Neural Morphological Analysis for Turkish with Transformers

Karahan Şahin (Boğaziçi University)
karahan.sahin@boun.edu.tr

Morphological analysis is the NLP subtask where the morpho-syntactic features of a linguistic item are extracted from a raw string. Although it is considered as preprocessing step, it is a vital operation in NLP for agglutinative languages. Different NLP subtasks are covered in the morphological analysis, such as lemmatization, part-of-speech classification, and proper noun detection. This information can be extracted from overt and non-overt features parsed in the morphological analysis.

Related Works. Traditional approaches often utilize finite-state transducers, which cover various rules defined for the morpho-phonotactics of Turkish. This method provides a lower computational complexity and comprehensive coverage of lemma information. The process outputs all possible parses for a task using a sentence. In a further step of the morphological analysis, morphological disambiguation is carried out for classification of most likely parse for the given context.

In later approaches, neural models (mainly RNN based encoder-decoder architecture) carried out the task with the sequence translation approach, which takes raw character embeddings and is translated into morphological analysis output (Shen et al. 2016). The models learn the relations of characters and the mapping of the corresponding features. However, these models only process raw tokens, limited to the training dataset, which is problematic for unseen lemmas. In addition, the disambiguation is carried at the end of the encoder state via the encoding of the contextual representation extracted from a pre-trained language model onto the encoded state of the token.

Dataset. We prepared our dataset using dependency parsing in a dataset annotated by the Google Research team (Kayadelen et al., 2020). We chose this dataset since the parsing is carried out at the morphological level. We have not seen this type of parsing in other Universal Dependencies datasets provided in Turkish. In the processing step, we concatenated lemma, part-of-speech tags, and morphological features along with the derivational boundaries with corresponding changes in part-of-speech.

Experiments. Our approach utilizes the best of both methods. While we use a state-of-the-art sequence model based on the original Transformer architecture (Vaswani, 2017), we also utilize the two-level morphological analysis, which provides intermediate representations of the tokens. The transformer model is selected for the task since the architecture includes a self-attention mechanism not included in the previous architectures. This is especially crucial for the morphological analysis since the cross-morphemic features such as the agreement features and part-of-speech category can be learned in this architecture. Another aspect of the model is that it is robust to the sparsity of the training data. Although the model provides significant baseline results as approximately 0.78 precision and recall scores, it is seen that the model does not learn the boundary between the stem and inflectional morphology for unseen tokens, which causes an overall interference for the model. The second innovation in our model is to process tokens into intermediate representations as in the two-level description. The two-level description is carried optimization of the finite-state transducer. However, in our model, the description is taken for the abstraction of morphemes at the phonological level. Therefore, rather than using the two-level description provided for the Turkish, we utilized the phonological representations described in (Göksel 2004). In the normalization process, we used regular expressions concerning morphological features. Yet defining the phonological features, the expressions do not implement for any disambiguation purposes. After the normalization, we observed a 0.08 - 0.1 increase in precision and recall. In the normalization step, we experimented with the feature normalization process for the morpho-syntactic rules, such as reducing case feature for the bare case at the nominal predicate third-person singular agreement at the present tense inflected predicate. This is carried out since we argued the linguistic representation of empty features is theory-dependent.

In the final step, we have implemented the encoding of contextual representations, which are the word representation of tokens extracted from the BertTürk model. The process is carried out by concatenating contextual word representations with each character embedding obtained from the encoding layer and forwarding the received vector into a linear layer where the embeddings are reduced to the same size as the initial encoder embeddings. We have seen 0.87 precision and recall after the implementation of cont. This architecture is novel since the contextual encoding is carried into each encoded state rather than the final hidden-state extracted from encoder output.

Results. In the experimental results, standard metric provided in the literature is accuracy. However, our study provides a various metrics to evaluate a model of morphological analysis. Rather than accuracy, we first implemented the precision and recall metrics used in the parsing tasks as the $Precision = \frac{1}{n_{test}} \sum_{n_{test}} \frac{S_{gold} \cap S_{pred}}{S_{gold}}$ and $Recall = \frac{1}{n_{test}} \sum_{n_{test}} \frac{S_{gold} \cap S_{pred}}{S_{pred}}$. Also we obtained the F-measure for the Part-of-Speech tags and the exact lemma match. The studies in the literature provide accuracy for all the given features below. However, part-of-speech tag accuracy is calculated for evaluating the performance on morphological disambiguation and lemma accuracy is measured for evaluating the performance on unseen lemma.

	Precision	Recall	POS	Lemma
Baseline	0.783	0.784	0.855	0.840
Feature Reduction	0.784	0.785	0.833	0.852
Phonological Normalization	0.839	0.838	0.886	0.940
Normalization+Reduction	0.855	0.858	0.838	**0.943**
Contextual Embeddings	**0.878**	**0.872**	**0.926**	0.905

Table 1: Experiment Results