Joint model for intent and entity recognition

Petr Lorenc

Dept. of Cybernetics, Czech Technical University, Technická 2, 166 27 Praha, Czech Republic

petr.lorenc@cvut.cz

Abstract. *The semantic understanding of natural dialogues composes of several parts. Some of them, like intent classification and entity detection, have a crucial role in deciding the next steps in handling user input. Handling each task as an individual problem can be wasting of training resources, and also each problem can benefit from each other. This paper tackles these problems as one. Our new model, which combine intent and entity recognition into one system, is achieving better metrics in both tasks with lower training requirements than solving each task separately. We also optimize the model based on the inputs.*

Keywords

intent classification, entity recognition, machine learning, natural language processing, word embeddings

1. Introduction

In recent years, a significant amount of smart speakers have been deployed and achieved great success, such as Amazon Echo, Google Home, and many others, which interact with the user through voice interactions. Natural language understanding (NLU) is critical to the performance of spoken dialogue systems. NLU typically includes many parts, typically based on the usage, but the core usually contains the intent classification and entity recognition. For example, we have a sentence 'I would like to move to London.' We want to extract the information that the intent is 'change_address' and also recognize that the entity is 'London'. We can use this information to query a database or other knowledge resource. In some cases, for example, in chatbot system Alquist[1], we require to catch not only the valid named entities defined here [5] but also pseudo-entities. An excellent example of pseudo-entity is 'rock music' in a sentence: 'Let's listen to rock music.'

2. Related work

Deep learning models have been extensively explored in NLU for several years. The traditional way to deal with intent classification and named entity recognition is to encounter both tasks separately. Named entity recognition has been thoroughly studied in [5], which can be used as a source of other resources, and intent classification is the topic of [6].

In recent years, the multi-tasks models start occurring. The promising result is shown in [8], where the authors present the model called ERNIE, which is trained using several tasks as knowledge masking or prediction distance between sentences. In the work [1], we can see a model constructed for joint slot-filling and intent classification. In general, the task of slot-filling is similar to named entity recognition. For example query: **What flights are available from pittsburgh to baltimore on thursday morning** has intent **flight info** and following slots:

- from_city: pittsburgh
- to_city: baltimore
- depart_date: thursday
- depart_time: morning

Nevertheless, we can approach the slots as entities. They presented results based on recurrent neural networks and softmax-based attentions mechanism. Based on that reasoning we will be using terms *named entity recognition* and *slot-filling* interchangeably.

The work [2] also focused on slot-filling and intent classification but in comparison with [1] are using sparse attention mechanism. The sparse constraint assigns bigger weights for important words and lower the weights or even totally ignores the less meaningful words such as “the” or “a”.

The more recent work [3] is also focusing on slot filling. The authors were using the novel approach for getting contextualized embeddings called BERT. We will also use other types of word embeddings, namely glove and fastext, which are commonly used for dealing with text input, see [9] and [10].

To compare our results with [1], [2] and [3], we will be working with ATIS dataset described below.

[1] http://alquistai.com/

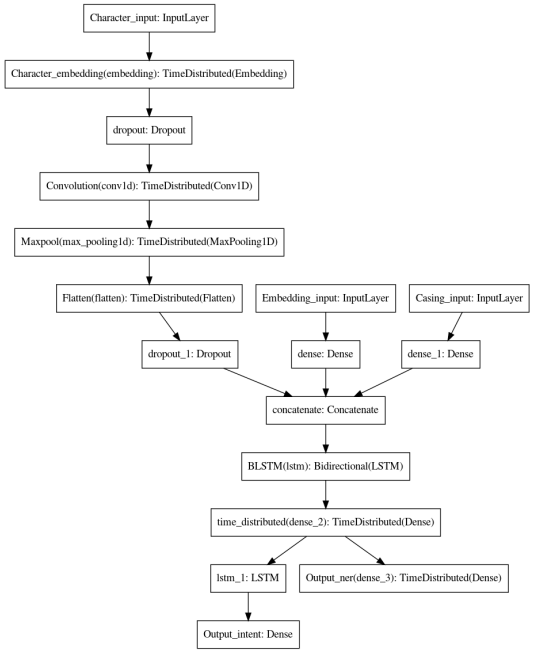

Fig. 1. Model 1 of joint entity recognition and intent classification

3. Model architecture

Based on [7] and [3] we design a model shown in Figure 1. The novel approach of the model is that it uses the convolutional neural network (CNN) over char-based embeddings. It also uses features extracted from each word (such as if the word is numeric or start on lower letter) and word embedding. The architecture is word-embedding agnostic so that we can measure the metrics on different word embedding without a significant change in code. The model is also prepared for changing the architecture of the internal neural network. We have tried two types of further processing (aka types of neural networks): bidirectional Long-shot term memory[13] (LSTM) and neural network distributed through time. The difference is that LSTM is using context over time, but in the neural network distributed through time we are using the same network at each timestep.

To find the influence of word embeddings on our result, we used three types of word representation as a vector of floats:

- Glove[9]
- Fasttext[10]
- Bert[3]

All types of word embeddings were not fine-tuned because of possibly misleading the results (glove and fasttext cannot be fine-tuned). All files for creating embeddings are publicly available, and links are provided in github page of this project.[2]

[2]https://github.com/petrLorenc/poster2020

4. Dataset

The dataset was created by the team in Microsoft and is publicly available[3]. The dataset (described in Table 1) contains spoken utterances classified into one of 26 intents. Each token in a query utterance is aligned with IOB labels. Primary, the dataset is used for intent recognition and slot filling, but as stated above, the tasks of slot filling and entity recognition are interchangeable.

Name	# Examples
Training Set - Intent	***4 978***
Training Set - NER	***20 426***
Test Set - Intent	***893***
Test Set - NER	***3 665***

Tab. 1. ATIS dataset

5. Results

Based on model shown in 3 we measure F1 score for NER and accuracy for intent classification on data shown in Table 1. These metrics were chosen because other studies are using them. The results are shown for all three types of embeddings. The results can be seen in Table 2 and 3.

Embeddings	NER F1 accuracy
BERT	**95.78**
fasttext	**96.83**
glove	**98.44**

Tab. 2. Results on ATIS dataset - model 1

Embeddings	Intent F1 accuracy
BERT	**92.45**
fasttext	**90.70**
glove	**95.74**

Tab. 3. Results on ATIS dataset - model 1

The tables showed that Glove embeddings (the eldest approach for getting word embeddings) are getting the best results. A technique like Dropout is used in the model to avoid overfitting. Another technique was to stop training when the validation loss is not decreasing for more than two epochs. The train and validation loss are shown in Figure 3. The assumption for a reason for that is because Bert should

[3]Available at github.com/Microsoft /CNTK/tree/master/Examples/LanguageUnderstanding/ATIS

Fig. 2. Model 2 of joint entity recognition and intent classification

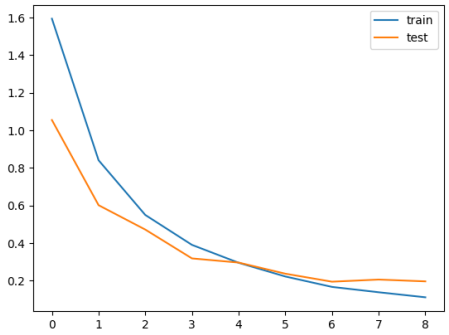

Fig. 3. Train and validation loss of Glove model 1

be fine-tuned, and it already has the context information in it, so the use of another context-aware neural network (like LSTM) is very useless and even adding the noise. To test that we used another network architecture, shown in Figure 2.

The results for the model 2 can be seen in Table 4 and 5.

The result confirms our hypothesis that the LSTM is making classification problem harder when using BERT embeddings.

The overall result is that the joint model is performing very well on these tasks with preserving or even improving the metrics on NER and intent classification tasks. For a better comparison of results, we can look at tables 6 and 7 where we compare our algorithm with approaches from other researchers. We also include the results, which can be shown at Table 10, of the best single-task model for intent classification measured on ATIS dataset. The data for NER

Embeddings	NER F1 accuracy
BERT	94.31
fasttext	88.26
glove	88.20

Tab. 4. Results on ATIS dataset - model 2

Embeddings	Intent F1 accuracy
BERT	92.14
fasttext	92.10
glove	89.68

Tab. 5. Results on ATIS dataset - model 2

on this dataset is not available. In table 8 we can see the average number of parameters of our joint models. We also measured average training time per epoch (the overall time wasn't comparable because of early stopping criteria - joint model 2 was trained for 7 epochs, the joint model 1 and only intent model was trained for 9 epochs and only ner model was trained for 12 epochs). The results of time can be seen in Table 9. The computation was performed on 1xK80 GPU instance - ml.p2.xlarge - Amazon SageMaker ML Instance[4].

Algorithm	Slot/NER F1 score
Our Glove model 1	98.44
Approach in [4]	95.90
Approach in [1]	95.87
Approach in [7]	96.89

Tab. 6. Comparison on ATIS dataset

5.1. Future work

Based on [1] and [2] with their models, which are using the attention neural network, we propose to explore the possibility of improving our model with the attention mechanism.

6. Conclusion

In this paper, we explored strategies in utilizing a joint model for named entity recognition and intent classification. The model is preserving the intent classification accuracy score and also got great results in the F1 score on NER. We compare the results of our joint model with published results

[4] https://aws.amazon.com/sagemaker/

Algorithm	Intent accuracy
Our Glove model 1	95.74
Approach in [4]	96.90
Approach in [1]	98.43
Approach in [7]	98.99

Tab. 7. Comparison on ATIS dataset 2

ModelEmbeddings	# parameters
Glove/Model 1	2,514,066
Fasttext/Model 1	2,514,066
BERT/Model 1	2,884,754
Glove/Model 1 - Only Intent	2,480,656
Glove/Model 1 - Only NER	1,982,072
Glove/Model 2	724,882
Fasttext/Model 2	724,882
BERT/Model 2	910,226

Tab. 8. Number of parameters

in the scientific papers to get the result which we can compare. The joint model is found to be more efficient with almost the same number of parameters and requires less computational resources in comparison with separated models for each task. We also explore the importance of word embeddings on our tasks.

7. Acknowledgements

The research described in the paper was supervised by Ing. Jan Šedivý, CSc. (CIIRC CTU in Prague) and supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS20/092/OHK3/1T/37

References

[1] B. LIU, I. LANE, *Attention-Based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling*, 2016, Available at *https://arxiv.org/pdf/1609.01454.pdf* [Online; accessed 21-February-2020]

[2] B. LIU, I. LANE, *Jointly Trained Sequential Labeling and Classification by Sparse Attention Neural Networks*, 2017, Available at *https://arxiv.org/pdf/1709.10191.pdf* [Online; accessed 22-February-2020]

[3] Q. CHEN, Z. ZHUO, W. WANG, *BERT for Joint Intent Classification and Slot Filling*, 2019, Available at *https://arxiv.org/pdf/1902.10909.pdf* [Online; accessed 15-February-2020]

ModelEmbeddings	Time
Glove/Model 1	23s
Glove/Model 1 - Only Intent	22s
Glove/Model 1 - Only NER	20s
Glove/Model 2	8s

Tab. 9. Training times

Algorithm	Intent accuracy
Approach in [11]	95.66
Approach in [12]	95.25

Tab. 10. Intent classification task

[4] L. QIN, W. CHE, Y. LI, H. WEN, T. LIU, *A Stack-Propagation Framework with Token-Level Intent Detection for Spoken Language Understanding*, 2019, Available at *https://arxiv.org/pdf/1909.02188.pdf* [Online; accessed 2-February-2020]

[5] D. NADEAU, S. SEKINE, *A survey of named entity recognition and classification [online]*, 2007, Available at *https://nlp.cs.nyu.edu/sekine/papers/li07.pdf* [Online; accessed 21-November-2019]

[6] J. ZHONG, W. LI , *Prediction customer call intent by analyzing phone call transcript based on CNN for multi-class classification [online]*, 2019, Available at *https://arxiv.org/pdf/1907.03715.pdf* [Online; accessed 21-January-2020]

[7] Y.WANG, Y. SHEN, H. JIN *A Bi-model based RNN Semantic Frame Parsing Model for Intent Detection and Slot Filling*, 2018, Available at *https://arxiv.org/abs/1812.10235* [Online; accessed 2-February-2020]

[8] Z. ZHANG, X. HAN, Z. LIU, X. JIANG, M. SUN, Q LIU *ERNIE: Enhanced Language Representation with Informative Entities*, 2019, Available at *https://arxiv.org/abs/1905.07129* [Online; accessed 2-February-2020]

[9] J. PENNINGTON, R. SOCHER, CH. D. MANNING *GloVe: Global Vectors for Word Representation*, 2014, Available at *https://nlp.stanford.edu/pubs/glove.pdf* [Online; accessed 2-February-2020]

[10] E. GRAVE, P. BOJANOWSKI, P. GUPTA, A. JOULIN, T. MMIKOLOV *Learning Word Vectors for 157 Languages*, 2018, Available at *https://arxiv.org/abs/1802.06893* [Online; accessed 2-February-2020]

[11] G. KURATA, B. XIANG, B. ZHOU, M. YU *Leveraging Sentence-level Information with Encoder LSTM for Semantic Slot Filling* , 2016, Available at *https://arxiv.org/abs/1601.01530* [Online; accessed 2-February-2020]

[12] B. PENG, K. YAO *Recurrent Neural Networks with External Memory for Language Understanding*, 2015, Available at *https://arxiv.org/abs/1506.00195* [Online; accessed 2-February-2020]

[13] S. HOCHREITER, J. SCHMIDHUBER *LONG SHORT-TERM MEMORY*, 1997, Available at *https://www.bioinf.jku.at/publications/older/2604.pdf* [Online; accessed 2-February-2020]

About Authors...

Petr LORENC is a Ph.D. student of conversational artificial intelligence at Faculty of Electrical Engineering, CTU.

He works on conversational AI Alquist, which was the second prize winner of Amazon Alexa Prize. He finished his master degree in 2019 with the master's topic *Semantic understanding of natural conversation* at FIT CTU.