# The Automatic Characterization of Grammars from Small Wordlists

**Jordan Kodner**[1], **Spencer Caplan**[1], **Hongzhi Xu**[2], **Mitchell P. Marcus**[2], **Charles Yang**[1]
**University of Pennsylvania** [1] Department of Linguistics, [2] Department of Computer and Information Science
{jkodner, spcaplan}@sas.upenn.edu, {xh, mitch}@cis.upenn.edu, charles.yang@ling.upenn.edu

## Vowel Harmony Overview

**An unsupervised algorithm for detecting and describing vowel harmony systems in small wordlists.** It answers the following questions about an unknown language:

- Does the language have harmony?
- What are its harmonizing sets?
- Does it have neutral (transparent or opaque) vowels?
- Does it have secondary harmony?

## Vowel Harmony Algorithm

The algorithm is designed to work on short wordlists (**down to about 500 types**) without frequency counts. If the standard orthography roughly approximates a phonemic representation, **no transcription is needed**. If available, token frequencies may be used to improve results. Furthermore, the algorithm can provide **a mapping between harmonizing sets** if the researcher provides vowel features as input.

**if** frequencies provided **then**
  Trim tail off *wordlist*
**while** $True$ **do**
  Calculate tier-adjacent V-V co-occurrence matrix
  Calculate MI between each vowel pair
  Identify vowels whose MI distributions uniform within threshold.
  Assign these to the neutral vowel set and remove from consideration
  **if** number of non-neutral vowels $\leq 1$ **then**
    **return**
  Run k-means ($k = 2$) clustering on the remaining vowels' MI vectors
  **if** no *features* provided **then**
    **return**
  **else**
    Map vowels between harmonizing sets by finding pairs that share the most features in common.
    *vowel list* $\leftarrow$ Collapse vowels along the harmonizing feature
    rerun for secondary harmony
**return**

## Results

| Lang. | # Types | 1ary H? | Correct | 2ary H? | Correct |
|---|---|---|---|---|---|
| **Turkish** | 303,013 | ✓ | 8/8 | ✓ | 4/4 |
| **Finnish** | 396,770 | ✓ | 8/8 | – | – |
| **Hungar.** | 53,839 | ✓ | 11/15 | – | – |
| **Uyghur** | 392,403 | ✓ | 7/8 | – | – |
| **Warlpiri** | 28,885 | ✓ | 3/3 | – | – |
| **German** | 225,327 | – | 5/5 | – | – |
| **English** | 101,438 | – | 6/6 | – | – |

Table 1: Vowel co-occurrences are taken from corpus orthographies. Marginal vowels (e.g. Finnish *å* and German *y*) are automatically detected and removed. Corpora are from MorphoChallenge Kurimo et al. (2010) when available. Uyghur and Hungarian were provided for the DARPA LORELEI project. Warlpiri is from Swartz (1997).

## Next Steps

We are continuing to develop this algorithm.

- **Leveraging paradigms** from our morphological segmentation will allow it to map harmonizing vowels with explicitly provided features.
- The same distributional processes can discover **other typological features:** whether a language exhibits stem alternations, has agglutinative morphology, tends towards prefixation or suffixation, reduplication, etc.

## References

Kurimo, M., Virpioja, S., Turunen, V., and Lagus, K. (2010). Morpho challenge competition 2005–2010: evaluations and results. In *Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology*, pages 87–95. Association for Computational Linguistics.

Narasimhan, K., Barzilay, R., and Jaakkola, T. (2015). An unsupervised method for uncovering morphological chains. *arXiv preprint arXiv:1503.02335.*

Swartz, S. (1997). *Warlpiri yimi kuja karlipa wangka.* Summer Institute of Linguistics, Australian Aborigines and Islanders Branch, Warlpiri Translation Project.

## Segmentation Overview

**An unsupervised morphological segmentation algorithm designed with small wordlists in mind.** Our algorithm is built around the concept of **paradigms**. Each root is attached to a paradigm containing all the proposed suffixes with which it is attested.
This algorithm achieves **state-of-the-art results** on English and Turkish. We are preparing gold standards for testing on other languages as well.

## Segmentation Algorithm Summary

The morphological segmentation algorithm combines three processes: **segmentation, paradigm construction, and pruning.**

- **Segmentation** - A Bayesian model estimates probability $P(r, s, t|w)$ over candidate roots, affixes, and transformations for each word

$$P(r, s, t|w) = \frac{P(r) \times P(s) \times P(t|f(r, s))}{\sum_{(r',s',t') \in w} P(r', s', t')}$$

- **Paradigm Construction** - Affix appearing with each root are grouped together into paradigms. The more common its paradigm, the greater its *support*.

| Paradigm | Support |
|---|---|
| (-ed, -ing, -s) | 772 |
| (-ed, -ing) | 331 |
| (-ed, -er, -ing, -s) | 219 |
| (-ly, -ness) | 208 |
| (-ed, -ing, -ion, -s) | 154 |

- **Pruning** - Affixes which do not appear in enough well-supported paradigms are pruned. For example, if *closet* is incorrectly segmented as `close-t`, the *close* paradigm becomes {*-er*, *-est*, *-ed*, *-ing*, *-s*, ***-t***}. Pruning corrects the *-t*.

## Results

| Lang. | Model | Prec. | Recall | F1 |
|---|---|---|---|---|
| **English** | Morfessor-Base | 0.740 | 0.623 | 0.677 |
| | AGMorph | 0.696 | 0.604 | 0.647 |
| | MorphChain-C | 0.555 | 0.792 | 0.653 |
| | MorphChain-All | 0.807 | 0.722 | 0.762 |
| | **Our model** | 0.804 | 0.764 | **0.784** |
| **Turkish** | Morfessor-Base | 0.827 | 0.362 | 0.504 |
| | AGMorph | 0.878 | 0.466 | 0.609 |
| | MorphChain-C | 0.516 | 0.652 | 0.576 |
| | MorphChain-All | 0.743 | 0.520 | 0.612 |
| | **Our model** | 0.589 | 0.726 | **0.650** |

Table 2: All numbers except for ours are reported in Narasimhan et al. (2015). Best results are reported.

## Next Steps

We are still developing this tool. We expect improvements to come from integration with the vowel harmony analyzer as well as more theoretically involved morphological transformations.

- Have run it on other languages: Tagalog, Navajo, Yoruba, Somali; but cannot score the outputs yet
- Are designing a **segmentation annotation scheme** to create more gold standards
- Will **leverage the vowel harmony tool** to create more coherent paradigms
- Will **enrich transformations** for languages with non-concatenative morphology

## Acknowledgements

We thank the rest of the University of Pennsylvania's LORELEI summer research team, audiences at the 10th Northeast Computational Phonology Circle (NECPhon), Penn's Computational Linguistics and Lunch, and our anonymous ComputEL-2 reviewers. *This research was funded by the DARPA LORELEI program under Agreement No. HR0011-15-2-0023.*

DARPA Penn University of Pennsylvania