology is explained, followed by results and conclusion.

To the best of my knowledge no such work dealing with detailed analysis of the nature of Neural Machine translation with carefully curated dataset has been done. The novel feature of this paper are:

1) A new template generation methodology
2) A new ranking mechanism to determing which generated templates are more popular and subject to be asked more frequently.
3) A detailed analysis of the model performance as the datasets were tweaked to better understand the capabilities of variour NMT models.
4) A more controlled test set for checking the extent of understanding that the NMT models are capable of reaching.

III. Data generation

To give you a sense of the data set, here is an example of the composition we are working on:

```
Test: BoA: What is the division of family
      of parthenium incanum ?
Query: select ?x where{<http://dbpedia.org/
resource/Parthenium_incanum> dbo:family ?x1
. ?x1 dbo:division ?x }
```

```
Train: A: What is the family of parthenium
incanum ?
Query: select ?x where{<http://dbpedia.org/
resource/Parthenium_incanum> dbo:family ?x }
```

```
Train: B: What is the division of species of
kale ?
Query: select ?x where{<http://dbpedia.org/
resource/Kale> dbo:species ?x1 . ?x1
dbo:division ?x }
```

This is a typical composition setup this paper works on: A,B were introduced in completely different context, Testa are done on AoB or BoA. Here in this context. A and B represent the ontology class entities

These points were considered while generating the train and test sets:

1) The train and test set were made with a compositionality depth of 2, which means questions like:
   Depth 1:

   $$What\ is\ the < Property1 > of < A >?$$

   Depth 2:

   $$What\ is\ the < Property1 > of < Property2 > of < A >?$$

2) Apart from what based question **when, number of, who etc.** based questions were also made using a template generation protocol.
3) For the special test set, the templates were qualified to be in the test set if:
   a) The entity used was once added to the the train set at a given depth.
   b) i.e. If the same entity was encountered again in the same depth scenario as the first encounter, it was added to the test set in the subsequent encounters.
   Train: What is the abbreviation of family of $< A >$?
   Test: What is the abbreviation of species of $< A >$?
   c) In the example above abbreviationw as once added with depth 2 in the train set, so in the subsequent sets, it was added in the test set. But as you can see the correspinding entity in the depth 1 is different in both the scenarios i.e. family and species respectively.
4) To ensure that enough examples were present to train and test the model, the proposed ranking methodology was not used to eliminate unpopular/unnatural questions.

IV. Results

This section contains the results of the notable experiments carried out during the investigations. Before moving on to the results of the experiment, lets first look at the experimental setup used, the information is present in table 1.

The source code of the NMT model used can be found at https://github.com/tensorflow/nmt.

Theses steps were applied to the datasets to understand the change in the overall performance of the model as the dataset evolved:

Model	Attention Mechanism	Train Steps	Number of layers	Dropout	Metrics
NMT	NIL	30,000	2	0.2	BLEU, Accuracy
NMT + Attention	scaled_loung	40,000	2	0.2	BLEU, Accuracy

TABLE I
EXPERIMENTAL SETUP

Model	Differentiator	NMT (BLEU | Accuracy)	NMT + Attention (BLEU | Accuracy)
Test 1	General	97.69 | 89.75	97.3| 88.0 (16,000 iterations — Plateau)
Test 2	Separate test	39.65 | 0	43.3|0 (16,000 iterations — Plateau)
Test 3	Separate test + Same Vocab	62.76 | 0.42	NIL
Test 4	Separate test + Same Vocab + Frequency thresholding	66.39 | 5.71	85.16 | 34.29

TABLE II
EXPERIMENTAL RESULT | EUKARYOTES | BEST PERFORMANCE ON CORRESPONDING TEST SETS

General:The dataset was generated with no special test set, a single train set was generated which was then randomly distributed in the following ratio to train, dev and test: 80:10:10.

Separate test : As per the datageneration guidelilne mentioned above a separate test set was created to test the model performnace specifically on compositionality. The vocab may or may not contain all the elements of the test set.

Same Vocab : The test set was further augmented to ensure only those entities were present in the test set whose vocabulary was also present in the train set. No further elemination based on any threshold was done here. (train contained all templates not in the test set)

Frequency thresholding : The test set was further augmented to ensure that each entities in test set were present specific number of times in the train set. Ensuring proper learning of the entities.

A. Inferences

The results are presented in a tabulated in table 2.

1) (Test 1)The performance of the NMT model on the general sets which din't involve a special test set was very good. Infact the model was used in a separate partal for testing on the question on eukaryotes and performed well.
2) (Test 2) On the more carefullt curated special test set, it was to be note that all the questions were composition question with depth 2. The performance in the test 2 dropped drastically with a lot of <UNK> and wrong entity detection in the output of the test set.
3) (Test 3) In test 3, we tried to fix the problem of OOV by making sure that the vocabulary of the test and train set matched and that each vord in the vocabulary was present atleast once in the train set. We could still some occurences of <UNK> and wrong entitites in the test output.
4) (Test 4)To further make the test set more relevant to the training being done, we added a frequency thresholding which removed all entities with frequency less than 7 in the training set.
5) The model performance increased a bit with accuracy going above 1 for the first time on the non-attention model.
6) The last test was done on NMT model with attention, a dratic increase in performance was witnessed. The major reason might be that the questions in the test set are longer and attention mechanism in general are better at retaining information.
7) A very important aspect of attention model that I observed was that the model convergence was very fast compared to calssic LSTM based NMT models.

V. CONCLUSION

1) The given paper gave a brief information about the experimental setup and the results obtained in the work done in the project till now.
2) We can further look into LC-QuAD and DBNQA to get more question templates using the generalized templated creation mechanism proposed in this paper. (For 1 subject entity)
3) I believe that the putting all the mentioned filtering methodologies together and building a model on complete dbpedia might yield us a good complete product. Like the Eukaryotes based Q and A portal developed earlier as part of the project.