HAL open science

Surface Realisation Using Full Delexicalisation

Anastasia Shimorina, Claire Gardent

To cite this version:

Anastasia Shimorina, Claire Gardent. Surface Realisation Using Full Delexicalisation. 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Nov 2019, Hong Kong, China. pp.3084-3094, 10.18653/v1/D19-1305 . hal-02277069

HAL Id: hal-02277069

https://hal.science/hal-02277069v1

Submitted on 3 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Surface Realisation Using Full Delexicalisation

Anastasia Shimorina
LORIA / Lorraine University
anastasia.shimorina@loria.fr

Claire Gardent
LORIA / CNRS
claire.gardent@loria.fr

Abstract

Surface realisation (SR) maps a meaning representation to a sentence and can be viewed as consisting of three subtasks: word ordering, morphological inflection and contraction generation (e.g., clitic attachment in Portuguese or elision in French). We propose a modular approach to surface realisation which models each of these components separately, and evaluate our approach on the 10 languages covered by the SR'18 Surface Realisation Shared Task shallow track. We provide a detailed evaluation of how word order, morphological realisation and contractions are handled by the model and an analysis of the differences in word ordering performance across languages.

1 Introduction

Surface realisation maps a meaning representation to a sentence. In data-to-text generation, it is part of a complex process aiming to select, compress and structure the input data into a text. In text-to-text generation, it can be used as a mean to rephrase part or all of the input content. For instance, Takase et al. (2016) used surface realisation to generate a summary based on the meaning representations of multiple input documents and Liao et al. (2018) to improve neural machine translation.

By providing parallel data of sentences and their meaning representation, the SR'18 Surface Realisation shared task (Mille et al., 2018) allows for a detailed evaluation and comparison of surface realisation models. Moreover, as it provides training and test data for multiple languages, it also allows for an analysis of how well these models handle languages with different morphological and topological properties.

The SR'18 shared task includes two tracks: a shallow track where the input is an unordered, lemmatised dependency tree and a deep track where function words are removed and syntactic relations are replaced with semantic ones. In this paper, we focus on the shallow track of the SR'18 Shared Task and we propose a neural approach which decomposes surface realisation into three subtasks: word ordering, morphological inflection and contraction generation (e.g., clitic attachment in Portuguese or elision in French). We provide a detailed analysis of how each of these phenomena (word order, morphological realisation and contraction) is handled by the model, and we discuss the differences between languages.

For reproducibility, all our experiments including data and scripts are available at https://gitlab.com/shimorina/emnlp-2019.

2 Related Work

Early approaches for surface realisation adopted statistical methods, including both pipelined (Bohnet et al., 2010) and joint (Song et al., 2014; Puduppully et al., 2017) architecture for word ordering and morphological generation.

Multilingual SR'18 was preceded by the SR'11 surface realisation task for the English language only (Belz et al., 2011). The submitted systems in 2011 had grammar-based and statistical nature, mostly relying on pipelined architecture. Recently, Marcheggiani and Perez-Beltrachini (2018) proposed a neural end-to-end approach based on graph convolutional encoders for the SR'11 deep track.

The SR'18 shallow track received submissions from eight teams with seven of them dividing the task into two subtasks: word ordering and inflection. Only Elder and Hokamp (2018) developed a joint approach, however, they participated only in the English track.

For word ordering, five teams chose an approach based on neural networks, two used a

classifier, and one team resorted to a language model. As for the inflection subtask, five teams applied neural techniques, two used lexicon-based approaches, and one used an SMT system (Basile and Mazzei, 2018; Castro Ferreira et al., 2018; Elder and Hokamp, 2018; King and White, 2018; Madsack et al., 2018; Puzikov and Gurevych, 2018; Singh et al., 2018; Sobrevilla Cabezudo and Pardo, 2018). Overall, neural components were dominant across all the participants. However, official scores of the teams that went neural greatly differ. Furthermore, two teams (Elder and Hokamp, 2018; Sobrevilla Cabezudo and Pardo, 2018) applied data augmentation, which makes their results not strictly comparable to others.

One of the interesting findings of the shared task is reported by Elder and Hokamp (2018) who showed that applying standard neural encoder-decoder models to jointly learn word ordering and inflection is highly challenging; their sequence-to-sequence baseline without data augmentation got 43.11 BLEU points on English.

Our model differs from previous work in three main ways. First, it performs word ordering on fully delexicalised data. Delexicalisation has been used previously but mostly to handle rare words, e.g. named entities. Here we argue that surface realisation and, in particular, word ordering works better when delexicalising all input tokens. This captures the intuition that word ordering is mainly determined by the syntactic structure of the input. Second, we we provide a detailed evaluation of how our model handles the three subtasks underlying surface realisation. While all SR'18 participants provided descriptions of their models, not all of them performed an in-depth analysis of model performance. Exceptions are works of King and White (2018), who provided a separate evaluation for the morphological realisation module, and Puzikov and Gurevych (2018), who evaluated both word ordering and inflection modules. However, it is not clear how each of those modules affect the global performance when merged in the full pipeline. In contrast, we propose a detailed incremental evaluation of each component of the full pipeline and show how each component impacts the final scores. Third, we introduce a linguistic analysis, based on the dependency relations, of the word ordering component, allowing for deeper error analysis of the developed systems.

Furthermore, our model explicitly integrates a module for contraction handling, as done also before by Basile and Mazzei (2018). We also address all the ten languages proposed by the shared task and outline the importance of handling contractions.

3 Data

The SR'18 data (shallow track) is derived from the ten Universal Dependencies (UD) v2.0 treebanks (Nivre et al., 2017) and consists of (T, S) pairs where S is a sentence, and T is the UD dependency tree of S after word information has been removed and tokens have been lemmatised. The languages are those shown in Table 1 and the size of the datasets (training, dev and test) varies between 7,586 (Arabic) and 85,377 (Czech) instances with most languages having around 12K instances (for more details about the data see Mille et al. (2018)).

4 Model

As illustrated by Example 1, surface realisation from SR'18 shallow meaning representations can be viewed as consisting of three main steps: word ordering, morphological inflection and contraction generation. For instance, given an unordered dependency tree whose nodes are labelled with lemmas and morphological features (1a)[1], the lemmas must be assigned the appropriate order (1b), they must be inflected (1c) and contractions may take place (1d).

(1) a. the find **be not** meaning of life it about
b. it be not about find the meaning of life
c. It **is n't** about finding the meaning of life
d. It **isn't** about finding the meaning of life

We propose a neural architecture which explicitly integrates these three subtasks as three separate modules into a pipeline: word ordering (WO) is applied first, then morphological realisation (WO+MR) and finally, contractions (WO+MR+C) are handled.

4.1 Word Ordering

For word ordering, we combine a factored sequence-to-sequence model with an "extreme delexicalisation" step which replaces matching source and target tokens with an identifier.

[1] Features and tree structures have been omitted.

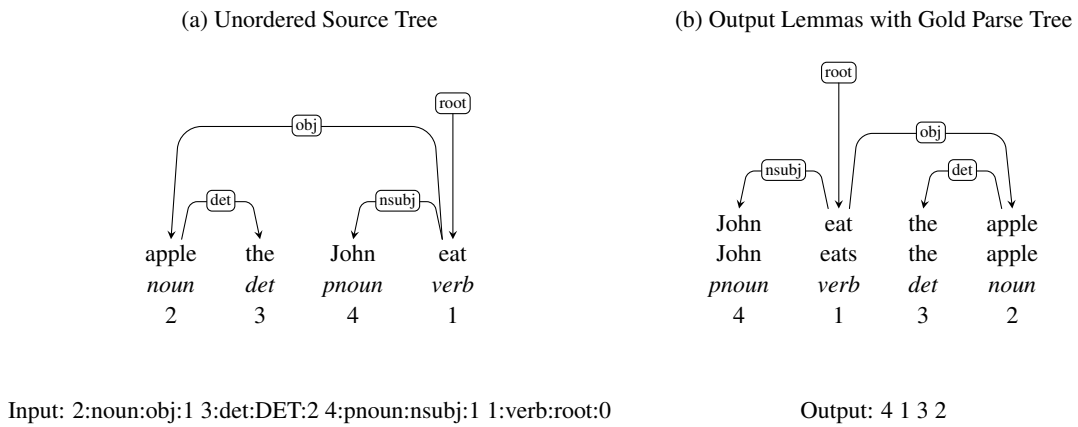

Figure 1: Delexicalising and linearising (in the parse tree of the output sentence the first row shows the lemmas, the second–the word forms, the third–the POS tags and the fourth–the identifiers). Identifiers are assigned to the source tree nodes in the order given by depth-first search.

Delexicalisation. Delexicalisation has frequently been used in neural NLG to help handle unknown or rare items (Wen et al., 2015; Dušek and Jurcicek, 2015; Chen et al., 2018). Rare items are replaced by placeholders both in the input and in the output; models are trained on the delexicalised data; and a post-processing step ensures that the generated text is relexicalised using the placeholders' original value. In these approaches, delexicalisation is restricted to rare items (named entities). In contrast, we apply delexicalisation to all input lemmas. Abstracting away from specific lemmas reduces data sparsity, allows for the generation of rare or unknown words and last but not least, it captures the linguistic intuition that word ordering mainly depends on syntactic information (e.g., in English, the subject generally precedes the verb).

To create the delexicalised data, we need to identify matching input and output elements and to replace them by the same identifier. We also store a mapping (id, L, F) specifying which identifier id refers to which (L, F) pair, where L is a lemma and F is its set of morpho-syntactic features.

We identify matching input and output elements by comparing the unordered input tree provided by the SR'18 task with the parse tree of the output sentence provided by the UD treebanks (cf. Figure 1). Source and target nodes which share the same path to the root are then mapped to the same identifier. For instance, in Figure 1, the lemma "apple" has the same path to the root (obj:eat:root) in both the input and the output tree. Hence the same identifier is assigned to the nodes. More generally, after linearisation through depth-first, left-to-right traversal of the input tree, each training instance captures the mapping between lemmas in the input tree and the same lemmas in the output sequence. For instance, given the example shown in Figure 1, delexicalisation will yield the training instance:

Input: tkn2 tkn3 tkn4 tkn1
Output: tkn4 tkn1 tkn3 tkn2

where tkn_i is the factored representation (see below) of each delexicalised input node.

Factored Sequence-to-Sequence Model. Following Elder and Hokamp (2018), we use a factored model (Alexandrescu and Kirchhoff, 2006) as a means of enriching the node representations input to the neural model. Each delexicalised tree node is modelled by a sequence of features. Separate embeddings are learned for each feature type and the feature embeddings of each input node are concatenated to create its dense representation. As exemplified in Figure 1, we model each input placeholder as a concatenation of four features: the node identifier, its POS tag, its dependency relation to the parent node and its parent identifier[2].

Sequence-to-sequence model. We use the OpenNMT-py framework (Klein et al., 2017)[3] to train factored sequence-to-sequence models with attention (Luong et al., 2015) and the copy and coverage mechanisms described in See et al. (2017). A single-layer LSTM is used for both encoder and decoder. We train using full vocabulary

[2]The parent identifier of a root node is represented as 0.

[3]commit e61589d, https://github.com/OpenNMT/OpenNMT-py

and the maximal length in the source and target for both baseline and the proposed model. Models were trained for 20 epochs, with a mini-batch size of 64, a word embedding size of 300, and a hidden unit size of 450. They were optimised with SGD with a starting learning rate of 1.0. A learning rate is halved when perplexity does not decrease on the development set. Preliminary experiments showed that the lowest perplexity was reached on average at epoch 17, so this model was kept for decoding. Decoding is done using beam search with a beam size of 5. For each language, we train three models with different random seeds, and report the average performance and standard deviation.

The model is trained on delexicalised data. At test time, the token/identifier mapping is used to relexicalise the model output.

4.2 Morphological Realisation

The morphological realisation (MR) module consists in producing inflected word forms based on lemmas coupled with morphological features. For that module, we used a model recently proposed by Aharoni and Goldberg (2017), which achieves state-of-the-art results on several morphological inflection datasets: the CELEX dataset (Baayen et al., 1993; Dreyer et al., 2008), the Wiktionary dataset (Durrett and DeNero, 2013) and the SIGMORPHON2016 dataset (Cotterell et al., 2016). Their model is based on a neural encoder-decoder architecture with hard monotonic attention and performs out-of-context morphologic realisation: given a lemma and a set of morpho-syntactic features, it produces a corresponding word form.

We trained the model of Aharoni and Goldberg (2017) on (lemma+morpho-syntactic_features[4], form) pairs extracted from the SR'18 training data. We trained the model for 20 epochs with the default parameters provided in the implementation[5].

In our pipeline architecture, morphological realisation is applied to the output of our word ordering model using the (id, L, F) mapping mentioned above. For each delexicalised token produced by the word ordering component, we retrieve the corresponding lemma and morpho-syntactic features (L, F) and apply our MR model to it so as to produce the corresponding word form.

[4]POS and morphological features

[5]https://github.com/roeeaharoni/morphological-reinflection/blob/master/src/hard_attention.py

While associating a lemma and its features to a corresponding form, the MR module operates without taking context into account, so it cannot perform some finer grained operations, such as contraction, elision, and clitic attachment. We address that issue in the following section.

4.3 Contraction Generation

Contraction handling is the last step of our surface realisation pipeline. Example 2 shows some types of contractions.

(2) French: "Le chat dort." / "L'alouette chante." (Elision for the definite article *le* before a vowel: Le → L')

Italian: *"In il mare." → "Nel mare." (Contraction of the preposition *in* and the article *il*: In il → Nel)

Portuguese: *"Eis lo." → "Ei-lo." (Clitic pronoun attachment: Eis lo → Ei-lo)

We developed two modules for the contraction generation: one based on regular expressions (C_{reg}) and another based on a sequence-to-sequence model (C_{s2s}).

The sequence-to-sequence model is trained on pairs of sentences without and with contractions. The sentence with contraction (S^{+c}) is the final sentence, i.e., the reference sentence in the SR'18 data. The sentence without contraction (S^{-c}) is the corresponding sequence of word forms extracted from the UD CoNLL data.

The regular expression module is inspired by the decomposition of multi-word expressions, such as contractions, which is applied during the tokenisation step in parsing (Martins et al., 2009). We reversed the regular expressions given in the TurboParser[6] for the surface realisation task, and also added our own to tackle, for example, elision in French. C_{s2s} and C_{reg} modules were created for three languages: French, Italian, and Portuguese[7].

5 Evaluation

We evaluate each component of our approach separately. We start by providing a detailed evaluation of how the model handles word ordering

[6]https://github.com/andre-martins/TurboParser/tree/master/python/tokenizer

[7]Although contractions are also present in Spanish, we did not develop a module for it, since the UD Spanish AnCora treebank does not split them on the token level in contrast to other UD treebanks.

	ar	cs	en	es	fi	fr	it	nl	pt	ru
BL	29.6±1.39	48±0.7	53.57±0.15	46.5±0.78	27.2±0.4	46.4±0.5	49.07±0.9	36.6±0.7	44.3±0.26	58.1±0.46
WO	34.9±0.2	57.97±0.06	59.1±0.36	52.33±0.31	43.1±0.53	50.0±0.0	53.17±0.5	47.03±0.59	51.77±0.32	64.73±0.23
Δ	+5.3	+9.97	+5.53	+5.83	+15.9	+3.6	+4.1	+10.43	+7.47	+6.63

Table 1: Word Ordering: BLEU scores on lemmatised data. Mean and standard deviation across three random seeds. BL: Baseline. All pairwise comparisons of BL and our model showed a statistically significant difference in BLEU via the bootstrap resampling (1000 samples, $p < .05$).

(Section 5.1). We then go on to analyse the respective contributions of morphological realisation (Section 5.2) and contraction generation (Section 5.3). Finally, we discuss the performance of the overall surface realisation model (Section 5.4). Throughout the evaluation, we used the SR'18 evaluation scripts to compute automatic metrics[8].

5.1 Word Ordering

5.1.1 BLEU scores

We evaluate our word ordering component by computing the BLEU-4 score (Papineni et al., 2002) between the sequence of lemmas it produces and the lemmatized reference sentence extracted from the UD files. The baseline is the same model without delexicalisation. As Table 1 shows, there is a marked, statistically significant, difference between the baseline and our approach which indicates that delexicalisation does improve word ordering.

5.1.2 Word Ordering Constraints

We also investigate the degree to which our results conform with the word ordering constraints of the various languages focusing on the following dependency relations: DET (determiner), NSUBJ (nominal subject), OBJ (object), AMOD (adjectival modifier) and ACL (nominal clausal modifier). For each of these dependency relations, we compare the relative ordering of the corresponding (head, dependent) pairs in the reference data and in our system predictions.

To determine whether the dependent should precede or follow its head, we use the gold standard dependency tree of the UD treebanks. Since for the system predictions we do not have a parse tree, we additionally record the distance between head and dependent (in the reference data) and we compare it with the distance between the same two items in the system output. For instance, for the DET relation, given the gold sentence (3a) and the generated sentence (3b), we extract (3c) from the UD parse tree and (3d) from the predicted sentence where each triple is of the form either (dep, head, distance) or (head, dep, distance) and distance is the distance between head and dependent.

(3) a. GOLD: The yogi tried the advanced asana
b. PRED: The yogi tried the asana advanced
c. G-triples: (the$_{dep}$, yogi$_{head}$, 1), (the$_{dep}$, asana$_{head}$, 2)
d. P-triples: (the, yogi, 1), (the, asana, 1)
Exact match: 1; Approximate match: 2

We then compute exact matches (the order and the distance to the head is exactly the same) and approximate matches (the order is preserved but the distance differs by 1 token[9]). Table 2 shows the results and compares them with a non-delexicalised approach.

Global Score. The **all deprels** column summarises the scores for all dependency relations present in the treebanks (not just DET, NSUBJ, OBJ, AMOD and ACL). For the exact match, most languages score above average (from 0.51 to 0.71). That is the relative word order and the position of the dependent with respect to the head is correctly predicted in more than half of the cases. Approximate match yields higher scores with most languages scoring between 0.65 and 0.80 suggesting that a higher proportion of correct relative orderings is achieved (modulo mispositioning and false positives).

Long Range Dependencies. It is noticeable that for all languages, accuracy drops for the ACL relation. We conjecture that two factors makes it difficult for the model to make the correct prediction: heterogeneity and long range dependencies. As the ACL relation captures different types of clausal modifiers (finite and non-finite), it is harder for the model to learn the corresponding patterns. As the modifier is a clause, the distance between head

[8] http://taln.upf.edu/pages/msr2018-ws/SRST.html#evaluation

[9] We could of course consider further approximates matches differing by, say 2, 3 or 5 tokens. But we refrain from this as this would increase the number of false positives.

	det		*nsubj*		*obj*		*amod*		*acl*		***all deprels***	
	+1	+2	+1	+2	+1	+2	+1	+2	+1	+2	+1	+2
ar	0.36	0.37	0.45	0.52	0.35	0.48	0.52	0.59	0.32	0.41	0.38	0.47
Δ	+0.036	+0.047	-0.036	-0.067	-0.069	-0.113	-0.088	-0.106	-0.063	-0.061	-0.044	-0.071
cs	0.86	0.9	0.49	0.63	0.51	0.64	0.83	0.87	0.47	0.62	0.63	0.74
Δ	-0.077	-0.071	-0.041	-0.054	-0.048	-0.053	-0.134	-0.126	-0.071	-0.073	-0.068	-0.074
en	0.76	0.85	0.71	0.85	0.76	0.84	0.71	0.76	0.53	0.65	0.63	0.74
Δ	-0.10	-0.086	-0.002	-0.053	-0.074	-0.064	-0.15	-0.138	-0.058	-0.104	-0.06	-0.08
es	0.73	0.83	0.55	0.71	0.54	0.70	0.43	0.49	0.39	0.58	0.55	0.68
Δ	-0.02	-0.074	-0.012	-0.073	+0.004	-0.057	+0.059	+0.036	-0.056	-0.12	-0.038	-0.088
fi	0.71	0.81	0.64	0.75	0.46	0.60	0.70	0.76	0.50	0.61	0.51	0.65
Δ	-0.241	-0.244	-0.194	-0.189	-0.118	-0.137	-0.303	-0.296	-0.261	-0.312	-0.154	-0.16
fr	0.76	0.86	0.60	0.78	0.60	0.75	0.46	0.51	0.51	0.68	0.58	0.71
Δ	-0.016	-0.235	+0.014	-0.037	+0.031	-0.029	+0.117	+0.104	-0.1	0.17	-0.038	-0.093
it	0.73	0.82	0.59	0.70	0.58	0.73	0.40	0.46	0.52	0.65	0.56	0.69
Δ	-0.022	-0.058	-0.021	-0.07	-0.044	-0.101	+0.058	+0.017	-0.054	-0.072	-0.04	-0.088
nl	0.71	0.79	0.46	0.56	0.38	0.49	0.74	0.77	0.41	0.53	0.49	0.60
Δ	-0.121	-0.115	-0.068	-0.07	-0.068	-0.087	-0.278	-0.29	-0.22	-0.277	-0.118	-0.13
pt	0.74	0.80	0.56	0.72	0.57	0.73	0.42	0.44	0.48	0.63	0.54	0.67
Δ	-0.032	-0.051	-0.072	-0.114	-0.041	-0.094	+0.034	+0.043	-0.122	-0.204	-0.068	-0.103
ru	NA	NA	0.65	0.79	0.65	0.74	0.79	0.85	0.45	0.67	0.71	0.80
Δ	—	—	-0.033	-0.057	-0.037	-0.033	-0.095	-0.1	+0.022	-0.027	-0.05	-0.059

Table 2: Proportion of correct head/dependent positioning for the five selected dependency relations: *det*, *nsubj*, *obj*, *amod*, *acl*, and overall performance across all dependency relations. +1: exact match; +2: approximate match, i.e. head and dependent are in the correct order but there is a one-token difference between gold and prediction. NA: no dependency relation found in a treebank. Δ indicates the difference between our delexicalised model and the baseline.

	ar	cs	en	es	fi	fr	it	nl	pt	ru
MR Accuracy	91.05	98.89	98.3	99.07	92.66	96.77	96.85	87.6	98.80	98.23
WO	34.9±0.2	57.97±0.06	59.1±0.36	52.33±0.31	43.1±0.53	50.0±0.0	53.17±0.5	47.03±0.59	51.77±0.32	64.73±0.23
WO+MR (S^{-c})	28.6±0.26	56.1±0.1	54.3±0.3	51.4±0.3	38.63±0.59	44.97±0.25	47.3±0.6	42.3±0.53	50.7±0.26	60.93±0.23
Δ	-6.3	-1.87	-4.8	-0.93	-4.47	-5.03	-5.87	-4.73	-1.07	-3.8

Table 3: Morphological Realisation Results. MR Accuracy: accuracy of the MR module. WO: BLEU scores on lemmas. WO+MR: BLEU scores on inflected tokens.

(the nominal being modified) and dependent (the verb of the clause modifier) can be long which again is likely to impede learning.

Irregular Order. For cases where the head/dependent order is irregular, the scores are lower. For instance, in Dutch the object may occur either before (46.9% of the cases in the test data) or after the verb depending on whether it occurs in a subordinate or a main clause. Relatedly, the OBJ exact match score is the lowest (0.38) for this language. Similarly, in Romance languages where the adjective (AMOD relation) can either be pre- (head-final construction, HF) or post-posed (head-initial construction, HI), exact match scores are lower for this relation than for the others. For instance, the Portuguese test data contains 71% HF and 29% HI occurrences of the AMOD relation and correspondingly, the scores for that relation are much lower than for the DET, NSUBJ and OBJ relations for that language. A similar pattern can be observed for Spanish, French and Italian.

More detailed statistics, including other relations and performance with respect to head-directionality, can be found in the supplementary material.

Non-delexicalised Baseline. We also compare our delexicalised model with the non-delexicalised baseline: Δ in Table 2 shows the difference in performance between the two models.

Overall, the scores favour the delexicalised approach (negative $delta$ in the **all deprels** column for all languages) supporting the results given by the automatic metric. However, for some dependency relations, the lexicalised baseline shows usefulness of word information, for ex-

	ar	cs	en	es	fi	fr	it	nl	pt	ru
S^{-c}/S^{+c}	47.1	96.1	90.9	98.4	98.5	70.8	65.1	99.5	66.3	96.9
WO+MR (S^{-c})	28.6±0.26	56.1±0.1	54.3±0.3	51.4±0.3	38.63±0.59	44.97±0.25	47.3±0.6	42.3±0.53	50.7±0.26	60.93±0.23
WO+MR (S^{+c})	15.8±0.1	54.83±0.21	53.3±0.53	51.03±0.31	38.37±0.55	36.23±0.42	31.5±0.4	42.27±0.55	34.0±0.62	60.43±0.15
Δ	-12.8	-1.27	-1.0	-0.37	-0.26	-8.74	-15.8	-0.03	-16.7	-0.5
WO+MR+C_{reg} (S^{+c})						41.8±0.26	40.0±0.66		46.13±0.29	
WO+MR+C_{s2s} (S^{+c})						40.33±0.36	39.93±0.17		44.57±0.64	

Table 4: Contraction Generation Results (BLEU scores). S^{-c}/S^{+c}: a sentence without contractions vs. a reference sentence including contractions; S^{-c}: BLEU with respect to sentences before contractions; S^{+c}: BLEU with respect to a reference sentence. The scores were computed on detokenised sequences.

ample, while predicting AMOD relations for Romance languages (positive $delta$ for French, Italian, Spanish, and Portuguese). Indeed, preposed adjectives in those languages constitute a limited lexical group.

5.2 Morphological Realisation

Table 3 shows the results for the WO+MR model.

The top line (MR Accuracy) indicates the accuracy of the MR model on the SR'18 test data which is computed by comparing its output with gold word forms. As the table shows, the accuracy is very high overall ranging from 87.6 to 99.07, with 9 of the 10 languages having an accuracy above 90. This confirms the high accuracy of the model when performing morphological inflection out of context.

The third line (WO+MR (S^{-c})) shows the BLEU scores for our WO+MR model, i.e., when the MR model is applied to the output of the WO model. Here we use an oracle setting which ignores contractions. That is, we compare the WO+MR output not with the final sentence but with the sentence before contraction applies (the ability to handle contractions is investigated in the next section).

As the table shows, the delta in BLEU scores between the model with (WO+MR) and without (WO) morphological realisation mirrors the accuracy of the morphological realisation model: as the accuracy of the morphological inflection model decreases, the delta increases. For instance, for Arabic, the MR accuracy is among the lowest (91.05) and, correspondingly, the decrease in BLEU score when going from word ordering to word ordering with morphological realisation is the largest (-6.3).

5.3 Contraction Generation

To assess the degree to which contractions are used, we compute BLEU-4 between the gold sequence of word forms from UD treebanks and the reference sentence (Table 4, Line S^{-c}/S^{+c}). As the table shows, this BLEU score is very low for some languages (Arabic, French, Italian, Portuguese) indicating a high level of contractions.

These differences are reflected in the results of our WO+MR model: the higher the level of contractions, the stronger the delta between the BLEU score on the reference sentence without contractions (WO+MR, S^{-c}) and the reference sentence with contractions (WO+MR, S^{+c}).

This shows the limits of out-of-context morphological realisation. While the model is good at producing a word form given its lemma and a set of morpho-syntactic features, the lack of contextual information means that contractions cannot be handled.

Adding a contraction module permits improving results for those languages where contraction is frequent (Table 4, Lines WO+MR+C_{reg}, WO+MR+C_{s2s}). Gains range from +5 points for French to +12 for Portuguese when comparing to WO+MR. We achieved better results with contraction module based on regular expressions (C_{reg}), rather than a neural module (C_{s2s}). In a relatively simple task, such as contraction generation, rule-based methods are more reliable, and, overall, are preferable due to their robustness and easy repair comparing to neural models, which may, for instance, hallucinate incorrect content.

5.4 Global Evaluation

Finally, we compare our approach with the best results obtained by the SR'18 participants and with OSU's results (King and White, 2018) using BLEU-4, DIST and NIST scores. OSU results are treated separately, since some of their scores were published after the shared task had ended. Table 5 shows the results. They are mixed. Our model yields the best results for Czech (BLEU: +1.63), Finnish (BLEU: +0.87), Dutch (BLEU: +9.99) and Russian (BLEU: +2.53). However it underperforms on Arabic (BLEU: -9.8), English

	ar	cs	en	es	fi	fr	it	nl	pt	ru
BLEU										
SR'18	16.2	25.05	55.29	49.47	23.26	**52.03**	**44.46**	32.28	30.82	34.34
OSU	**25.6**	53.2	**66.30**	**65.30**	37.5	38.2	42.1	25.5	**47.6**	57.9
Ours	15.8±0.1	**54.83±0.21**	53.3±0.53	51.03±0.31	**38.37±0.55**	41.8±0.26	40.0±0.66	**42.27±0.55**	46.13±0.29	**60.43±0.15**
DIST										
SR'18	44.37	36.48	79.29	51.73	41.21	55.54	58.61	57.81	60.7	34.56
OSU	46.7	58.1	70.2	61.5	58.7	53.7	59.7	57.8	66.0	59.9
Ours	27.63±0.06	63.53±0.15	62.77±0.15	61.33±0.15	51.83±0.55	55.23±0.72	53.73±0.21	54.13±0.15	57.0±0.4	71.23±0.12
NIST										
SR'18	7.15	10.74	10.86	11.12	9.36	9.85	9.11	8.64	7.55	13.06
OSU	7.15	13.5	12.0	12.7	9.56	8.00	8.70	7.33	9.13	14.2
Ours	6.04±0.02	13.6±0.05	10.95±0.09	11.63±0.02	10.41±0.03	8.93±0.08	8.99±0.01	9.51±0.03	9.44±0.04	13.96±0.03

Table 5: BLEU, DIST and NIST scores on the SR'18 test data (shallow track). SR'18 is the official results of the shared task but do not include OSU scores, since they are given in the line below. We also excluded the ADAPT and NILC scores as they were obtained using data augmentation. OSU is the submission of King and White (2018).

(BLEU: -13), Spanish (BLEU: -14.27), French (BLEU: -10.23), Italian (BLEU: -4.46) and Portuguese (BLEU: -1.47). Based on the evaluation of each of our modules, these results can be explained as follows.

The languages for which our model outperforms the state of the art are languages for which the WO model performs best, the accuracy of the morphological realiser is high and the level of contractions is low. For those languages, improving the accuracy of the word ordering model would further improve results.

For four of the languages where the model underperforms (namely, Arabic, French, Portuguese and Italian), the level of contraction is high. This indicates that improvements can be gained by improving the handling of contractions, e.g., by learning a joint model that would take into account both morphological inflection and contraction.

6 Conclusion

While surface realisation is a key component of NLG applications, most work in this domain has focused on the development of language specific models. By providing multi-lingual training and test set, the SR'18 shared task opens up the possibility to investigate how language specific properties such as word order and morphological variation impact performance.

In this paper, we presented a modular approach to surface realisation and applied it to the ten languages of the SR'18 shallow track.

For word ordering, we proposed a simple approach where the data is delexicalised, the input tree is linearised using depth-first search and the mapping between input tree and output lemma sequence is learned using a factored sequence-to-sequence model. Experimental results show that full delexicalisation markedly improves performance. Linguistically, this confirms the intuition that the mapping between shallow dependency structure and word order can be learned independently of the specific words involved.

We further carried out a detailed evaluation of how our word ordering model performs on the ten languages of the SR'18 shallow track. While differences in annotation consistency, number of dependency relations[10] and frequency counts for each dependency relations in each dataset make it difficult to conclude anything from the differences in overall scores between languages, the evaluation of head/dependent word ordering constraints highlighted the fact that long-distance relations, such as ACL, and irregular word ordering constraints (e.g., the position of the verb in Dutch main and subordinate clauses) negatively impact results.

For morphological realisation and contractions, we showed that applying morphological realisation out of context, as is done by most of the SR'18 participating systems[11], yields poor results for those languages (Portuguese, French, Arabic, Italian) where contractions are frequent. We explored two ways of handling contractions (a neural sequence-to-sequence model and a rule-based model) and showed that adding contraction han-

[10]The number of distinct dependency relations present in the treebank ranges between 29 (ar, es) and 44 (en).

[11]The only exception is Castro Ferreira et al. (2018) who train an SMT model on pairs of lemmatised/non-lemmatised sentences.

dling strongly improves performance (from +5.57 to 12.13 increase in BLEU score for the rule-based model depending on the language). More generally, our work on contractions points to the need for SR models to better take into account the fine-grained structure of words. For instance, in French, the article is elided (*le* → *l'*) when the following word starts with a vowel. In future work, we plan to explore the development of a joint model that simultaneously handles morphological realisation and word ordering while using finer grained word representations, such as fastText embeddings (Bojanowski et al., 2017) or byte pair encoding (BPE; Gage, 1994; Sennrich et al., 2016).

References

Roee Aharoni and Yoav Goldberg. 2017. Morphological inflection generation with hard monotonic attention. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2004–2015, Vancouver, Canada. Association for Computational Linguistics.

Andrei Alexandrescu and Katrin Kirchhoff. 2006. Factored neural language models. In *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, pages 1–4. Association for Computational Linguistics.

R Harald Baayen, Richard Piepenbrock, and Leon Gulikers. 1993. The CELEX lexical database (CD-ROM).

Valerio Basile and Alessandro Mazzei. 2018. The dipinfo-unito system for srst 2018. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 65–71. Association for Computational Linguistics.

Anja Belz, Mike 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*, pages 217–226. Association for Computational Linguistics.

Bernd Bohnet, Leo Wanner, Simon Mille, and Alicia Burga. 2010. Broad coverage multilingual deep sentence generation with a stochastic multi-level realizer. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 98–106. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Thiago Castro Ferreira, Sander Wubben, and Emiel Krahmer. 2018. Surface realization shared task 2018 (sr18): The tilburg university approach. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 35–38. Association for Computational Linguistics.

Mingje Chen, Gerasimos Lampouras, and Andreas Vlachos. 2018. Sheffield at e2e: structured prediction approaches to end-to-end language generation. Technical report, E2E Challenge System Descriptions.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016. The SIGMORPHON 2016 shared Task—Morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 10–22, Berlin, Germany. Association for Computational Linguistics.

Markus Dreyer, Jason Smith, and Jason Eisner. 2008. Latent-variable modeling of string transductions with finite-state methods. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 1080–1089, Honolulu, Hawaii. Association for Computational Linguistics.

Greg Durrett and John DeNero. 2013. Supervised learning of complete morphological paradigms. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1185–1195, Atlanta, Georgia. Association for Computational Linguistics.

Ondřej Dušek and Filip Jurcicek. 2015. Training a natural language generator from unaligned data. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 451–461. Association for Computational Linguistics.

Henry Elder and Chris Hokamp. 2018. Generating high-quality surface realizations using data augmentation and factored sequence models. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 49–53. Association for Computational Linguistics.

Philip Gage. 1994. A new algorithm for data compression. *C Users J.*, 12(2):23–38.

David King and Michael White. 2018. The osu realizer for srst '18: Neural sequence-to-sequence inflection and incremental locality-based linearization. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 39–48. Association for Computational Linguistics.

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.

Kexin Liao, Logan Lebanoff, and Fei Liu. 2018. Abstract meaning representation for multi-document summarization. *arXiv preprint arXiv:1806.05655*.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421. Association for Computational Linguistics.

Andreas Madsack, Johanna Heininger, Nyamsuren Davaasambuu, Vitaliia Voronik, Michael Käufl, and Robert Weißgraeber. 2018. Ax semantics' submission to the surface realization shared task 2018. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 54–57. Association for Computational Linguistics.

Diego Marcheggiani and Laura Perez-Beltrachini. 2018. Deep graph convolutional encoders for structured data to text generation. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 1–9, Tilburg University, The Netherlands. Association for Computational Linguistics.

André Martins, Noah Smith, and Eric Xing. 2009. Concise integer linear programming formulations for dependency parsing. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 342–350, Suntec, Singapore. Association for Computational Linguistics.

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 First Workshop on Multilingual Surface Realisation*, pages 1–12. Association for Computational Linguistics.

Joakim Nivre, Željko Agić, Lars Ahrenberg, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros, John Bauer, Kepa Bengoetxea, Riyaz Ahmad Bhat, Eckhard Bick, Cristina Bosco, Gosse Bouma, Sam Bowman, Marie Candito, Gülşen Cebiroğlu Eryiğit, Giuseppe G. A. Celano, Fabricio Chalub, Jinho Choi, Çağrı Çöltekin, Miriam Connor, Elizabeth Davidson, Marie-Catherine de Marneffe, Valeria de Paiva, Arantza Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Kira Droganova, Puneet Dwivedi, Marhaba Eli, Tomaž Erjavec, Richárd Farkas, Jennifer Foster, Cláudia Freitas, Katarína Gajdošová, Daniel Galbraith, Marcos Garcia, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Memduh Gökırmak, Yoav Goldberg, Xavier Gómez Guinovart, Berta Gonzáles Saavedra, Matias Grioni, Normunds Grūzītis, Bruno Guillaume, Nizar Habash, Jan Hajič, Linh Hà Mỹ, Dag Haug, Barbora Hladká, Petter Hohle, Radu Ion, Elena Irimia, Anders Johannsen, Fredrik Jørgensen, Hüner Kaşıkara, Hiroshi Kanayama, Jenna Kanerva, Natalia Kotsyba, Simon Krek, Veronika Laippala, Phng Lê H`ông, Alessandro Lenci, Nikola Ljubešić, Olga Lyashevskaya, Teresa Lynn, Aibek Makazhanov, Christopher Manning, Cătălina Mărănduc, David Mareček, Héctor Martínez Alonso, André Martins, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Amir More, Shunsuke Mori, Bohdan Moskalevskyi, Kadri Muischnek, Nina Mustafina, Kaili Müürisep, Lng Nguy˜ên Thị, Huy`ên Nguy˜ên Thị Minh, Vitaly Nikolaev, Hanna Nurmi, Stina Ojala, Petya Osenova, Lilja Øvrelid, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Guy Perrier, Slav Petrov, Jussi Piitulainen, Barbara Plank, Martin Popel, Lauma Pretkalniņa, Prokopis Prokopidis, Tiina Puolakainen, Sampo Pyysalo, Alexandre Rademaker, Loganathan Ramasamy, Livy Real, Laura Rituma, Rudolf Rosa, Shadi Saleh, Manuela Sanguinetti, Baiba Saulīte, Sebastian Schuster, Djamé Seddah, Wolfgang Seeker, Mojgan Seraji, Lena Shakurova, Mo Shen, Dmitry Sichinava, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Mária Šimková, Kiril Simov, Aaron Smith, Alane Suhr, Umut Sulubacak, Zsolt Szántó, Dima Taji, Takaaki Tanaka, Reut Tsarfaty, Francis Tyers, Sumire Uematsu, Larraitz Uria, Gertjan van Noord, Viktor Varga, Veronika Vincze, Jonathan North Washington, Zdeněk Žabokrtský, Amir Zeldes, Daniel Zeman, and Hanzhi Zhu. 2017. Universal dependencies 2.0. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

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 of the Association for Computational Linguistics*.

Ratish Puduppully, Yue Zhang, and Manish Shrivastava. 2017. Transition-based deep input linearization. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 643–654, Valencia, Spain. Association for Computational Linguistics.

Yevgeniy Puzikov and Iryna Gurevych. 2018. Binlin: A simple method of dependency tree linearization. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 13–28. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pages 1073–1083. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Shreyansh Singh, Ayush Sharma, Avi Chawla, and A.K. Singh. 2018. Iit (bhu) varanasi at msr-srst 2018: A language model based approach for natural language generation. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 29–34. Association for Computational Linguistics.

Marco Antonio Sobrevilla Cabezudo and Thiago Pardo. 2018. Nilc-swornemo at the surface realization shared task: Exploring syntax-based word ordering using neural models. In *Proceedings of the First Workshop on Multilingual Surface Realisation*, pages 58–64. Association for Computational Linguistics.

Linfeng Song, Yue Zhang, Kai Song, and Qun Liu. 2014. Joint morphological generation and syntactic linearization. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*.

Sho Takase, Jun Suzuki, Naoaki Okazaki, Tsutomu Hirao, and Masaaki Nagata. 2016. Neural headline generation on abstract meaning representation. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 1054–1059.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1721. Association for Computational Linguistics.