# AX Semantics' Submission to the Surface Realization Shared Task 2018

**Andreas Madsack**, **Johanna Heininger**, **Nyamsuren Davaasambuu**,
**Vitaliia Voronik**, **Michael Käufl**, and **Robert Weißgraeber**

AX Semantics, Stuttgart, Germany
{firstname.lastname}@ax-semantics.com

## Abstract

In this paper we describe our system and experimental results on the development set of the Surface Realisation Shared Task (Mille et al., 2017). Our system is an entry for Shallow-Task, with two different models based on deep-learning implementations for building the sentences combined with a rule-based morphology component. We trained our systems on all 10 given languages.

## 1 Introduction

This paper describes our approach for the First Multilingual Surface Realisation Shared Task (Mille et al., 2018). For the surface task the dependency parse trees were given unordered and the words lemmatized. The objective was to order the words in the sentences and to inflect the given lemmas. The data was provided in 10 languages: English, Spanish, French, Portuguese, Italian, Dutch, Czech, Russian, Arabic, and Finnish.

Our aim was to build new deep learning based ordering systems, augmented by using our already implemented (rule-based) morphology for the inflection part. System 1 implemented the initial idea and system 2 followed after mediocre results in system 1.

Final scoring for the MSR shared task was using System 2.

## 2 Linearization

Here we propose two systems: both are implemented using Keras (Chollet et al., 2015) and Tensorflow (Abadi et al., 2016), are trained using each language from the CoNLL data sets separately and finally also trained with all languages combined. These two systems, however, differ in their internal models (see the following two sections).

To generate training data the given training CoNLL data sets were matched to their corresponding original data using tree based matching. Each node was compared based on `deprel`, `lemma/form`, `upostag`, and number of children in a recursive manner traversing the tree from top to bottom.

### 2.1 System 1: Sequence-to-Sequence Model

System 1 is a new approach using sequence-to-sequence models (Vinyals et al., 2016), encoder-decoder, and attention as described in Bahdanau et al. (2014) for machine translation. Instead of using LSTM cells, we used bidirectional GRU cells (Cho et al., 2014). Some early stage evaluations showed GRU converges better than LSTM for this task.

The input sequence is an unordered list of words and their features; the features for each word consist of: `id`, `upostag`, `deprel`, `head-id`, `head-upostag`, `head-deprel`, and `level` in the syntax-tree. All features are encoded in embeddings. The embeddings are shared between the two matching fields (i.e. `deprel` and `head-deprel`). Figure 1 shows a visualization of the model.

The result of the sequence model is a sequence of correct positions of the words for a complete sentence. This order, together with the given lemma and features from the data set, is then processed by a morphology component, which also takes care of building the "final readable sentence" including e.g. capitalization.

We trained two sub-models for each language with the sequence lengths of 25 and 400. We chose these values based on the length of the sentences in the training data set – the 75% quantile is at length 25 which includes most of the sentences. 400 is the absolute maximal length of sentences (the longest sentence has 398 words and is

*Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 54–57
Melbourne, Australia, July 19, 2018.

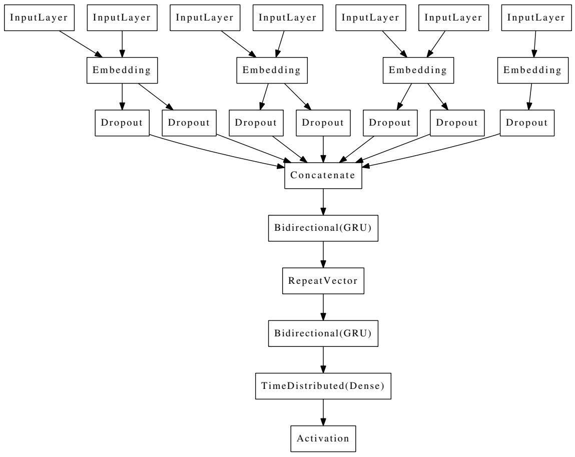

Figure 1: Sequence-to-Sequence Model
The inputs are: (`id`, `head-id`), (`upos`, `head-upos`), (`deprel`, `head-deprel`), (`level`)

in Arabic). We used 0 value padding for sequences shorter than the maximum given by the model.

These two sub-models are then available for the prediction phase, during which the model is chosen by the length of input sentence being shorter than that of the next fitting model. The predicted sequence probabilities are selected so that every word appears only once in the final sentence.

Automatic evaluation of the dev-set resulted in BLEU scores and DIST scores given in table 1. We used the evaluation code given by the shared task organizers. This evaluation step includes the morphology described in section 3. We used the matching model for the language and a model trained with all languages.

| lang | BLEU<br>(language) | BLEU<br>(ALL) | DIST<br>(language) | DIST<br>(ALL) |
|---|---|---|---|---|
| en | 0.020 | 0.019 | 0.124 | 0.133 |
| es | 0.007 | 0.007 | 0.071 | 0.071 |
| fr | 0.009 | 0.009 | 0.094 | 0.096 |
| pt | 0.012 | 0.013 | 0.106 | 0.107 |
| it | 0.007 | 0.008 | 0.095 | 0.098 |
| nl | 0.010 | 0.013 | 0.099 | 0.102 |
| cs | 0.009 | 0.011 | 0.082 | 0.085 |
| ru | 0.009 | 0.007 | 0.073 | 0.076 |
| ar | 0.002 | 0.003 | 0.071 | 0.072 |
| fi | 0.010 | 0.012 | 0.083 | 0.084 |

Table 1: Scores for Sequence-to-Sequence Model (development data)

### 2.2 System 2: Pairwise Classification

The second system is a classification model that calculates the word ordering by estimating if word1 is right of word2. Each word of a sentence is calculated against every other word in the same sentence. Features used in training for each of the two words are `upostag`, `deprel`, `head-upostag`, `head-deprel` and `level` in the syntax-tree. Same as System 1 the embeddings are shared between the two matching fields.

The predicted word1-is-right-of-word2 probabilities are used for each subtree to find the order. On the next level the subtree is ordered by the probability of the head node of the subtree.

The results show that particularly `upostag=PUNCT` is now mostly at the end of sentences even for commas and other punctuations. Human inspection results in a positively increased overall readability of the output compared to the Sequence-to-Sequence Model (our System 1). See table 2 for results on the given dev-set.

Like the Sequence-to-Sequence model, we have evaluated this model using the matching language and a model trained on all languages.

| lang | BLEU<br>(language) | BLEU<br>(ALL) | DIST<br>(language) | DIST<br>(ALL) |
|---|---|---|---|---|
| en | 0.205 | 0.175 | 0.430 | 0.354 |
| es | 0.100 | 0.139 | 0.182 | 0.273 |
| fr | 0.154 | 0.137 | 0.190 | 0.308 |
| pt | 0.153 | 0.137 | 0.314 | 0.308 |
| it | 0.105 | 0.106 | 0.309 | 0.258 |
| nl | 0.161 | 0.123 | 0.298 | 0.270 |
| cs | 0.099 | 0.110 | 0.279 | 0.235 |
| ru | 0.239 | 0.142 | 0.260 | 0.245 |
| ar | 0.044 | 0.059 | 0.163 | 0.199 |
| fi | 0.078 | 0.064 | 0.197 | 0.223 |

Table 2: Scores for Pairwise Classification Model (development data)

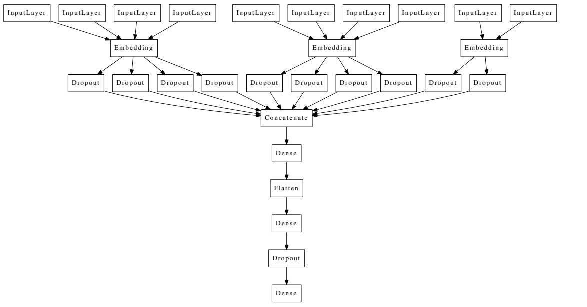

Figure 2: Pairwise Classification Model
The inputs are: (`w1-upos`, `w1-head-upos`, `w2-upos`, `w2-head-upos`), (`w1-deprel`, `w1-head-upos`, `w2-deprel`, `w2-head-upos`), (`w1-level`, `w2-level`)

| | |
|---|---|
| Reference | From the AP comes this story: |
| System 1 | Comes from story this AP the: |
| System 2 | This story comes from the AP: |
| Reference | I took my Mustang here and it looked amazing after they were done, they did a great job, I'm very satisfied with the results. |
| System 1 | With results job I my took satisfied it amazing here did very Mustang. great I, they were a are after looked done the, and they |
| System 2 | I took here Mustang my looked it amazing after they done were and they did a great job I are very satisfied the with results,,. |
| Reference | Lopulta saatiin halikuva otettua |
| System 1 | Sadaan lopulta ottattua halikuva. |
| System 2 | Lopulta sadaan ottattua halikuva. |
| Reference | Pastrana begon een politiek offensief om de Copa voor Colombia te behouden. |
| System 1 | Voor beginde. offensief te politiek een Pastrana Colombia Copa om behouden de |
| System 2 | Pastrana om behouden te Copa de voor Colombia beginde offensief een politiek. |

Table 3: Example outputs

## 3 Morphologization

The morphology step employs the NLG system from AX Semantics (Weißgraeber and Madsack, 2017). That system is rule-based and for each inflection request it runs through a decision chain, in which all parts of speech and corresponding grammatical features of the specific languages are implemented.

For irregular words the AX Semantics NLG system uses lexicon entries, which always supercede the rule-based inflection. Grammatical features like number, case, animacy and tense are implemented in a general way, then added to each language alongside its individual configuration.

Since the CoNLL features differ from our usual input parameters, some preprocessing was necessary to map the terms accordingly. The words were also cleaned with regard to special characters like hash tags or diacritics before they were processed by the NLG morphology component.

The accuracy of the morphology component was tested separately on the dev-set for each language. Results are summarized in table 4. Most of the languages show a decent accuracy score of over 90%, whereas Arabic and Finnish with their more complicated morphology still achieve around 80%.

The table also shows that for some languages the accuracy scores for verbs are significantly lower than for nouns or adjectives. For example, in case of Dutch this happens mainly because a given lemma is not the infinitive form as expected from our system but a finite verb form (3rd person singular) and first has to be transformed to the infinitive. This can largely be attributed to the specialization of the system for the language of commerce, which results in a partial under-coverage of certain language features for edge cases. We expect coverage to increase as usage expands to more fields.

| language | nouns | adjectives | verbs | mean |
|---|---|---|---|---|
| en | 0.90 | 0.92 | 0.93 | 0.94 |
| es | 0.94 | 0.96 | 0.86 | 0.94 |
| fr | 0.94 | 0.93 | 0.81 | 0.94 |
| pt | 0.91 | 0.95 | 0.79 | 0.95 |
| it | 0.94 | 0.95 | 0.77 | 0.93 |
| nl | 0.86 | 0.86 | 0.50 | 0.90 |
| cs | 0.82 | 0.91 | 0.88 | 0.91 |
| ru | 0.92 | 0.91 | 0.60 | 0.90 |
| ar | 0.73 | 0.69 | 0.40 | 0.81 |
| fi | 0.60 | 0.65 | 0.62 | 0.79 |

Table 4: Accuracy of the morphology step (examples for single POS categories and mean overall accuracy)

Furthermore, some of the errors are due to the data being erroneous or incomplete (e.g., only case is given, when number and animacy would also be needed).

## 4 Conclusion and Future Work

On the whole, none of the systems solve the task satisfactorily.

System 2 shows better scores and somewhat improved readability in contrast to System 1. See table 3 for illustration.

In both linearization systems, we use neither the lemma nor an embedding of the lemma to allow a comparison between the language models and the ALL-language model. This serves as a baseline for comparison against systems where language-specific features can be added.

Our focus for this workshop was to build a linearization system that is simple and does not receive any topic-specific or language-specific input data nor configurations, and without building a neuronal network for morphologization. For pure morphologization tasks, especially for Finnish, Arabic and Hungarian with a large list of very rare cases, we will improve inflection by adding a NN-based morphology component as well.

## References

Martin Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek G. Murray, Benoit Steiner, Paul Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2016. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 265–283.

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

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078.

François Chollet et al. 2015. Keras. `https://keras.io`.

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.

Simon Mille, Bernd Bohnet, Leo Wanner, and Anja Belz. 2017. Shared task proposal: Multilingual surface realization using universal dependency trees. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 120–123. Association for Computational Linguistics.

Oriol Vinyals, Samy Bengio, and Manjunath Kudlur. 2016. Order matters: Sequence to sequence for sets. In *International Conference on Learning Representations (ICLR)*.

Robert Weißgraeber and Andreas Madsack. 2017. A working, non-trivial, topically indifferent nlg system for 17 languages. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 156–157. Association for Computational Linguistics.