# OSU at SigMorphon 2022: Analogical Inflection With Rule Features

**Micha Elsner** and **Sara Court**
Department of Linguistics
Ohio State University
melsner0@gmail.com and court.22@osu.edu

## Abstract

OSU's inflection system is a transformer whose input is augmented with an analogical exemplar showing how to inflect a different word into the target cell. In addition, alignment-based heuristic features indicate how well the exemplar is likely to match the output. OSU's scores substantially improve over the baseline transformer for instances where an exemplar is available, though not quite matching the challenge winner. In Part 2, the system shows a tendency to over-apply the majority pattern in English, but not Arabic.

## 1 Introduction

Many theories of inflection production propose a central role for memorized word forms in shaping the outcomes for unknown or weakly represented words (Bybee, 1995). In such memory-based models, speakers retrieve *exemplar* forms $A$ from memory for which the outcomes $B$ are known and use them to predict the outcome for a word $C$ via a process of analogical reasoning: *exemplar source* $A$ : *exemplar target* $B$ :: *source* $C$ : *target* $D$. This type of analogical reasoning is detectable in historical changes (Sims-Williams, 2021) and in experiments with nonce-words (Dąbrowska, 2008), and underlies some influential computational models of inflection (Albright and Hayes, 2003; Daelemans, 2002). Recently, Elsner (2021) and Liu and Hulden (2020) show that transformer models for inflection prediction can also benefit from access to exemplars.

OSU's inflection prediction system[1] builds on this recent work, also using a transformer for prediction, but adds a heuristic set of "rule features" intended to make the system more flexible in its use of analogical reasoning. Rule features are necessary because the source-target pair $C$ : $D$ may not correspond directly to the exemplar pair due to morphophonological alternations or inflection class mismatch. Consider an analogy from Anglo-Saxon, *ēþel : ēþle :: ġelīca : ġelīcan* ("homeland", "equal".DAT.SG), for which the target suffixes do not match. Below is a prediction instance and its desired output, based on previous work:

(1) ġelīca DAT.SG ēþel : ēþle → ġelīcan

When instances like this are common in training, the relative unreliability of the exemplar information leads the system to concentrate on the output cell label DAT.SG and ignore the exemplar, which results in performance very similar to a transformer baseline without exemplars. To prevent this, we augment training examples to indicate whether the desired output matches or mismatches the exemplar; these augmented features are predicted by the transformer at test time (see Section 3). For example, we can add features indicating that the exemplar has a suffix which does not match, so that the system can learn whether to attend to it:

(2) ġelīca DAT.SG ēþel : ēþle SUFF REPLACE.SUFF
→ ġelīcan

In pilot experiments, systems trained with these features behaved qualitatively differently from the baseline, reacting more to exemplar information and producing a wider variety of outputs when the exemplar was varied.

## 2 Results

OSU entered systems for both Part 1 (multilingual inflection; Kodner et al. (2022)) and Part 2 (learning trajectories: Kodner and Khalifa (2022)). However, we did not attempt all parts of the Part 1 task. First, we ran each language from Part 1 with the largest available dataset; we submitted results for the **small** partition only for languages which

[1] https://github.com/melsner/transformerbyexample



*19th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 220 - 225

| | Overall | Both | Cell |
|---|---|---|---|
| Small part. | 47.688 | 79.31 | 82.308 |
| Large part. | 46.734 | 89.565 | 85.308 |
| Large winner | 67.853 | 90.991 | 87.171 |
| Large neural base | 62.391 | 80.462 | 77.627 |

Table 1: Official results for Task 0, Part 1: score overall, score for items with known lemma and cell, score for items with unknown lemma and known cell.

lacked a **large** training set. Second, our system relies on being able to recall an exemplar with a known output for the target cell. Thus, we did not attempt instances for which the target cell was unseen (**lemma-only** and **neither**); for such instances, we output the original lemma as a placeholder prediction.

Our results overall (Table 1) reflect our inability to make predictions on unknown cells. However, for known cells, performance is fairly close to the challenge winner CLUZH, though the differences are statistically significant. Moreover, the system comfortably outperforms the neural baseline. This is particularly interesting since the baseline uses the same transformer model, Wu et al. (2021), for predictions; only the instance generation and training procedure differ. Nonetheless, the system improves by almost 10% absolute when the cell is known.

OSU surpassed the neural baseline in the known cell, unknown lemma condition by 1% absolute or more on Armenian, Karelian, Polish, Slovak, Turkish and Veps (for all these except Armenian, the improvement was at least 10%). It performed worse than baseline on Arabic, Assamese, Hungarian, Korean, Ludic, Old Norse and Pomak (with a 12% drop on Korean)[2]. There is no obvious typological pattern in these results. Two Slavic languages (Polish and Slovak) performed excellently while a third (Pomak) underperformed; similarly, one Finnic language (Karelian) performed well while another (Ludic) did not. While several underperforming languages used non-Latin scripts, which can cause trouble for inflector models (Murikinati et al., 2020), OSU was the best-performing system on Gothic, with some words written in Gothic script and others in Latin characters, and also performed well on Khalkha Mongolian, written in Cyrillic, and on Hebrew.

Task 2 (learning trajectories) did not involve held-out cells, but did vary the amount of training data. A representative set of learning curves for Arabic is shown in Figure 1; curves for English and German are qualitatively similar. Our system experiences a rapid rise in performance between 100 and 300 training items, with diminishing returns around 600 items. We attribute poor early performance to under-regularization; unlike in Part 1, we did not use cross-lingual training, which helps to regularize small-data inflectors (Kann et al., 2017).

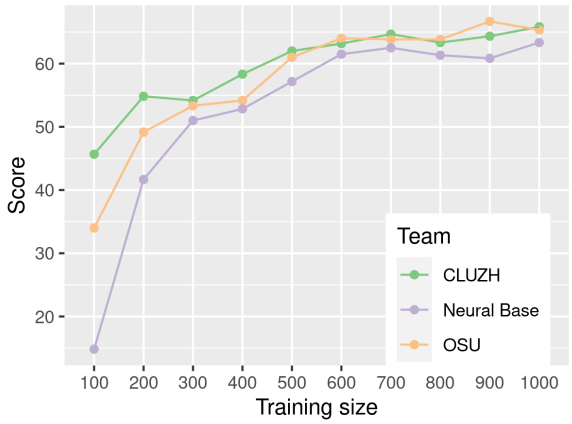


Figure 1: Comparative learning trajectories for Arabic (Task 2).

## 3 System design

Training using the OSU system involves the following steps: (1) generation of training instances, (2) training a language-agnostic string edit model, (3) multilingual training, (4) language-specific fine-tuning. Each training instance includes the input lemma, the morphosyntactic features of the output cell, the language and the language family (each encoded as a character), the exemplar lemma and form (separated by a diacritic), and the rule features. Rule features for training instances are generated by aligning the lemma and output form as in Ahlberg et al. (2015), aligning the exemplar and its output, and then comparing the two. Looking only at the lemma and output, we generate features to indicate whether there is a prefix, a suffix, a stem-internal edit, or no edit. Based on the comparison, we indicate whether prefix/suffix/stem edits are identical between source-target pairs, contain some but not all matching characters, or are disjoint. For instance, the pair *fill* ∼ *filled* (suffix *-ed*) with exemplar *die* ∼ *died* (suffix *-d*) would be marked SUFFIX to indicate the rule type and SIMILAR.SUFFIX to indicate that the edits match partly

[2]Our development score for this condition in Korean is 80.679%; our test score is 50.602%, suggesting there may be a dataset mismatch.

but not completely. The transformer is not forced to obey these features when generating outputs, but uses them to learn how to attend to the exemplar.

For each training item, we generate one instance for inflection prediction (with rule features generated using the gold output) and one for feature prediction, containing the exemplar and cell label, but with the rule features as output. The transformer thus learns to predict a likely alignment configuration for each query/exemplar pair. For example, a feature prediction instance corresponding to Example (2) would be:

(3) ġelīca DAT.SG ēþel : ēþle PREDICT.FEATS → SUFF REPLACE.SUFF

Language-agnostic string edit instances for step (2) (random strings with prefixes, suffixes or internal edits) were generated as in Elsner (2021). In step (3), we trained all languages together for 18 hours (during which we ran 57 epochs). We then trained sub-models by language family, but since many families this year had only one or two representatives, we decreased this training process to only 5 epochs, anticipating that it would make little difference. Finally, we trained for 50 more epochs on the individual language training sets. The learning model itself is a transformer with settings from Wu et al. (2021).

Inference is a multistep process involving the following steps: (1) generation of multiple test instances with different exemplars, (2) prediction of rule features for each instance, (3) prediction of inflected forms for each instance, (4) majority voting to produce a single inflected output. In step (1), we sampled 5 random exemplars from the training set for each test item; the exemplar output was always drawn from exactly the same morphosyntactic cell as the target output.[3] We generated an instance for each test item × exemplar. We used the transformer in feature prediction mode to produce rule features for each instance (step 2), then concatenated these rule features with the inputs to produce inflection instances. By re-running the transformer on these augmented instances, we output an inflected form for each instance (step 4). Finally, we chose the most likely output across the 5 exemplars as the model's final prediction, with ties broken at random.

[3]As stated, if a suitable exemplar cannot be found, we produce the input form as a placeholder prediction.

As an example of this process, suppose the instance *ġelīca* DAT.SG occurred in the test set, and we had selected the pair *ēþel : ēþle* as one of our five exemplars. We would first generate a feature prediction instance (example 3) and present it to the trained transformer. Suppose the transformer incorrectly assumed the suffix would be shared, and output SUFF SAME.SUFF (rather than REPLACE.SUFF). In step (3), we create an inflection instance using these predicted features:

(4) ġelīca DAT.SG ēþel : ēþle SUFF REPLACE.SUFF → ġelīcan

As with any pipelined prediction system, an error cascade may occur; the transformer may not decode this instance correctly due to the incorrect features proposed in the previous step. In any case, we would collect this output, and those of the four other exemplars, and select the most frequently proposed form as the final prediction.

System development was carried out before the shared task commenced, using datasets from SIGMORPHON 2020 (Vylomova et al., 2020); we made no effort to tune on the 2022 datasets.

## 4 Analysis

We analyze some outputs from Part 2 with an alignment-based analysis tool as in King et al. (2020); Gorman et al. (2019), leveraging some of the same code as our rule feature extractor. In English, the model shows a strong preference for over-applying the regular (*-ed*) suffix throughout the learning process; using the 100-example (severely under-regularized) dataset, the model produces suffixes 84% of the time, but by 200 examples, this rises to 90% and continues to rise slowly thereafter. Nearly all of the rise in accuracy is due to the model's gradual acquisition of orthographic allomorphs of *-ed*, such as *drum* ∼ *drummed*, first produced with 400 examples. No irregular allomorphs improve consistently, although some (*swear* ∼ *swore*, *grind* ∼ *ground*) are occasionally produced correctly. The zero past tense (*bet* ∼ *bet*) is produced less often as the dataset increases. In other experiments, we have observed that our model often produces zero outputs when trained with insufficient data; we believe our initial success with this class is the product of this tendency rather than learning.

The lack of generalization of irregular allomorphs is generally consistent with the claim of Xu

and Pinker (1995) that infants rarely produce such errors. It is not clear from results on held-out data whether a "U-shaped curve" (Marcus et al., 1992) would appear, since this phenomenon results from over-application of the regular suffix to previously *memorized* irregulars and would require inspection of training outputs. It is also likely that the token, as well as type, frequency distribution of the training data matters for the acquisition of irregulars (Frank et al., 2020).

In Arabic, the model is able to learn suffixing 'sound' plurals starting from the first 100 words, and performs best on these examples overall, reaching over 80% accuracy on concatenative patterns when trained on all available data. The model initially struggles with nonconcatenative 'broken' plural forms, but shows consistent improvement as the amount of training data increases. The alignment method used to generate training instances groups alternations into microclasses, taking changes in short vowel diacritics into account. One of these classes, the CaCCaC ∼ CaCaaCiC class, containing nouns such as *maslak* ∼ *masaalik* 'path' (35 examples), reaches 100% accuracy with 600 words. Gradual improvement is also seen in nouns of the CaCaC ∼ 'aCCaaC class, for example *khtar* ∼ *akhtaar* 'danger' (50 examples), which goes from 2% accuracy using 100 words to 86% accuracy on the full dataset. Another interesting class is the CiCaa' ∼ 'aCCiya class, for example *binaa'* ∼ *'ibniya* 'building', with only 5 examples in the dataset. Unlike other microclasses of similar size which the model fails to ever learn, the model is able to accurately produce 4 of the 5 examples (80% accuracy) using the 600-word and 900-word datasets (although with 1,000 words the model only produces 1 of the 5). Other similar nouns, such as the CaCiiC ∼ 'aCCiCaa' pattern including *qariib* ∼ *aqribaa'* 'relative' (5 examples) are never learned by the model.

The model's performance reflects broad generalizations found in the literature on child acquisition of dialectal Arabic plural inflection. In general, while nonconcatenative 'broken' plural nouns are present in the speech of very young children, nonconcatenative inflection isn't productive until late preschool (Ravid and Farah, 1999), and a study on the acquisition of plural inflection in Egyptian Arabic found that children as old as 15 may commonly produce errors when inflecting broken plural nouns in the language (Omar, 2017). In their study on plural acquisition of native Arabic speakers across multiple age groups, Saiegh-Haddad et al. (2012) found that the feminine sound plural marker is acquired earlier and faster than broken plural inflection patterns, and that differences in the production of broken plural forms are affected both by speakers' familiarity with the singular form and the type frequency of its associated plural template. Both the human and machine acquisition trajectories are likely related to the sheer number of possible ways (i.e., 'templates') of nonconcatenatively relating singular and plural nouns in Semitic languages. There are comparatively far fewer productive suffixes in MSA (one feminine and one masculine) than there are templates (perhaps more than 70: Plunkett and Nakisa (1997)).

## 5 Conclusion and Future work

The competition alerts us to one obvious weak point: our inability to predict fillers for cells in which no training example is given. This is particularly problematic for languages with very large paradigms. Such paradigms generally involve some degree of agglutination (separatist exponence) which renders low-frequency cells predictable (Plank, 2017). The relationships between cells can be modeled by using multiple input forms to predict a target (Rathi et al., 2021). The ability to do this would be a valuable addition to our model.

While our system was not the best in the competition, we are encouraged to find that analogical examples allow a transformer inflector to achieve near-state-of-the-art results. An analogical model is both cognitively plausible and easy to implement, and the resulting system is substantially more robust and generalizable than the simple transformer baseline.

## Acknowledgements

We thank Jordan Kodner, Salam Khalifa and all the SM'22 shared task organizers, and Andrea Sims for design discussions. All experiments were run on the Ohio Supercomputer (OSC, 1987).

## References

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2015. Paradigm classification in supervised learning of morphology. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1024–1029, Denver,

Colorado. Association for Computational Linguistics.

Adam Albright and Bruce Hayes. 2003. Rules vs. analogy in english past tenses: A computational/experimental study. *Cognition*, 90(2):119–161.

Joan Bybee. 1995. Diachronic and typological properties of morphology and their implications for representation. In Laurie Beth Feldman, editor, *Morphological aspects of language processing*, pages 225–246. Erlbaum Hillsdale.

Ewa Dąbrowska. 2008. The effects of frequency and neighbourhood density on adult speakers' productivity with polish case inflections: An empirical test of usage-based approaches to morphology. *Journal of Memory and Language*, 58(4):931–951.

Walter Daelemans. 2002. A comparison of analogical modeling to memory-based language processing. In *Analogical modeling : an exemplar-based approach to language*, pages 157–179.

Micha Elsner. 2021. What transfers in morphological inflection? experiments with analogical models. In *Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 154–166, Online. Association for Computational Linguistics.

Stella Frank, Kenny Smith, and Christine F. Cuskley. 2020. Learner dynamics in a model of wug inflection: integrating frequency and phonology. In *CogSci*.

Kyle Gorman, Arya D McCarthy, Ryan Cotterell, Ekaterina Vylomova, Miikka Silfverberg, and Magdalena Markowska. 2019. Weird inflects but ok: Making sense of morphological generation errors. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 140–151.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017. One-shot neural cross-lingual transfer for paradigm completion. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1993–2003, Vancouver, Canada. Association for Computational Linguistics.

David King, Andrea D. Sims, and Micha Elsner. 2020. Interpreting sequence-to-sequence models for Russian inflectional morphology. In *Proceedings of the Society for Computation in Linguistics (SCiL) 3*, pages Article 39, 402–411. Society for Computation in Linguistics.

Jordan Kodner and Salam Khalifa. 2022. SIGMORPHON-UniMorph 2022 Shared Task 0: Modeling Inflection in Language Acquisition. In *Proceedings of the SIGMORPHON 2022 Shared Task: Morphological Inflection*, Seattle. Association for Computational Linguistics.

Jordan Kodner, Salam Khalifa, Khuyagbaatar Batsuren, Hossep Dolatian, Faruk Akkuş, Antonios Anastasopoulos, Taras Andrushko, Aryaman Arora, Nona Atanelov, Elena Budianskaya, Bella Gábor, Yustinus Ghanggo Ate, Omer Goldman, Simon Guriel, Silvia Guriel-Agiashvili, Ritvan Karahodja, Andrew Krizhanovsky, Natalia Krizhanovsky, Igor Marchenko, Magdalena Markowska, Polina Mashkovtseva, Maria Nepomniashchaya, Daria Rodionova, Elizabeth Salesky, Alexandra Serova, Anastasia Yemelina, Jeremiah Young, and Ekaterina Vylomova. 2022. SIGMORPHON-UniMorph 2022 Shared Task 0: Generalization and Typologically Diverse Morphological Inflection. In *Proceedings of the SIGMORPHON 2022 Shared Task: Morphological Inflection*, Seattle. Association for Computational Linguistics.

Ling Liu and Mans Hulden. 2020. Analogy models for neural word inflection. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2861–2878, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Gary F Marcus, Steven Pinker, Michael Ullman, Michelle Hollander, T John Rosen, Fei Xu, and Harald Clahsen. 1992. Overregularization in language acquisition. *Monographs of the society for research in child development*, pages i–178.

Nikitha Murikinati, Antonios Anastasopoulos, and Graham Neubig. 2020. Transliteration for cross-lingual morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 189–197, Online. Association for Computational Linguistics.

Margaret K. Omar. 2017. *The Acquisition of Egyptian Arabic as a Native Language*. De Gruyter Mouton.

OSC. 1987. Ohio supercomputer center.

Frans Plank. 2017. Split morphology: How agglutination and flexion mix. *Linguistic Typology*, 21(2017).

Kim Plunkett and Ramin Charles Nakisa. 1997. A connectionist model of the arabic plural system. *Language and Cognitive processes*, 12(5-6):807–836.

Neil Rathi, Michael Hahn, and Richard Futrell. 2021. An information-theoretic characterization of morphological fusion. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10115–10120, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Dorit Ravid and Rola Farah. 1999. Learning about noun plurals in early palestinian arabic. *First Language*, 19(56):187–206.

Elinor Saiegh-Haddad, Areen Hadieh, and Dorit Ravid. 2012. Acquiring noun plurals in palestinian arabic: Morphology, familiarity, and pattern frequency. *Language learning*, 62(4):1079–1109.

Helen Sims-Williams. 2021. Token frequency as a determinant of morphological change. *Journal of Linguistics*, online first.

Ekaterina Vylomova, Jennifer White, Elizabeth Salesky, Sabrina J. Mielke, Shijie Wu, Edoardo Maria Ponti, Rowan Hall Maudslay, Ran Zmigrod, Josef Valvoda, Svetlana Toldova, Francis Tyers, Elena Klyachko, Ilya Yegorov, Natalia Krizhanovsky, Paula Czarnowska, Irene Nikkarinen, Andrew Krizhanovsky, Tiago Pimentel, Lucas Torroba Hennigen, Christo Kirov, Garrett Nicolai, Adina Williams, Antonios Anastasopoulos, Hilaria Cruz, Eleanor Chodroff, Ryan Cotterell, Miikka Silfverberg, and Mans Hulden. 2020. SIGMORPHON 2020 shared task 0: Typologically diverse morphological inflection. In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 1–39, Online. 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*, pages 1901–1907, Online. Association for Computational Linguistics.

Fei Xu and Steven Pinker. 1995. Weird past tense forms. *Journal of child language*, 22(3):531–556.