KB-NLG: From Knowledge Base to Natural Language Generation

Wen Cui, Minghui Zhou, Rongwen Zhao and Narges Norouzi
University of California, Santa Cruz
{wcui7, mzhou35, rzhao17, nanorouz}@ucsc.edu

Abstract

We perform the natural language generation (NLG) task by mapping sets of Resource Description Framework (RDF) triples into text. First we investigate the impact of increasing the number of entity types in delexicalisaiton on the generation quality. Second we conduct different experiments to evaluate two widely applied language generation systems, encoder-decoder with attention and the Transformer model on a large benchmark dataset. We evaluate different models on automatic metrics, as well as the training time. To our knowledge, we are the first to apply Transformer model to this task.

1 Introduction

Knowledge bases are playing an increasingly important role in enhancing the intelligence of conversational agents, web search engines and information integration. In recent years, several large knowledge bases have been built representing semantic Web of Data. Recent work has focused on generating natural language from knowledge bases (Gardent et al., 2017b). Delexicalisation technique has been often applied to such task and thus in this work we explore the impact of varying level of abstraction in delexicalisation process. Furthermore we conduct experiments and compare performance using encoder-decoder with attention and Transformer models.

2 Dataset and Preprocess

We utilized all WebNLG challenge 2017 dataset (Gardent et al., 2017a) and maintained the original split of training, development and test sets. We applied delexicalisation by replacing entity mentions with an index and its type. Table 1 shows an example of model input and output after delexicalisation. By querying DBPedia ontology[1] we could get entities and a list of types. Then we set different level thresholds in the ontology to assign types. A 'THING' type was assigned if the entity cannot be resolved. Table 2 shows the most common 10 entity types with various ontology levels (namely 1, 3, 5) and their impact on the distribution of entity types as well as the total number of types.

Input	Aarhus_Airport location Tirstrup
Del. input	ENTITY_1 AIRPORT location ENTITY_2 PLACE
Del. target	ENTITY_1 is located in ENTITY_2 .
Target	Aarhus Airport is located in Tirstrup.

Table 1: An example of model delexicalised input and output

[1] http://dbpedia.org/ontology/

Level 1 (35)	**Level 3 (41)**	**level 5 (41)**
THING 1192	THING 1192	THING 1192
AGENT 772	PERSON 291	PERSON 252
PLACE 526	SETTLEMENT 288	SOCCERCLUB 159
TIMEPERIOD 70	SPORTSTEAM 168	SETTLEMENT 154
FOOD 66	ORGANISATION 120	ORGANISATION 132
WORK 55	OFFICEHOLDER 111	CITY 131
ETHNICGROUP 50	COMPANY 84	OFFICEHOLDER 109
SPECIES 49	REGION 73	COMPANY 83
MEANOFTRANSPORTATION 45	YEAR 71	ADMINISTRATIVEREGION 81
PERSON 39	FOOD 61	YEAR 71

Table 2: Entity types distribution in terms of various ontology level 1, 3 and 5. The total number of entity types is in the parenthesis in bold.

3 Models

Following the success of encoder-decoder framework (Cho et al., 2014; Sutskever et al., 2014; Luong et al., 2015) in machine translation, we adopted the same framework in our task. We also explored incorporating multi-head attention (Vaswani et al., 2017) which allows the model to attend multiple locations of the input triples and ignore the ordering.

4 Experiments

We first experimented with encoder-decoder framework by tuning the hyper-parameters and the best model achieved at assigning leaf-most entity types with word vector size of 500, 2 layers of encoder and decoder with LSTM size of 100, and dropout of 0.3. Then we used the same model parameters to test with different number of entity types described in Section 2. Finally we compared with Transformer model by setting the hyper-parameters originally described in the paper without fine-tuning. Table 3 shows the performance of different models in terms of two automatic evaluation matrices BLEU4 (Papineni et al., 2002) and METEOR (Denkowski and Lavie, 2014).

Models	**BLEU4**	**METEOR**	**Training time**
Encoder-decoder (Level 1)	30.87	32.14	1.5 hour
Encoder-decoder (Level 2)	31.75	33.85	1.5 hour
Encoder-decoder (Level 3)	31.62	34.73	1.5 hour
Encoder-decoder (Level 4)	30.73	33.03	1.5 hour
Encoder-decoder (Leaf-most)	**33.09**	**34.01**	1.5 hour
Transformer	31.39	36.05	**30 min**

Table 3: Models comparison with automatic metrics

From Table 3 we can see there is no obvious trend showing the number of entity types would have a significant effects on the output but the model with leaf-most entity types does yield the best performance. And with less training time the Transformer model could also achieve relatively good performance.

5 Conclusion

In this work, we first performed delexicalisation with various number of entity types, and results were not significantly different. One reason could be due to the size of DBPedia ontology, there is no significant difference in the distribution of entity types showed in Table 2 that could have an effect on the model performance. Second we experimented with Transformer model and compared it with encoder-decoder model. The Transformer model achieved relatively good results during one third of the encoder-decoder training time.

References

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*, pages 376–380.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017a. Creating training corpora for micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada, August. Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017b. The webnlg challenge: Generating text from rdf data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.