# WebNLG Phase 2 Class Project: CSE 597, Spring 2019

By
Saptarashmi Bandhyopadyay,
Mitchel Myers,
Yuchen Sun
and
Namita Kalady-Prathap

April 23, 2019

# Outline of the presentation

- Objective
- Data preparation
- Phase 1 - Delexicalisation
- Phase 2
  - Improved Delexicalisation
  - Relexicalisation
  - Multiple Reference Files for BLEU score calculation
- Train/Dev/Test Split
- Training Parameters and the Attention model in Phase 2
- Results
- Comparison with WebNLG systems
- Observations
- Conclusion
- Future Work
- References

# Objective of the Project

- To develop one or more neural models to generate natural language sentences from the WebNLG data.

- To improve the translation quality of RDF triples to natural language.

# Data Preparation

- The provided training and development files are split into the training, development and testing datasets.
- These files are henceforth referred as the 20k dataset due to the total number of lines in the training, development and testing dataset being 20283.
- Due to paucity of data, the unseen test data is considered along with the provided training and development files, which are then split into the training, development and testing datasets.
- These files are henceforth referred as the 22k dataset due to the total number of lines in the training, development and testing dataset being 22716.
- The 20k and 22k datasets have different delexicalisation dictionaries in our project as all the entities in the subject of the triples in the 2 datasets are mapped to their DBpedia categories.
- The testing of the unseen dataset is also done separately by using the training, and development files of the 20k dataset and by delexing the unseen test file with the delex dictionary of the 20k dataset.

# Dataset Statistics

| **Dataset** | **Size of Training file (in number of lines)** | **Size of Development file (in number of lines)** | **Size of Testing file (in number of lines)** | **Total number of lines in the training, development and testing files** |
|---|---|---|---|---|
| 20k Dataset | 16226 | 2029 | 2028 | 20283 |
| 22k Dataset | 18172 | 2269 | 2275 | 22716 |

# Phase 1 - Delexicalisation

- For the RDF triple <s,p,o> where s refers to subject, p refers to the predicate and o refers to the object.

- o is replaced by p in upper case.

- s is replaced by the DBPedia category.

- It is useful to capture the semantic representation of RDF and lex.

- Delex_dict.json provided by WebNLG to recreate their challenge.

# Phase 1 - Improvements over WebNLG Baseline

- Entities are replaced by direct matching before tokenization.
    - “A . S . Roma league LEAGUE A . S . Roma ground GROUND”  to  “SPORTSTEAM league LEAGUE SPORTSTEAM ground GROUND

- The bug in WebNLG baseline of subjects being replaced by objects has been corrected.
    - WebNLG Triple : WRITTENWORK country COUNTRY COUNTRY ethnicGroup Native Americans in the COUNTRY
    - Phase 1 Triple: WRITTENWORK country COUNTRY United_States ethnicGroup ETHNICGROUP

# Phase 2 - Improved Delexicalisation

- Improved Delexicalisation from Baseline.

- Used SPARQL to query and extract the DBpedia categories for all the named entities in the train, dev and test data for the 20k dataset and 22k dataset from the DBpedia knowledge base.

- Our delexicalisation process ensured that all the entities present in the subject position were delexicalized correctly, allowing for better generalization.

# Phase 2 - Example of the Improved Delexicalisation

- ‘Aarhus_Airport’ entity in ‘1triple_allSolutions_Airport_train_challenge.xml’ file.
- 23 RDF types obtained; one of which is Airport.
- The category in the delexicalisation dictionary is automatically assigned to Airport.
- ‘Spain’ entity in ‘1triple_allSolutions_Food_train_challenge.xml’ file.
- 29 RDF types obtained, none of which are Food.
- The rdf types are observed manually and ‘Spain’ entity is mapped to the ‘Country’ type manually in the delexicalisation dictionary.

# Statistics of the Updated Delexicalisation Dictionary

| | **20K Dataset** | **22K Dataset** |
|---|---|---|
| Total DBpedia Categories | 43 | 58 |
| Total number of Entities | 434 | 719 |
| Entities Assigned Automatically | 208 | 323 |
| Entities Assigned Manually | 226 | 396 |

# Modification of the Vocabulary for Training

- Constructed in two ways :

- First method - Each word in the triples and the lexicalisations are tokenized and stored in each line of the corresponding vocabulary files of triples and lexes.
- The size of the vocabulary file reduces after delexicalisation which makes the dataset more general and helps in faster training.

# Byte Pair Encoding (BPE) of the Vocabulary

- Second method - Byte Pair Encoding (BPE).
- Word surface forms are divided into its root word and affix.
- Inflected words are often not consistently segmented due to unsupervised word segmentation.
- The objective is to handle the out-of-vocabulary words.
- BPE vocabulary improves the vocabulary and hence the BLEU score.
- However, the improvement is not much, as BPE expects words to be aligned on the source and target side of the Neural Machine Translation.
- In the NLG task, the structure of the predicates are getting translated in the triples to lexicalisations.

# Vocabulary Statistics

| Vocabulary Size | 20K dataset | 22K dataset |
|---|---|---|
| Source (triples) before delex | 4931 | 5964 |
| Source (triples) after delex | 527 | 750 |
| Target (lex) before delex | 5908 | 7603 |
| Target (lex) after delex | 2449 | 3629 |
| Using BPE (triple + lex) | 4677 | 5722 |

# Improved Pre-Processing of the Delexicalised Triples

- In Phase one, WebNLG baseline model generated predicates and objects separated by blank spaces ‘ ‘ in between multiple tokens after delexicalisation.

- e.g. A triple was “ASTRONAUT was selected by NASA WAS SELECTED BY NASA”

- Difficult to distinguish the subject, predicate and the object in the triples as the subject, predicate and object are also joined by a blank space.

- The pre-processing has been updated in such a way, that multiple tokens in the predicate and object after delexicalisation are joined by ‘_’.

# Multiple Reference Files for BLEU score calculation

- We noticed that a chain of triples can have multiple lexicalisations in the data.

- This leads to repetitions of the triple chains in the triple files for training, development and testing.

- Therefore, multiple reference files are created from the lex files.
  - Test-referene0.lex file has the first lexicalisation of all unique triples

- The BLEU scores of the generated lexicalisation improves significantly on using the multiple reference files.

# Relexicalisation

- Implemented relexicalisation in the post-processing step of Phase 2.
- Definition - process of converting the categories in the delexicalized file to corresponding tokenized entities for the category in the source triple file.
- Done on the category, as category represents multiple entities.
- Relex dictionary file is created for each chain of triples while the data is being delexicalised.
- The system generated delexicalised file is provided as input to the relexicaliser which  refers to the particular dictionary in the file for each lexicalisation.
- The extracted entity is then tokenized by replacing the underscores with blank spaces.

# An example of delexicalisation and relexicalisation

- **Initial triple**: Taylor_County,_Texas largestCity Abilene,_Texas
- **Initial lex**: Abilene is the largest city in Taylor County , Texas
- **Delexicalised triple**: LOCATION largestCity LARGESTCITY
- **Delexicalised lex**: Abilene is the largest city in LOCATION
- **Dictionary entry in the Relex dictionary file**: [["Taylor_County,_Texas", "LOCATION"], ["Abilene,_Texas", "LARGESTCITY"]]
- **Output lex after testing**: The nearest city to LOCATION is NEARESTCITY.
- **Relexicalised lex**: The nearest city to Taylor County, Texas is NEARESTCITY.

# Train/Dev/Test Split (Shimorina and Gardent)

- WebNLG has split the data into train/dev/test by an 80%/10%/10% split.
- Triples of different size and different categories are distributed equally.
- Test has equal numbers of seen and unseen categories.
- It has also been ensured that there is no overlap among the training, development and test datasets before delexicalisation.

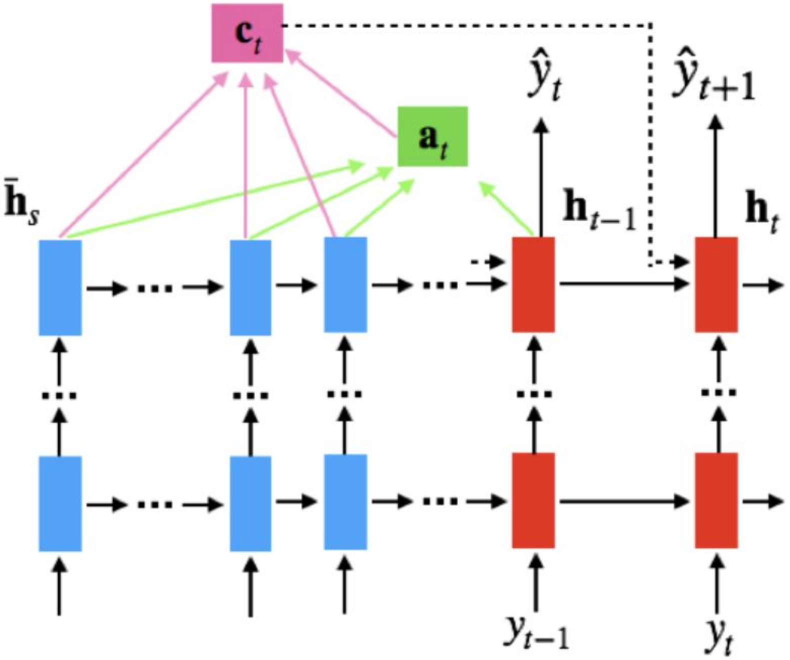
# Training parameters

- 2/4 layered uni-directional recurrent neural network
- Drop-out rates = 0.3 (as per WebNLG) / 0.5
- Attention models = standard (as per WebNLG) / scaled_luong / normed_bahdanau models
- Batch size = 128
- 500 hidden layers (as per WebNLG)
- Learning rate: The initial learning rate is 1
- Num_train_steps = 12000
- Num_units (network size) = 500

# Attention Model: Bahdanau

## Bahdanau Attention Mechanism

$$\mathbf{h}_{t-1} \rightarrow \mathbf{a}_t \rightarrow \mathbf{c}_t \rightarrow \mathbf{h}_t$$



$$\mathbf{a}_t(s) = \text{align}(\mathbf{h}_{t-1}, \bar{\mathbf{h}}_s)$$
$$\mathbf{c}_t = \sum a_t \mathbf{h}_s$$
$$\mathbf{h}_t = \text{RNN}(\mathbf{h}_{t-1}^{l-1}, [\mathbf{c}_t; \mathbf{h}_{t-1}])$$

Compared to Bahdanau, Luong:

- Use hidden states at the top LSTM layers in both the encoder and decoder.
- Concat context vector and the current hidden state by RNN-like structure.
- Has more alignment scoring functions.

# Attention Model: Luong

$$c_t = \sum_{k=1}^{T} \alpha_{k,t} h_k$$

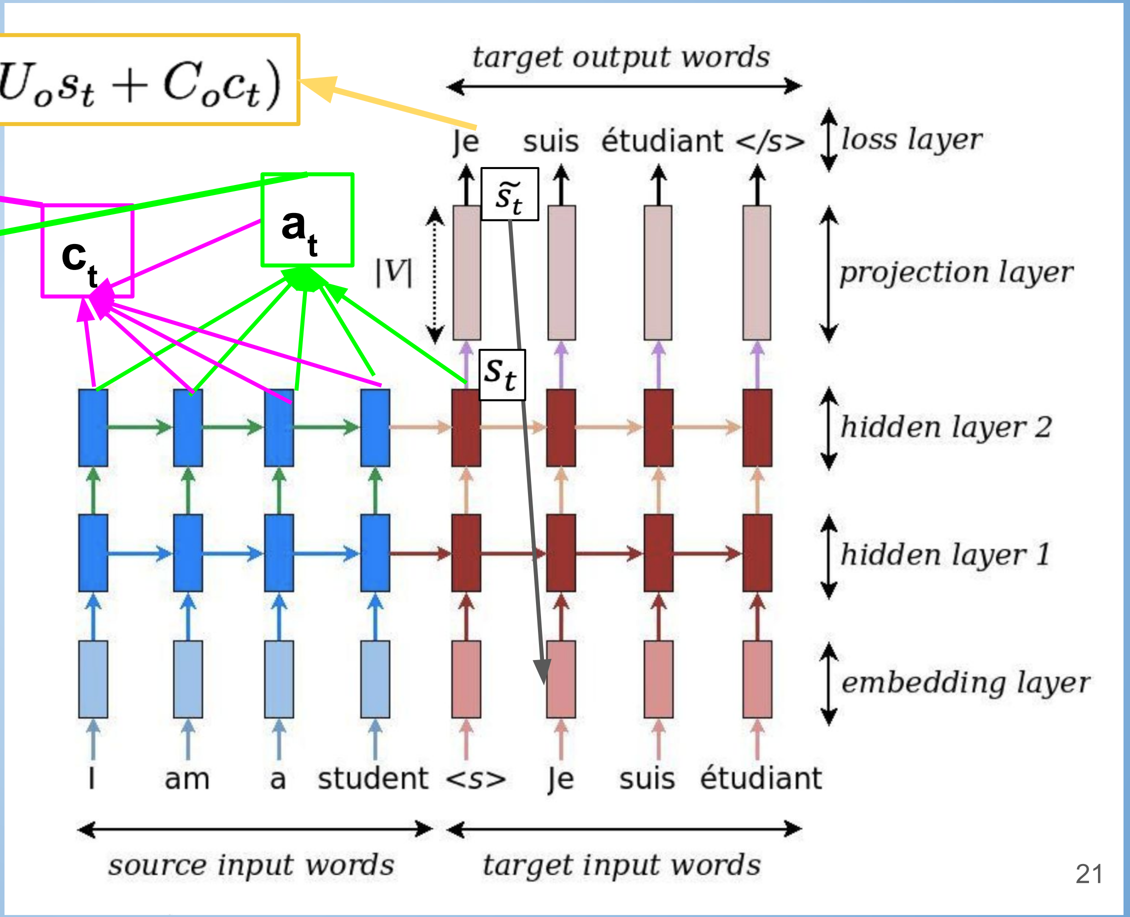
$$y_t = \sigma(W_o y_{t-1} + U_o s_t + C_o c_t)$$

$$e_{j,t} = V_a \cdot \tanh(W_a s_{t-1} + U_a h_j)$$

$$\alpha_{j,t} = \frac{\exp(e_j)}{\sum_{k=1}^{T} \exp(e_k)}$$

$$\begin{aligned} r_t &= \sigma(W_r y_{t-1} + U_r s_{t-1} + C_r c_t) \\ z_t &= \sigma(W_z y_{t-1} + U_z s_{t-1} + C_z c_t) \\ \hat{s}_t &= \tanh(W_p y_{t-1} + U_p[r_t \circ s_{t-1}] + C_p c_t) \\ s_t &= (1 - z_t) \circ s_{t-1} + z_t \circ \hat{s}_t \end{aligned}$$

$$\mathrm{score}(\boldsymbol{h}_t, \bar{\boldsymbol{h}}_s) = \begin{cases} \boldsymbol{h}_t^\top \bar{\boldsymbol{h}}_s \\ \boldsymbol{h}_t^\top \boldsymbol{W_a} \bar{\boldsymbol{h}}_s \\ \boldsymbol{v}_a^\top \tanh\left(\boldsymbol{W_a}[\boldsymbol{h}_t; \bar{\boldsymbol{h}}_s]\right) \end{cases}$$



# Weight Normalization

- Typical neural networks are composed of conceptually simple computational building blocks sometimes called neurons: each such neuron computes a weighted sum over its inputs and adds a bias term.

$$y = \phi(\mathbf{w} \cdot \mathbf{x} + b),$$

- Reparameterize each weight vector w in terms of a parameter vector v and a scalar parameter g :

$$\mathbf{w} = \frac{g}{||\mathbf{v}||}\mathbf{v}$$

$$\nabla_g L = \frac{\nabla_{\mathbf{w}} L \cdot \mathbf{v}}{||\mathbf{v}||}, \qquad \nabla_{\mathbf{v}} L = \frac{g}{||\mathbf{v}||}\nabla_{\mathbf{w}} L - \frac{g \nabla_g L}{||\mathbf{v}||^2}\mathbf{v}$$

- Smoothing the gradient to make learning robust.
- Speeds up convergence of stochastic gradient descent.

# Results - Influence of Delex / Relex / BPE

| Dropout | Encoder Type | Attention model | Number of Layers | BLEU Scores [delex] | BLEU Scores [relex] | Data | Vocab file |
|---|---|---|---|---|---|---|---|
| 0.3 | Unidirectional | Standard attention model | 2 | dev bleu = 19.7<br>test bleu = 18.1 | dev bleu = 38.83<br>test bleu = 34.68 | 20k | normal |
| 0.5 | Unidirectional | Standard attention model | 2 | dev bleu = 19.3,<br>test bleu = 19.1 | dev bleu = 39.87<br>test bleu = 39.03 | 20k | normal |
| 0.3 | Unidirectional | Standard attention model | 2 | dev bleu = 19<br>test bleu = 18.5 | dev bleu = 39.10<br>test bleu = 36.48 | 20k | BPE |
| 0.5 | Unidirectional | Standard attention model | 2 | dev bleu = 18.6<br>test bleu = 18.4 | dev bleu = 40.20<br>test bleu = 40.93 | 20k | BPE |
| 0.3 | Unidirectional | Standard attention model | 2 | dev bleu = 19<br>test bleu = 18.5 | dev bleu = 37.10<br>test bleu = 36.48 | 22k | BPE |

# Results - Influence of number of layers

| Dropout | Encoder Type | Attention model | Number of Layers | BLEU Scores [delex] | BLEU Scores [relex] | Data | Vocab file |
|---|---|---|---|---|---|---|---|
| 0.3 | Unidirectional | Standard attention model | 2 | dev bleu = 19.7<br>test bleu = 18.1 | dev bleu = 38.83<br>test bleu = 34.68 | 20k | normal |
| 0.3 | Unidirectional | Standard attention model | 4 | dev bleu = 19.3,<br>test bleu = 18.2 | dev bleu = 38.04<br>test bleu = 36.74 | 20k | normal |
| 0.5 | Unidirectional | Standard attention model | 2 | dev bleu = 19.3,<br>test bleu = 19.1 | dev bleu = 39.87<br>test bleu = 39.03 | 20k | normal |
| 0.5 | Unidirectional | Standard attention model | 4 | dev bleu = 19.4<br>test bleu = 18.4 | dev bleu = 38.78<br>test bleu = 37.31 | 20k | normal |
| 0.3 | Unidirectional | Scaled Luong | 2 | dev bleu = 23<br>test bleu = 21.5 | dev bleu = 45.05<br>test bleu = 40.07 | 20k | normal |
| 0.3 | Unidirectional | Scaled Luong | 4 | dev bleu = 22.6<br>test bleu = 19.4 | dev bleu = 40.26<br>test bleu = 38.61 | 20k | normal |

# Result - unseen data : WebNLG formulation

| Dropout | Encoder Type | Attention model | Number of Layers | BLEU Scores | Data | Vocab file |
|---|---|---|---|---|---|---|
| 0.5 | Unidirectional | Normed Bahdanau | 2 | test bleu after relex = 8.34 test bleu of delex data = 4.5 | 20k for train and dev, unseen test for testing | normal |

# Results - Influence of attention mechanism

| Dropout | Encoder Type | Attention model | Number of Layers | BLEU Scores [delex] | BLEU Scores [relex] | Data | Vocab file |
|---|---|---|---|---|---|---|---|
| 0.3 | Unidirectional | Scaled Luong | 2 | dev bleu = 23<br>test bleu = 21.5 | dev bleu = 45.05<br>test bleu = 40.07 | 20k | normal |
| 0.5 | Unidirectional | Scaled Luong | 2 | dev bleu = 24.1<br>test bleu = 21.7 | dev bleu = 44.43<br>test bleu = 40.47 | 20k | normal |
| 0.3 | Unidirectional | Scaled Luong | 2 | dev bleu = 24.5<br>test bleu = 22 | dev bleu = 46.33<br>test bleu = 41.33 | 20k | BPE |
| 0.5 | Unidirectional | Scaled Luong | 2 | dev bleu = 25.5<br>test bleu = 23.7 | dev bleu = 45.28<br>test bleu = 41.51 | 20k | BPE |

# Results - Influence of attention mechanism

| Dropout | Encoder Type | Attention model | Number of Layers | BLEU Scores [delex] | BLEU Scores [relex] | Data | Vocab file |
|---|---|---|---|---|---|---|---|
| 0.3 | Unidirectional | Normed Bahdanau | 2 | dev bleu = 22.6<br>test bleu = 20.8 | dev bleu =40.2<br>test bleu =  39.8 | 20k | normal |
| 0.5 | Unidirectional | Normed Bahdanau | 2 | dev bleu = 23.2<br>test bleu = 21.8 | dev bleu =42.1<br>test bleu =39.7 | 20k | normal |
| 0.3 | Unidirectional | Normed Bahdanau | 2 | dev bleu = 20.9<br>test bleu = 19.7 | dev bleu = 41.31<br>test bleu = 40.59 | 20k | BPE |
| 0.5 | Unidirectional | Normed Bahdanau | 2 | dev bleu = 22<br>test bleu = 21.2 | dev bleu = 43.83<br>test bleu = 42.67 | 20k | BPE |

# Results - Influence of Dropout Rate

| Dropo ut | Encoder Type | Attention model | Number of Layers | BLEU Scores [delex] | BLEU Scores [relex] | Data | Vocab file |
|---|---|---|---|---|---|---|---|
| 0.3 | Unidirectional | Standard attention model | 2 | dev bleu = 19.7<br>test bleu = 18.1 | dev bleu = 38.83<br>test bleu = 34.68 | 20k | normal |
| 0.5 | Unidirectional | Standard attention model | 2 | dev bleu = 19.3<br>test bleu = 19.1 | dev bleu = 39.87<br>test bleu = 39.03 | 20k | normal |
| 0.3 | Unidirectional | Standard attention model | 2 | dev bleu = 19<br>test bleu = 18.5 | dev bleu = 39.10<br>test bleu = 36.48 | 20k | BPE |
| 0.5 | Unidirectional | Standard attention model | 2 | dev bleu = 18.6<br>test bleu = 18.4 | dev bleu = 40.20<br>test bleu = 40.93 | 20k | BPE |
| 0.3 | Unidirectional | Scaled Luong | 2 | dev bleu = 23 test bleu = 21.5 | dev bleu = 45.05<br>test bleu = 40.07 | 20k | normal |
| 0.5 | Unidirectional | Scaled Luong | 2 | dev bleu = 24.1<br>test bleu = 21.7 | dev bleu = 44.43<br>test bleu = 40.47 | 20k | normal |

# Comparison of BLEU scores with the WebNLG 2017 challenge systems

| Serial No. | System | BLEU Score (Seen Categories) |
|---|---|---|
| 1 | ADAPT | 60.59 |
| 2 | Melbourne | 54.52 |
| 3 | Tilb-SMT | 54.29 |
| 4 | Baseline | 52.39 |
| 5 | PKUWriter | 51.23 |
| 6 | Tilb-NMT | 43.28 |
| **7** | **Our Model** | **42.67** |
| 8 | UPF-FORGe | 40.88 |
| 9 | UIT-VNU | 19.87 |

# Comparison of BLEU scores with the WebNLG 2017 challenge systems

| **Serial No.** | **System** | **BLEU Score (All Categories)** |
|---|---|---|
| 1 | Melbourne | 45.13 |
| 2 | Tilb-SMT | 44.28 |
| **3** | **Our Model** | **41.5** |
| 4 | PKUWriter | 39.88 |
| 5 | UPF-FORGe | 38.65 |
| 6 | Tilb-NMT | 34.6 |
| 7 | Baseline | 33.24 |
| 8 | ADAPT | 31.06 |
| 9 | UIT-VNU | 7.07 |

# Observations

- The BLEU scores have improved significantly.
- Higher dropout rate gives either a similar or higher accuracy compared to that of a lower drop out rate. (helps avoid overfitting)
- The accuracy increases drastically when we use the attention mechanisms of scaled luong and normed bahdanau.
- BLEU scores for the BPE vocabulary files are higher than the scores obtained from testing the data with the normal vocabulary files.

# Conclusion

- We identified weaknesses of the baseline system and modified our logic for delexicalisation and implemented relexicalisation as well.
- Delexicalisation and relexicalisation provides the greatest increase to the performance of the model, producing a significant increase in the BLEU scores.
- Use of an attention model can lead to a further increase in performance.
- Performance increased by small amounts by BPE for vocabulary generation and use of a dropout rate of 0.5 rather than 0.3.

# Future Work

- Automate the selection of entity category information for the delexicalisation dictionary.

- Train a model to automatically organize entities and predicates into a ordering that aligns with natural language.

# Future work

- Improve delexicalisation procedure from the replacement of entities by categories by replacing direct matching of the tokenized entities with some similarity metrics.
- Rules can be devised before delexicalisation to capture the syntactic information of the predicate structure in the lexicalisation.
- Rules can be devised during post-processing the data, to incorporate missing categories in the translation with respect to the source triples.
- Further experiments can be done by varying the training parameters to improve the quality of the lexicalisation.

# References

1. The WebNLG Challenge: Generating Text from RDF Data, Gardent et. al., Proceedings of the 10th International Natural Language Generation Conference, pp. 124-133, Spain, 2017.

2. Handling Rare Items in Data-to-Text Generation, Shimorina and Gardent, Proceedings of the 11th International Natural Language Generation Conference, pp. 360-370, Netherlands, 2018.

3. Bahdanau, D., Cho, K. and Bengio, Y. (2015). Neural Machine Translation by Jointly Learning to Align and Translate.

4. Salimans, T., and Kingma, D.P. (2016). Weight Normalization: A Simple Reparameterization to Accelerate Training of Deep Neural Networks.

5. https://github.com/rsennrich/subword-nmt

6. https://www.nltk.org/

# Thank you!