NILC-SWORNEMO at the Surface Realization Shared Task: Exploring Syntax-Based Word Ordering using Neural Models

Marco A. S. Cabezudo and **Thiago A. S. Pardo**
Interinstitutional Center for Computational Linguistics (NILC)
Institute of Mathematical and Computer Sciences, University of São Paulo
São Carlos - SP, Brazil
msobrevillac@usp.br, taspardo@icmc.usp.br

Abstract

This paper describes the submission by the NILC Computational Linguistics research group of the University of São Paulo/Brazil to the Track 1 of the Surface Realization Shared Task (SRST Track 1). We present a neural-based method that works at the syntactic level to order the words (which we refer by NILC-SWORNEMO, standing for "Syntax-based Word ORdering using NEural MOdels"). Additionally, we apply a bottom-up approach to build the sentence and, using language-specific lexicons, we produce the proper word form of each lemma in the sentence. The results obtained by our method outperformed the average of the results for English, Portuguese and Spanish in the track.

1 Introduction

In recent years, Universal Dependencies[1] (UD) have gained interest from many researchers across different areas of Natural Language Processing (NLP). Currently, there are treebanks for about 50 languages that are freely available[2].

UD treebanks have already proved useful in the development of multilingual applications, becoming an advantage for developers. Thus, the creation of an application for a specific language may be replicable to other languages.

The Surface Realization Shared Task (Mille et al., 2018) aims at continuing with the development of natural language generation methods focused on the surface realization task. In this edition of the task, two tracks were proposed: (1) Shallow Track, which aimed at ordering the words in a sentence and recovering their correct forms, and (2) Deep Track, which aimed at ordering the words and introducing missing functional words and morphological features.

For building the dataset for the Shallow Track, the UD structures were processed as follows:

- the information on word ordering is removed by randomly scrambling the words;
- the words are replaced by their lemmas.

An example of the input data to this track is shown in Figure 1. In this example, we may see information about lemmas, grammatical categories, inflection information and dependency relations.

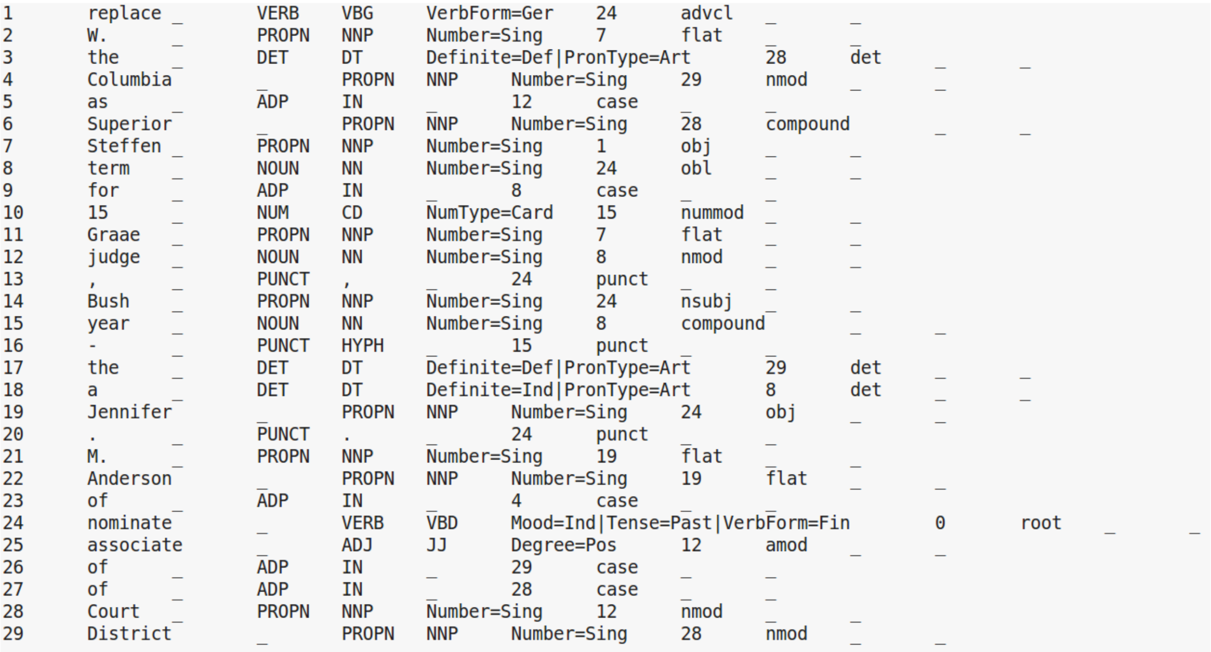
Track 1 can be seen as word ordering and inflection generation tasks. Word ordering is a fundamental problem in Natural Language Generation (Reiter and Dale, 2000). This problem have been widely studied, e.g., we may see the works proposed for the Shared Task in Surface Realization (Belz et al., 2011). In relation to this problem, this has been addressed using language modeling (Schmaltz et al., 2016) and syntax-based approaches (Zhang and Clark, 2015). Recently, sequence-to-sequence models have also been used to tackle this problem, obtaining good results (Hasler et al., 2017).

In this paper, we present a neural-based method that works at the syntactic level to order the words (which we refer by NILC-SWORNEMO, standing for "Syntax-based Word ORdering using NEural MOdels", developed by the NILC research group on Computational Linguistics). Additionally, we apply a bottom-up approach to build the sentence and, using language-specific lexicons, we produce the word forms of each lemma in the sentence. Our system is described in Section 2. In Section 3, the results of our proposal are presented. Finally,

[1] Available at http://universaldependencies.org/#en

[2] Available at https://lindat.mff.cuni.cz/repository/xmlui/handle/11234/1-1983

Proceedings of the First Workshop on Multilingual Surface Realisation, pages 58–64
Melbourne, Australia, July 19, 2018. ©2018 Association for Computational Linguistics



```
1       replace _       VERB    VBG     VerbForm=Ger    24      advcl   _       _
2       W.      _       PROPN   NNP     Number=Sing     7       flat    _       _
3       the     _       DET     DT      Definite=Def|PronType=Art       28      det     _       _
4       Columbia        _       PROPN   NNP     Number=Sing     29      nmod    _       _
5       as      _       ADP     IN      _       12      case    _       _
6       Superior        _       PROPN   NNP     Number=Sing     28      compound        _       _
7       Steffen _       PROPN   NNP     Number=Sing     1       obj     _       _
8       term    _       NOUN    NN      Number=Sing     24      obl     _       _
9       for     _       ADP     IN      _       8       case    _       _
10      15      _       NUM     CD      NumType=Card    15      nummod  _       _
11      Graae   _       PROPN   NNP     Number=Sing     7       flat    _       _
12      judge   _       NOUN    NN      Number=Sing     8       nmod    _       _
13      ,       _       PUNCT   ,       _       24      punct   _       _
14      Bush    _       PROPN   NNP     Number=Sing     24      nsubj   _       _
15      year    _       NOUN    NN      Number=Sing     8       compound        _       _
16      -       _       PUNCT   HYPH    _       15      punct   _       _
17      the     _       DET     DT      Definite=Def|PronType=Art       29      det     _       _
18      a       _       DET     DT      Definite=Ind|PronType=Art       8       det     _       _
19      Jennifer        _       PROPN   NNP     Number=Sing     24      obj     _       _
20      .       _       PUNCT   .       _       24      punct   _       _
21      M.      _       PROPN   NNP     Number=Sing     19      flat    _       _
22      Anderson        _       PROPN   NNP     Number=Sing     19      flat    _       _
23      of      _       ADP     IN      _       4       case    _       _
24      nominate        _       VERB    VBD     Mood=Ind|Tense=Past|VerbForm=Fin        0       root    _       _
25      associate       _       ADJ     JJ      Degree=Pos      12      amod    _       _
26      of      _       ADP     IN      _       29      case    _       _
27      of      _       ADP     IN      _       28      case    _       _
28      Court   _       PROPN   NNP     Number=Sing     12      nmod    _       _
29      District        _       PROPN   NNP     Number=Sing     28      nmod    _       _
```

Figure 1: Unordered sentence in CoNLL format - “Bush nominated Jennifer M. Anderson for a 15-year term as associate judge of the Superior Court of the District of Columbia, replacing Steffen W. Graae.”

some conclusions and future work are discussed in Section 4.

2 System Description

Our proposal was motivated by the works of (Hasler et al., 2017) and (Zhang and Clark, 2015). Thus, we tackled the problem by applying a syntax-based word ordering strategy using a sequence-to-sequence model (seq-2-seq). This way, we could take advantage of the importance of the syntactic information in the word ordering process (in this case, dependency relations) and the length of the sequence of words to be ordered. Thus, we could try to order sub-trees and then apply a bottom-up approach to compose the original sentence. We have to note that our approach have a limitation related to non-projective tree structures, because the allowed realizations will be generated from the dependency structure.

Additionally, we could benefit from the ability of the seq-2-seq model to deal with short sequences (delimited by the length of words in a syntactic level, i.e., a sub-tree generated by the dependency relations), and the few number of hyperparameters to tune, facilitating the training.

2.1 Data Preparation

As we mentioned, we used a neural model to order the words in the syntactic level, and this kind of model requires several instances to learn. Therefore, the first step was to generate and prepare our dataset.

The dataset used to train our models was composed by the training dataset provided by the task and a portion of the Europarl corpus (Koehn, 2005), comprising approximately 70,000 sentences for each language (English, Portuguese, and Spanish).

As our neural model works on words of a sentence according to their syntactic levels, we had to preprocess the dataset to get the words of each sentence by syntactic level. Thus, we run the UDPipe tool (Straka and Straková, 2017) on the dataset and obtained all the information about lemmas, grammatical categories, and dependency relations. Then, we got all the sub-trees (sub-root and children, only via breadth search) and generated a sequence for each sub-tree.

Each sequence was composed by tokens in the sub-tree and each token had the notation “lemma|POS-Tag|dep”, where the POS-Tag is the grammatical category and dep is the name of the dependency relation. Besides, the first token in a sequence contains the word “root” as its dependency relation. We used the POS-Tags and the dependency relations to bring more linguistic information into our models.

An example of a sub-tree may be seen in Figure 2. The returned sequence of this sub-tree was as follows: “term|NOUN|root for|ADP|case judge|NOUN|nmod year|NOUN|compound a|DET|det”.

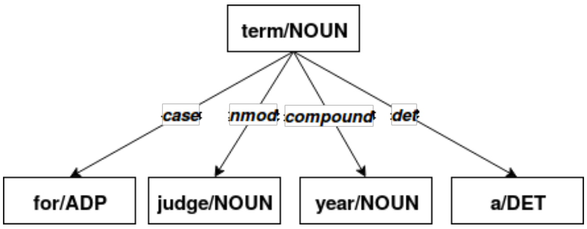


Figure 2: Sub-tree of the sentence that includes “term”, “for”, “judge”, “year”, and “a”

One problem related to the training dataset generation was the possibility of the sub-tree’s elements to appear in different ordering in the CoNLL format. This would produce different instances, as we build the samples by breadth search in a sub-tree. Thus, we could get the sample “term|NOUN|root *for|ADP|case* judge|NOUN|nmod year|NOUN|compound a|DET|det” or “term|NOUN|root judge|NOUN|nmod year|NOUN|compound for|ADP|case a|DET|det”, depending on the order in which they are presented in the CoNLL format, and producing different outputs in our model. This should not be a problem because models have to generalize independently of the order. However, we adopted a strategy to deal with this problem. The strategy was to generate a few permutations for each initial instance of the dataset and join them to build the dataset. We might generate all possible permutations for each initial instance of the dataset, but this would not be good in our case. Instead, we assumed that few permutations would be enough to generalize. Thus, we experimented generating 5, 10 and 15 instances for each instance in the dataset and tested in the neural model. Experiments showed that 5 permutations were enough to achieve a good performance and incrementing to 10 or 15 did not bring improvements.

Finally, it is important to highlight that the lemmas of proper nouns were replaced by the expression “PROPN” in order to reduce data sparsity.

2.2 Word Ordering

The neural model that we used was a sequence-to-sequence model (Encoder-Decoder) (Sutskever et al., 2014) in which the input was composed by a sequence of tokens in a sub-tree extracted by the syntactic dependency relations (described in Subsection 2.1) and the output was composed by the lemmas of the same sequence in the correct order.

In general, each token in the encoder was represented by embeddings composed by the concatenation of the word embedding, the embedding of the grammatical category and the embedding of the dependency relation. We used word embeddings of 300 dimensions provided by GloVe (Pennington et al., 2014) for English[3], Portuguese[4] (Hartmann et al., 2017), and Spanish (built over the corpus provided by Cardellino (2016)). In the case of the other features, we used the number of values that they may assume to generate the size of the embedding.

The type of cells in the Recurrent Neural Network (RNN) that we used was the Long Short-Term Memory (LSTM). We used a Bidirectional LSTM (Bi-LSTM) in the Encoder because it could give us a general understanding of the sentence (saving relations in two directions). In the case of the Decoder, we used two layers and the attention mechanism proposed by Bahdanau et al. (2014) in order to consider all words in the contexts (due to the unordered words). This proposal was similar to the recurrent neural network language model proposed in (Hasler et al., 2017).

Finally, we used a Adam Optimizer with a initial learning rate of 0.001, a dropout value of 0.3, 500 hidden units, 15 epochs, and, for the generation of the sequence, we applied beam search of size 10. Let us mention that we used OpenNMT (Klein et al., 2017) to train our model. These parameters were effective during the training, excepting the number of epochs because we did not try other settings.

2.3 Sentence Building

After the execution of the neural model, we got the words of all sub-trees (obtained by the syntactic levels) in the correct order. In order to build the sentence, we applied a bottom-up approach. Thus, we continuously started to join fragments (belong-

[3] Available at https://nlp.stanford.edu/projects/glove/

[4] Available at http://www.nilc.icmc.usp.br/nilc/index.php/repositorio-de-word-embeddings-do-nilc

ing to sub-trees) with the sub-trees in an immediately higher level until the top of the tree. The joining was performed using the token in common in both sub-trees. For example, in Figure 3, it may be seen the fragment "15 - year" in a sub-tree and the fragment "for a year term judge" in an immediate higher level, where the joining produced the fragment "for a 15 - year term judge".

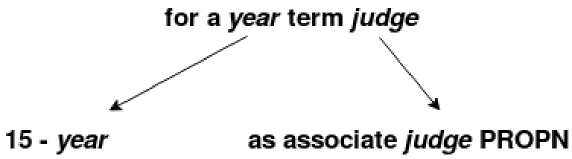


Figure 3: Portion of the ordered sub-trees

As we may see in Figure 3, one of the fragments contains the expression "PROPN". In cases where there was a "PROPN" symbol, our method simply replaced it by the correct proper noun in the original fragment. In other cases, our method had to find the correct place for each proper noun in the fragment. Additionally, there were several cases where the neural model could not obtain all the words in the fragment, mainly in situations where the number of tokens in the input was too long.

To solve these problems, we used a 3-gram language model for English (Chelba et al., 2013), Portuguese (Cunha, 2016) and Spanish (Cardellino, 2016) in order to find the correct position of the words and the proper nouns. That motivated us to follow a bottom-up approach to build a sentence. Thus, the joining between two neighbor syntactic levels makes more sense (as analyzing from the lowest levels brings correct expressions like "15 - year" or "as associate judge", instead of "for a year term judge").

2.4 Inflection Generation

In order to recover the correct words included in a sentence (and not lemmas), we created a lexicon for each language (English, Portuguese and Spanish).

To do this, we ran the UDPipe tool[5] on the Europarl corpus for English, Portuguese and Spanish (Koehn, 2005) in order to get the lemmas and the inflection information. For example, in the sentence "I ran all day", we got the following information about "ran": "run Mood=Ind|Tense=Past|VerbForm=Fin", which means that "ran" is in indicative mood, in the past tense and in its finite form, and the lemma is "run".

It is important to highlight that we only extracted the inflection information of words that belong to some specific grammatical categories, as auxiliary verbs, verbs, determiners, adjectives, pronouns, and nouns, since these categories usually contain inflection information.

The lexicons generated for English, Portuguese and Spanish contain 44,667, 143,058, and 155,482 entries, respectively. With these lexicons, we executed the last step of our process, the inflection generation. Once the target sentence was ordered, we analyzed each token of the sentence and found its respective inflection word using the appropriate lexicon. It should be noted that there was no preference in inflection selection because we used our lexicon as a hash table, i.e., we were worried about the occurrence of the lemma and the morphological information to get the inflection.

Finally, we applied some rules to handle contractions and other types of problems (as the use of commas).

3 Results and Analysis

The performance of the methods in the Task 1 was computed using the following four metrics:

- BLEU (Papineni et al., 2002): precision metric that computes the geometric mean of the n-gram precisions between the generated text and reference texts, adding a brevity penalty for shorter sentences. We use the smoothed version and report results for n = 1, 2, 3, and 4;
- NIST (Doddington, 2002): related n-gram similarity metric weighted in favor of less frequent n-grams, which are taken to be more informative;
- CIDEr (Vedantam et al., 2015): designed for image description, and similar in spirit to NIST (in that it assigns lower weights to n-grams that are common to the reference texts) (determined by using TF-IDF metric);
- Normalized edit distance (DIST): inverse, normalized, character-based string-edit distance that starts by computing the minimum

[5] UDPipe is a trainable pipeline for tokenization, part of speech tagging, lemmatization and dependency parsing of CoNLL files. It contains models for several languages. It is available at http://ufal.mff.cuni.cz/udpipe.

number of character insertions, deletions and substitutions (all at cost 1) required to turn the system output into the (single) reference text.

For now, only the results for BLEU, NIST and DIST have been released. The results of our method for the test data are shown in Table 1, as well as the average results for all the systems that participated in the track. One may see that our method outperformed the average for each language.

Some examples of the results obtained for English, Portuguese and Spanish are shown in Table 2. As we may see, in sentence 1 for English, Portuguese and Spanish, the generated sentences were exactly the same as the reference. This may be explained by the short size of the sentences (excepting for Spanish, whose sentence is not so short).

In sentence 3 for English and 2 and 3 for Spanish, we may see that, even though the results were not correct (in relation to the ordering), some fragments could make sense ("The stocking for my 150 gallon tank is here..." in sentence 3 for English) and, sometimes, texts are still understandable (like sentences 2 and 3 for Spanish), preserving the overall meaning of the sentence.

We could also realize some limitations in our proposal. Firstly, we had some troubles with the software for lexicon building and it was necessary to review and correct some entries. For example, the sentence 2 in English contains the word "v" and the correct word was "have", and the sentence 2 in Portuguese shows the word "levá" and the correct word should be "levar".

Another limitation is related to the number of children in each level of the syntactic tree. In cases where the root of a sub-tree had several children, the seq-to-seq model returned incomplete sequences and the post-processing had more work to do, and, therefore, it usually performed poorly. For example, in sentence 3 for Portuguese, the syntactic tree has "Holland" as root in a level and "Spain", "Itália", "Belgium", ",", "or", and "em" are its children, and the result was not in correct order. Besides, a higher number of punctuations, missing words and proper nouns produced some mistakes in some cases, like sentence 2 for Spanish.

4 Conclusions and Future Work

In this paper, we presented a neural-based surface generation method that works at the syntactic level to order the words. Overall, our method outperformed the average results for English, Portuguese and Spanish. For Portuguese, the language in which we are particularly interested, we produced the best results for the NIST metric (although there is no statistical difference in relation to the system in the second place), and the second best results for BLEU and DIST, which we consider to be very good results.

Among the positive aspects, we noted that our method works fine when the length of the sentence is not too long. Furthermore, even though the results were not correct in some cases (in relation to the ordering), some fragments could make sense and, sometimes, texts were still understandable.

As future work, we may mention the review of the lexicons and possibly the implementation of a better inflection generator. Moreover, we would like to explore algorithms to deal with punctuations in order to improve the performance of our method.

Acknowledgments

The authors are grateful to FAPESP, CAPES and CNPq for supporting this work.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Anja Belz, Michael White, Dominic Espinosa, Eric Kow, Deirdre Hogan, and Amanda Stent. 2011. The first surface realisation shared task: Overview and evaluation results. In *Proceedings of the 13th European Workshop on Natural Language Generation*, ENLG '11, pages 217–226, Stroudsburg, PA, USA. Association for Computational Linguistics.

Cristian Cardellino. 2016. Spanish Billion Words Corpus and Embeddings.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, and Phillipp Koehn. 2013. One billion word benchmark for measuring progress in statistical language modeling. *CoRR*, abs/1312.3005.

Andre Luiz Verucci da Cunha. 2016. Coh-Metrix-Dementia: análise automática de distúrbios de linguagem nas demências utilizando Processamento de Línguas Naturais. Master's thesis, Instituto de

Language	BLEU	NIST	DIST	AVG BLEU	AVG NIST	AVG DIST
English	50.74	10.62	77.56	41.30	10.15	67.86
Portuguese	27.12	7.56	57.43	24.71	7.36	55.30
Spanish	51.58	11.17	53.78	33.66	9.01	35.65

Table 1: Achieved results

Language	Reference	Output
English	(1) Iran says it is creating nuclear energy without wanting nuclear weapons. (2) You have to see these slides....they are amazing. (3) Here is the stocking for my 150 gallon tank i upgraded it to 200 at the weekend because of the clownloach A 200 gallon with 6 pairs of Breeding Angel fish fire mouth honey Gouramis 5 8 inch clownloach a Krib and 5 1 inch clown loach with 16 cory cats 5 Australian Rainbows	(1) Iran says it is creating nuclear energy without wanting nuclear weapons. (2) You *v* to saw these slides.... they're amazing. (3) The stocking for my 150 gallon tank is here *at the weekend because of the clownloach i upgraded it to 200 an 200 gallon 8 inch clownloach 5 with an krib pairs fire mouth honey gourami 6 with 16 cory cats 5 australian rainbow of breeding angel fishes loach and 5 clown 1 inch*
Portuguese	(1) “Vivo num Estado de Ironia”. (2) Gosto de levar a sério o meu papel de consultor encartado. (3) Na Holanda, Bélgica, Itália e Espanha, os números oscilam entre 250 mil e 300 mil muçulmanos.	(1) ”vivia num estado de ironia”. (2) Gosto de *levá* a sério a *seu* papel consultor de encartado. (3) *, nas e Espanha Holanda Itália, Bélgica,* os números oscilam entre 250 mil e 300 mil muçulmanos.
Spanish	(1) El IMIM sólo controla muestras remitidas por el COI y de competiciones extranjeras. (2) Tras la violación, la mujer fue a interponer una denuncia en comisaría, ”pero como sufría hemorragias y pérdida de conocimiento, la propia policía llamó a una ambulancia y la envió al Hospital La Paz”. (3) El COI abrió ayer, por orden de su presidente, el belga Jacques Rogge, una investigación al descubrir, por casualidad, material médico para realizar transfusiones, bolsas vacías de sangre y restos de glucosa en una casa alquilada, en Soldier Hollow, muy cerca de Salt Lake City, por el equipo de fondo de la Federación Austriaca de Esquí durante la disputa de los recientes JJOO.	(1) El IMIM sólo controla muestras remitidas por el COI y de competiciones extranjeras. (2) La mujer fue a interponer una denuncia en comisaría *tras la violación,”., pero,* como *sufriría* hemorragias y pérdida de conocimiento”la propia policía llamó a una ambulancia y la envió al Hospital La Paz (3) El COI abrió ayer *una investigación* por orden de su presidente, el belga Jacques Rogge,*, al descubrir, por casualidad, material médico, bolsas vacías de sangre y restos de glucosa para transfusiones realizamos* en una casa *alquilado* por el equipo de fondo de la Federación Austriaca de Esquí durante la disputa de los recientes JJOO, en Soldier Hollow, *mucho* cerca de Salt Lake City,.

Table 2: Examples of generation for English, Portuguese and Spanish

Ciências Matemáticas e de Computação - Universidade de São Paulo, Brasil.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the Second International Conference on Human Language Technology Research*, HLT '02, pages 138–145, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Nathan Hartmann, Erick Fonseca, Christopher Shulby, Marcos Treviso, Jéssica Silva, and Sandra Aluísio. 2017. Portuguese word embeddings: Evaluating on word analogies and natural language tasks. In *Proceedings of the 11th Brazilian Symposium in Information and Human Language Technology*, pages 122–131. Sociedade Brasileira de Computação.

Eva Hasler, Felix Stahlberg, Marcus Tomalin, Adrià de Gispert, and Bill Byrne. 2017. A comparison of neural models for word ordering. In *Proceedings of the 10th International Conference on Natural Language Generation, INLG 2017, Santiago de Compostela, Spain, September 4-7, 2017*, pages 208–212.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. Opennmt: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72. Association for Computational Linguistics.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the tenth Machine Translation Summit*, pages 79–86, Phuket, Thailand. AAMT, AAMT.

Simon Mille, Anja Belz, Bernd Bohnet, Yvette Graham, Emily Pitler, and Leo Wanner. 2018. The First Multilingual Surface Realisation Shared Task (SR'18): Overview and Evaluation Results. In *Proceedings of the 1st Workshop on Multilingual Surface Realisation (MSR), 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1–10, Melbourne, Australia.

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, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press, New York, NY, USA.

Allen Schmaltz, Alexander M. Rush, and Stuart M. Shieber. 2016. Word ordering without syntax. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2319–2324. The Association for Computational Linguistics.

Milan Straka and Jana Straková. 2017. Tokenizing, pos tagging, lemmatizing and parsing ud 2.0 with udpipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 3104–3112, Cambridge, MA, USA. MIT Press.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 4566–4575.

Yue Zhang and Stephen Clark. 2015. Discriminative syntax-based word ordering for text generation. *Computational Linguistics*, 41(3):503–538.