Transformers on Multilingual Clause-Level Morphology

Emre Can Acikgoz
KUIS AI, Koç University
eacikgoz17@ku.edu.tr

Tilek Chubakov
KUIS AI, Koç University
tchubakov@ku.edu.tr

Müge Kural
KUIS AI, Koç University
mugekural@ku.edu.tr

Gözde Gül Şahin
KUIS AI, Koç University
gosahin@ku.edu.tr

Deniz Yuret
KUIS AI, Koç University
dyuret@ku.edu.tr

Abstract

This paper describes our winning systems in MRL: The 1st Shared Task on Multilingual Clause-level Morphology (EMNLP 2022 Workshop) designed by KUIS AI NLP team. We present our work for all three parts of the shared task: inflection, reinflection, and analysis. We mainly explore transformers with two approaches: (i) training models from scratch in combination with data augmentation, and (ii) transfer learning with prefix-tuning at multilingual morphological tasks. Data augmentation significantly improves performance for most languages in the inflection and reinflection tasks. On the other hand, Prefix-tuning on a pre-trained mGPT model helps us to adapt analysis tasks in low-data and multilingual settings. While transformer architectures with data augmentation achieved the most promising results for inflection and reinflection tasks, prefix-tuning on mGPT received the highest results for the analysis task. Our systems received 1st place in all three tasks in MRL 2022.[1]

1 Introduction

The shared task on multilingual clause-level morphology was designed to provide a benchmark for morphological analysis and generation at the level of clauses for various typologically diverse languages. The shared task is composed of three subtasks: *inflection*, *reinflection* and *analysis*. For the inflection task, participants are required to generate an output clause, given a verbal lemma and a specific set of morphological tags (features) as an input. In the reinflection task the input is an inflected clause, accompanied by its features (tags). Participants need to predict the target word given a new set of tags (features). Finally, the analysis task requires predicting the underlying lemma and tags (features) given the clauses.

Task1: Inflection		
Source	Lemma	give

	Features	IND;FUT;NOM(1,SG);
		ACC(3,SG,MASC);DAT(3,SG,FEM)

Target	Clause	I will give him to her
Task2: Reinflection		
Source	Clause	I will give him to her

	Features	IND;FUT;NOM(1,SG);
		ACC(3,SG,MASC);DAT(3,SG,FEM)

	Desired Features	IND;PRS;NOM(1,PL); ACC(2);DAT(3,PL);NEG
Target	Desired Clause	We don't give you to them
Task3: Analysis		
Source	Clause	I will give him to her
Target	Lemma	give

	Features	IND;FUT;NOM(1,SG);
		ACC(3,SG,MASC);DAT(3,SG,FEM)

Table 1: Description of the each three task: inflection, reinflection, analysis. **Task1 (Inflection).** For the given lemma and the features, target is the desired clause.**Task2 (Reinflection).** Input is the clause, its features, and the desired output features. Target is the desired clause that represented by the desired features in the source. **Task3 (Analysis).** For a given clause, output is the corresponding lemma and the morphological features.

Literature has examined morphology mainly at the word level, but morphological processes are not confined to words. Phonetic, syntactic, or semantic relations can be studied at phrase-level to explain these processes. Thus, this shared task examines phrase-level morphology and questions the generalization of the relations between the layers of language among languages with different morphological features. The shared task includes eight languages with different complexity and varying morphological characteristics: English, French, German, Hebrew, Russian, Spanish, Swahili, and Turkish.

In our work, we explored two main approaches: (1) training character-based transformer architectures from scratch with data augmentation, (2) adapting a recent prefix-tuning method for language models at multilingual morphological tasks.

[1] https://github.com/emrecanacikgoz/mrl2022

Proceedings of the The 2nd Workshop on Multi-lingual Representation Learning (MRL), pages 100 - 105
December 8, 2022 ©2022 Association for Computational Linguistics

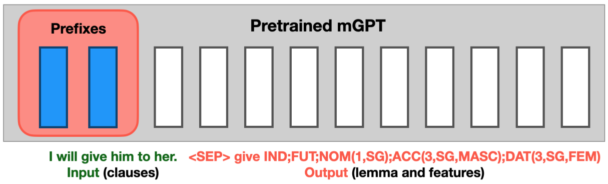
Prefixes
Pretrained mGPT
I will give him to her.
<SEP> give IND;FUT;NOM(1,SG);ACC(3,SG,MASC);DAT(3,SG,FEM)
Input (clauses)
Output (lemma and features)

Figure 1: **Task3 (Analysis)** example by using prefix-tuning method. We freeze all the parameters of the pre-trained mGPT model and only optimize the prefix, which are shown inside the red block. Each vertical block denote transformer activations at one time step.

2 Methods

In this section, first we cover the model architectures and training strategies that we have used (Vaswani et al., 2017; Shliazhko et al., 2022; Li and Liang, 2021), and then discuss our data augmentation strategies in details (Anastasopoulos and Neubig, 2019).

2.1 Vanilla Transformer

We used a modified version of vanilla Transformer architecture in Vaswani et al. (2017) which contains 4 layers of encoder and decoder with 4 multi-head attentions. The embedding size and the feed-forward dimension is set to 256 and 1024, respectively. As suggested in Wu et al. (2021), we used layer normalization before the self-attention and feed-forward layers of the network that leads to slightly better results. We used these in inflection and reinflections tasks.

2.2 Prefix-Tuning

Using prefix-tuning reduces computational costs by optimizing a small continuous task-specific vectors, called prefixes, while keeping frozen all the other parameters of the LLM. We added two prefixes, called virtual tokens in Li and Liang (2021), the gradient optimization made across these prefixes that is described in the Figure 1. We used Shliazhko et al. (2022) weights during prompting. Prefix-tuning method outperforms other fine-tuning approaches in low-data resources and better adapts to unseen topics during prompting (Li and Liang, 2021).

2.3 Data Augmentation

Hallucinating the data for low-resource languages results with a remarkable performance increase for inflection Anastasopoulos and Neubig (2019). The hallucinated data is generated by replacing the stem characters of the aligned word with random characters by using the validation or test sets (see Fig. 2). This way, the amount increase in the training data helps the model to learn and generalize rare seen samples. On the other hand, the amount of hallucinated data that will be added to the training set, hyperparameter N, is also another parameter that directly effects our accuracy. Therefore, hyperparameter N needs to be decided specifically for each language according to corresponding language's complexity and topology.

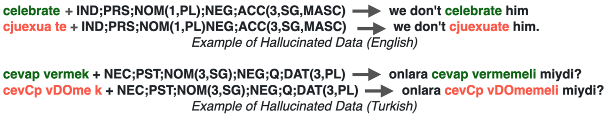
celebrate + IND;PRS;NOM(1,PL);NEG;ACC(3,SG,MASC) → we don't celebrate him
cjuexua te + IND;PRS;NOM(1,PL)NEG;ACC(3,SG,MASC) → we don't cjuexuate him.
Example of Hallucinated Data (English)
cevap vermek + NEC;PST;NOM(3,SG);NEG;Q;DAT(3,PL) → onlara cevap vermemeli miydi?
cevCp vDOme k + NEC;PST;NOM(3,SG);NEG;Q;DAT(3,PL) → onlara cevCp vDOmemeli miydi?
Example of Hallucinated Data (Turkish)

Figure 2: In order to create the hallucinated samples, we first align the characters of the lemma and the inflected forms. After that, we substitute the stem parts of the input with random characters that comes from the validation set and test set, as shown in the figure.

3 Experimental Settings

3.1 Dataset

In the shared task, there are eight different languages with varying linguistic complexity which comes from different language families: English, French, German, Hebrew, Russian, Swahili, Spanish, Turkish. For Hebrew there are two versions as Hebrew-vocalized and Hebrew-unvocalized. Training data contains 10,000 instances for each language and there are 1,000 samples both in development set and test set. Swahili and Spanish are the surprise languages that announced two weeks before the final submission day, together with the unlabeled test data for each language.

3.2 Evaluation

Models are evaluated according to Exact Match (EM), Edit Distance (ED), and F1 accuracy. For task1 (inflection) and task2 (reinflection) ED is the leaderboard metric. For task3 (analysis), F1 score is the objective. EM accuracy represents the ratio of correctly predicted lemma and features, and ED is calculated based on Levenshtein Distance which indicates how different two strings are, (the ground truth and prediction for our case) from each other. F1 accuracy is the harmonic mean of the precision and recall. F1 accuracy is upweighted for the lemma score in our task. In the leaderboard, the results are averaged across each language.

	Task1: Inflection			Task2: Reinflection			Task3: Analysis		
Model	**Transformer + D.A.**			**Transformer**			**Prefix Tuning**		
Metrics	F1↑	EM↑	ED↓	F1↑	EM↑	ED↓	F1↑	EM↑	ED↓
Deu	97.71	91.80	0.241	92.40	66.50	0.788	95.89	83.40	0.991
Eng	98.02	88.90	0.221	95.42	72.30	0.477	99.61	98.50	0.064
Fra	98.59	93.20	0.124	92.64	68.30	0.758	95.63	81.90	0.933
Heb	97.73	89.80	0.550	94.00	83.30	0.796	92.84	73.50	1.322
Heb-Unvoc	97.96	94.20	0.113	86.70	57.70	1.002	82.09	36.20	2.044
Rus	97.57	87.70	0.828	97.29	84.90	0.854	97.51	88.60	3.252
Swa	99.72	99.61	0.019	92.05	84.47	0.182	90.51	62.63	3.114
Spa	98.79	92.00	0.199	96.42	77.60	0.480	98.11	89.40	0.560
Tur	97.50	89.80	0.333	95.36	84.70	0.593	95.36	84.70	0.593
Average	91.89	98.18	**0.292**	93.14	74.72	**0.705**	**94.17**	77.65	1.430

Table 2: Results on the test sets for all tasks and languages with the corresponding models. Edit Distance is the leaderboard ranking metric for Task1: Inflection and Task2: Reinflection, and F1 score is used for leaderboard ranking in Task3: Analysis. D.A. indicates data augmentation.

3.3 Shared Task

Multilingual Clause-level Morphology (MRL 2022) contains three different tasks as Task1: Inflection, Task2: Reinflection, and Task3: Analysis. As KUIS AI team, we have attended each of them separately.

3.3.1 Task1: Inflection

The goal of the task is to produce the output clause and its features forgiven verbal lemma and a set of morphological features, see Table 1. For inflection task, we have trained a vanilla transformer model from scratch by adding some hallucinated data for the training set. The data hallucination method, discussed in 2.3, improved our results significantly. As suggested in Wu et al. (2021), we observed the effect of the large batch sizes that results with an increase in accuracy. Thus, we set the batch size to 400 and we trained our model for 20 epochs. We used Adam optimizer by setting β_1 to 0.9 and β_2 to 0.98. We started with a learning rate of 0.001 with 4,000 warm-up steps. Then, we decrease it with the inverse of the square-root for the remaining steps. We have used label smoothing with a factor of 0.1 and applied the same dropout rate of 0.3.

3.3.2 Task2: Reinflection

In reinflection the task is to generate the desired output format as in inflection; however, the input is consist of an inflected clause, its corresponding features, and a new set of features that represents the desired output form. We again use the same vanilla Transformer architecture, and exactly the same training parameters that we have used in inflection task. We tried both (i) giving the all source data as input, and (ii) using only the inflected clause and its desired features. We have examined that, both our EM and ED accuracy increased in a large manner when we ignore source clause's features in input before feeding it to the model.

3.3.3 Task3: Analysis

Analysis task can be seen as the opposite of the inflection task. For given clauses and its features, we try to generate the lemma and the corresponding morphological features. We used the prefix-tuning method for the analysis task. The prefix template was given as the source and the features were masked. During prompting, we gave the clause-level in input and the target lemma together with its features were expected from the output, like a machine translation task. The source and target are given together with the trainable prefixes, i.e. continuous prompt vectors, and the gradient optimization made across these prefixes. For the mGPT-based Prefix-Tuning model, we have used the *Huggingface*, Wolf et al. (2019) and the corresponding model weights *sberbank-ai/mGPT*. The prefixes were trained for 10 epochs with a batch size of 5 due computational resource constraints. We used Adam optimizer with weight decay fix which is introduced in Loshchilov and Hutter (2017) with β_1=0.9 and β_2=0.999. The learning rate is initialized to 5×10^{-5} and a linear scheduler is used without any warm-up steps.

System	Inflection	Reinflection	Analysis
Transformer Baseline	3.278	4.642	80.00
mT5 Baseline	2.577	2.826	84.50
KUIS AI	**0.292**	**0.705**	**94.17**

Table 3: Submitted results for MRL shared task that is averaged across 9 languages. Metrics for the inflection and reinflection tasks is the edit distance, and for analysis the metric is averaged F1 score with the lemma being treated as an up-weighted feature.

3.4 Results

Our submitted results are provided in Table 2. The announced results by the shared task are in the Table 3 which are evaluated among the provided unlabeled test set.

For the inflection task, with the help of data augmentation, we have achieved best average edit distance for languages. Specially, for Swahili the edit distance is nearly perfect as well as the exact match. It is followed by Hebrew-Unvoc and French. We observed the highest edit distance and the lowest exact match scores for Russian. At the end, we observed that, reducing edit distance does not always bring better exact match.

For the reinflection task, using trained transformer models from scratch, we again see the best results for Swahili with the lowest edit distance. This time, the highest edit distance belongs to Hebrew-Unvoc as well as the lowest exact match. The number of words and characters in the examples of task datasets may be the factors and should also be considered.

Finally for the analysis, with the help of prefix-tuning, we achieved the best results for English with highest F1 score. The ease of finding English pre-trained models led us to experiment with English-only GPT models, and we subsequently discovered that multilingual GPT gives better results when using prefix-tuning. Tuning on mGPT has the lowest performance with Hebrew-Unvoc, due the low ratio of training samples in Hebrew during pre-training compared to other languages.

4 Related Work

Word-level morphological tasks have been studied to a great extent, with LSTM (Wu and Cotterell, 2019; Cotterell et al., 2016; Malaviya et al., 2019; Sahin and Steedman, 2018), GRU (Conforti et al., 2018), variants of Transformer Vaswani et al. (2017); Wu et al. (2021) and other neural models (e.g., invertible neural networks (Sahin and Gurevych, 2020)). Unlike word-level, there is limited work on clause-level morpho-syntactic modeling. Goldman and Tsarfaty (2022) presents a new dataset for clause-level morphology covering 4 typologically-different languages (English, German, Turkish, and Hebrew); motivates redefining the problem at the clause-level to enable the cross-linguistical study of neural morphological modeling; and derives clause-level inflection, reinflection, and analysis tasks together with baseline model results.

Pre-trained LLMs have been successfully applied to downstream tasks like sentiment analysis, question answering, named entity recognition, and part-of-speech (POS) tagging (Devlin et al., 2019; Yang et al., 2019; Raffel et al., 2020). Even though, there is limited work on applications of LLMs to morphological tasks, it has been demonstrated that using pre-trained contextualized word embeddings can significantly improve the performance of models for downstream morphological tasks. Inoue et al. (2022) explored BERT-based classifiers for training morphosyntactic tagging models for Arabic and its dialect. Anastasyev (2020) explored the usage of ELMo and BERT embeddings to improve the performance of joint morpho-syntactic parser for Russian. Hofmann et al. (2020) used a fine-tuning approach to BERT for the derivational morphology generation task. Finally, Seker et al. (2022) presented a large pre-trained language model for Modern Hebrew that shows promising results at several tasks.

On the other hand, since fine-tuning LLMs requires to modify and store all the parameters in a LM that results with a huge computational cost. Rebuffi et al. (2017); Houlsby et al. (2019) used adapter-tuning which adds task-specific layers (adapters) between the each layer of a pre-trained language model and tunes only the 2%-4% parameters of a LM. Similarly, Li and Liang (2021) proposed prefix-tuning which is a light-weight alternative method for adapter-tuning that is inspired by prompting.

5 Conclusion

In this paper, we described our winning methods multilingual clause-level morphology shared task for inflection, reinflection, and analysis. Due to the different complexity between tasks and the varying morphological characteristics of languages, there is no single best model that achieves the best results for each task in each language. Thus, we try to implement different types of systems with different objectives. For inflection we used a vanilla Transformer adapted from Vaswani et al. (2017) and applying data hallucination substantially improves accuracy (Anastasopoulos and Neubig, 2019). The reinflection task is more challenging compared to the other tasks due to its complex input form. To overcome this issue, we have removed the original feature tags from the input. We only used the inflected clause and target features in the input. We again used a vanilla Transformer as a model choice. Finally, for the analysis task, we used the prefix-tuning method based on mGPT. On average, we have achieved the best results for every three tasks among all participants.

Acknowledgements

This work is supported by KUIS AI Center from Koç University, Istanbul. We gratefully acknowledge this support. Last but not least, we would like to kindly thank our organizers for answering our questions and for the effort they have made to fix the issues that we struggled during the competition process.

References

Antonios Anastasopoulos and Graham Neubig. 2019. Pushing the limits of low-resource morphological inflection. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 984–996. Association for Computational Linguistics.

D.G. Anastasyev. 2020. Exploring pretrained models for joint morpho-syntactic parsing of russian. volume 2020-June, page 1 – 12. Cited by: 4; All Open Access, Bronze Open Access.

Costanza Conforti, Matthias Huck, and Alexander M. Fraser. 2018. Neural morphological tagging of lemma sequences for machine translation. In *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas, AMTA 2018, Boston, MA, USA, March 17-21, 2018 - Volume 1: Research Papers*, pages 39–53. Association for Machine Translation in the Americas.

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, Berlin, Germany, August 11, 2016*, pages 10–22. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Omer Goldman and Reut Tsarfaty. 2022. Morphology without borders: Clause-level morphological annotation. *CoRR*, abs/2202.12832.

Valentin Hofmann, Janet B. Pierrehumbert, and Hinrich Schütze. 2020. Dagobert: Generating derivational morphology with a pretrained language model. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 3848–3861. Association for Computational Linguistics.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. *CoRR*, abs/1902.00751.

Go Inoue, Salam Khalifa, and Nizar Habash. 2022. Morphosyntactic tagging with pre-trained language models for arabic and its dialects. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 1708–1719. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 4582–4597. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Chaitanya Malaviya, Shijie Wu, and Ryan Cotterell. 2019. A simple joint model for improved contextual neural lemmatization. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1517–1528. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Sylvestre-Alvise Rebuffi, Hakan Bilen, and Andrea Vedaldi. 2017. Learning multiple visual domains with residual adapters. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 506–516.

Gözde Gül Sahin and Iryna Gurevych. 2020. Two birds with one stone: Investigating invertible neural networks for inverse problems in morphology. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 7814–7821. AAAI Press.

Gözde Gül Sahin and Mark Steedman. 2018. Character-level models versus morphology in semantic role labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 386–396. Association for Computational Linguistics.

Amit Seker, Elron Bandel, Dan Bareket, Idan Brusilovsky, Refael Shaked Greenfeld, and Reut Tsarfaty. 2022. Alephbert: Language model pre-training and evaluation from sub-word to sentence level. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 46–56. Association for Computational Linguistics.

Oleh Shliazhko, Alena Fenogenova, Maria Tikhonova, Vladislav Mikhailov, Anastasia Kozlova, and Tatiana Shavrina. 2022. mgpt: Few-shot learners go multilingual. *CoRR*, abs/2204.07580.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Shijie Wu and Ryan Cotterell. 2019. Exact hard monotonic attention for character-level transduction. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1530–1537. Association for Computational Linguistics.

Shijie Wu, Ryan Cotterell, and Mans Hulden. 2021. Applying the transformer to character-level transduction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 1901–1907. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764.