

UNIVERSITY OF TILBURG

---

Tilburg University models for the  
WebNLG challenge

---

*Author:*

Chris van der Lee and  
Thiago Castro Ferreira

*Supervisors:*

Dr. Emiel Krahmer and  
Dr. Sander Wubben

August 21, 2017

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Delexicalization</b>	<b>2</b>
<b>3</b>	<b>Pipeline Model</b>	<b>3</b>
<b>4</b>	<b>Statistical Machine Translation model</b>	<b>4</b>
<b>5</b>	<b>Neural Machine Translation model</b>	<b>4</b>
<b>6</b>	<b>Results</b>	<b>5</b>

# 1 Introduction

This report aims to describe the models developed by the group from Tilburg University to the WebNLG challenge. In order to compare the performance of different approaches for automatic text generation, we introduce 3 models to convert RDF triples into text: a pipeline, a statistical machine translation and a neural machine translation one. In Section 2, we explained the delexicalization procedure used to obtain the templates to be predicted; whereas Sections 3, 4 and 5 describes the 3 models proposed to the task, respectively.

## 2 Delexicalization

The delexicalization process is responsible for replacing all the references to entities on the text for tags. Given a set of RDF triples and a text that describes the information in this set, delexicalization is given in the following way: all entities that appear on the left side of the triples are called AGENTs, the ones at the right side are called PATIENTs and the ones that appear on both sides are called BRIDGEs. For example, consider the following set of triples:

```
Appleton_International_Airport | location | Greenville,_Wisconsin
Greenville,_Wisconsin | isPartOf | Ellington,_Wisconsin
Greenville,_Wisconsin | isPartOf | Menasha_(town),_Wisconsin
Greenville,_Wisconsin | country | United_States
Appleton_International_Airport | cityServed | Appleton,_Wisconsin
```

The set contains one AGENT entity (Appleton\_International\_Airport), one BRIDGE entity (Greenville,\_Wisconsin) and four PATIENT entities (Ellington,\_Wisconsin, Menasha\_(town),\_Wisconsin, United\_States and Appleton,\_Wisconsin). Now given the following text that describes the set of triples:

The Appleton International Airport is located in Greenville, Wisconsin, United States and serves the city of Appleton, Wisconsin. Greenville is part of the town of Menasha and Ellington, Wisconsin.

Its delexicalized version can be:

**AGENT-1** is located in **BRIDGE-1** , **PATIENT-1** and serves the city of **PATIENT-2** . **BRIDGE-1** is part of **PATIENT-3** and **PATIENT-4** .

To distinguish different AGENTs, PATIENTs and BRIDGEs in a set, an ordinary ID is given to each entity of each kind as well (PATIENT-1, PATIENT-2, etc.). The delexicalized texts are the templates that will be produced by our pipeline and neural models given a set of RDF triples as we will explain further.

### 3 Pipeline Model

Our pipeline system converts a set of triples into text in 4 steps: discourse ordering, template selection, referring expression generation and text reranking.

In discourse ordering, we aim to find the most likely order(s) of the triples by making use of machine learning techniques. Later, the model extracts the most likely template(s) that describe the ordered set. The system first looks for templates of the same semantic category of the triple set, and only consider all the templates on the training set if no template is found in the first attempt. In case no template is found for the triple set, the model breaks it in subsets so that a template can be found for each subset and combined into one, always following the discourse order.

Referring expression generation is performed to replace the tags for real references to the entities. In other words, it is the step responsible for lexicalize the templates. The choice of referential form is performed according to Castro Ferreira et al. (2016). Later, according to the form choice chosen, proper names, pronouns, descriptions and demonstrative references are realized, making sure to keep the coherence of the text. For instance, pronouns are only realized when there are no distractors to the target.

Finally, the 100 most likely texts that describes the set of triples given as input are reranked using a 6-gram language model trained on Gigaword corpus Third Edition with KenLM (Heafield et al., 2013). The most likely text according to the language model is returned as the one that best describes the triple set.

## 4 Statistical Machine Translation model

We used a standard Phrase-based Machine Translation system built using Moses toolkit (Koehn et al., 2007). As training set, we used the parallel triples-text provided for training augmented by the Wikipedia IDs of the entities in the source side and their referring expressions extracted on the delexicalization process in the target side. Templates are not used in this approach.

At training time, we extract and score phrase sentences up to the size of 20 tokens. All the feature functions - direct and inverse phrase translation probabilities and lexical weighting; word, unknown word and phrase penalties - were trained using alignments from the training set obtained by MGIZA (Gao and Vogel, 2008). Model's weights were tuned on the development data using k-batch MIRA with  $k = 60$  (Cherry and Foster, 2012) with BLEU as the evaluation metric. A distortion limit of 6 was used for the reordering models. We used two lexicalized reordering models: phrase-msd-bidirectional-fe and hier-mslr-bidirectional-fe. At decoding time, we use a stack size of 1000.

The language model is also a 6-gram LM trained on the Gigaword Third Edition corpus using KenLM.

## 5 Neural Machine Translation model

Our neural model is based on Edinburgh Neural MT submission (UEDIN-NMT) for the shared translation task at the 2016 Workshops on Statistical Machine Translation<sup>1</sup>. This model aims to predict a template from a linearized and delexicalized set of triples with a maximum sentence length of 50.

Source and target word embeddings are 620D each, whereas hidden units are 1024D. Gradient are normalized to 1.0. Models are trained using stochastic gradient descent with Adadelta (Zeiler, 2012) and minibatches of size 80. We apply early stopping for model selection based on BLEU scores. We apply dropout with a probability of 0.1 in both source and target word embeddings and 0.2 for hidden units. Decoding is performed with beam search with a size of 12.

Once the template is predicted, we use the same Referring Expression Generation module from the pipeline system to lexicalize the template.

---

<sup>1</sup><https://github.com/rsennrich/wmt16-scripts>

Model	BLEU
Pipeline	41.37
SMT	54.53
NMT	43.36

Table 1: Results in the development set

## 6 Results

Table 1 depicts the BLEU score of the different models.

## References

- Thiago Castro Ferreira, Emiel Krahmer, and Sander Wubben. Towards more variation in text generation: Developing and evaluating variation models for choice of referential form. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 568–577, Berlin, Germany, August 2016. Association for Computational Linguistics. URL <http://www.aclweb.org/anthology/P16-1054>.
- Colin Cherry and George Foster. Batch tuning strategies for statistical machine translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL HLT '12*, pages 427–436, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics. ISBN 978-1-937284-20-6. URL <http://dl.acm.org/citation.cfm?id=2382029.2382089>.
- Qin Gao and Stephan Vogel. Parallel implementations of word alignment tool. In *Software Engineering, Testing, and Quality Assurance for Natural Language Processing, SETQA-NLP '08*, pages 49–57, Stroudsburg, PA, USA, 2008. Association for Computational Linguistics. ISBN 978-1-932432-10-7. URL <http://dl.acm.org/citation.cfm?id=1622110.1622119>.
- Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. Scalable modified kneser-ney language model estimation. In *Proceedings of the*

*51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 690–696, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL <http://www.aclweb.org/anthology/P13-2121>.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL <http://dl.acm.org/citation.cfm?id=1557769.1557821>.

Matthew D. Zeiler. ADADELTA: An Adaptive Learning Rate Method. *CoRR*, abs/1212.5701, 2012. URL <http://arxiv.org/abs/1212.5701>.