UNIVERSITY OF COPENHAGEN
DEPARTMENT OF COMPUTER SCIENCE

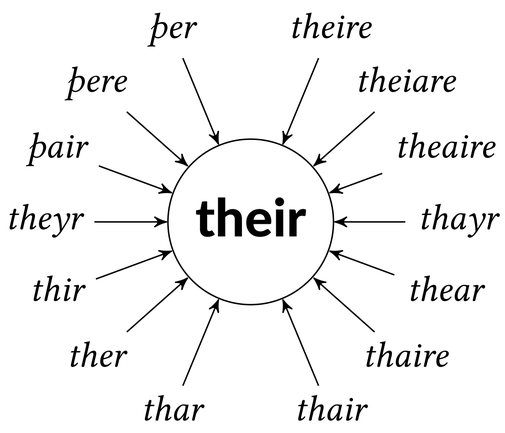
Few-shot and zero-shot learning for historical text normalization

Marcel Bollmann, Natalia Korchagina, Anders Søgaard

@mmbollmann
marcel@di.ku.dk

Main task

- Mapping words in historical text to modern equivalent



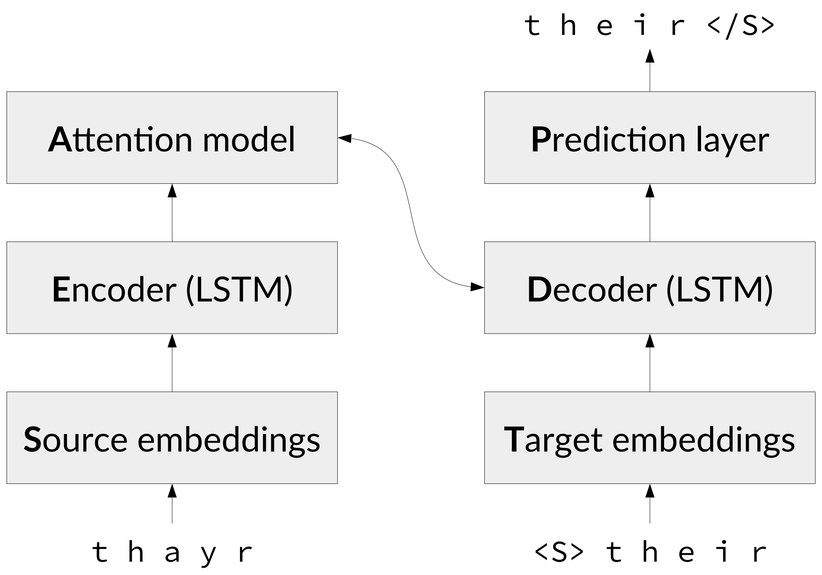
Auxiliary tasks

- **Autoencoding** *(Wikipedia data)*
- **Grapheme-to-phoneme** mapping *(Deri & Knight, 2016)*
- **Lemmatization** *(UniMorph data)*

- **Eight languages** *(Bollmann, 2019)*
  - English
  - German
  - Hungarian
  - Icelandic
  - Portuguese
  - Slovene
  - Spanish
  - Swedish

We analyze different **multi-task learning** strategies for a low-resource sequence-to-sequence task.

Model

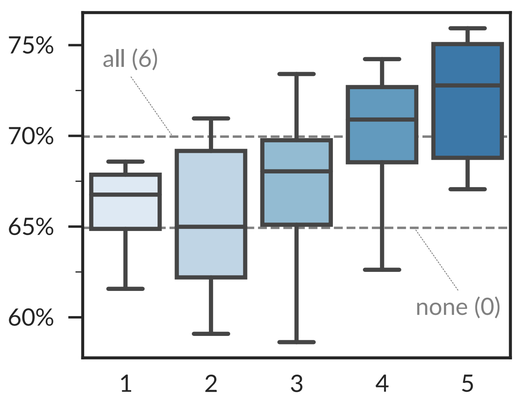


We use a standard, character-level encoder–decoder architecture and look at **what to share**.

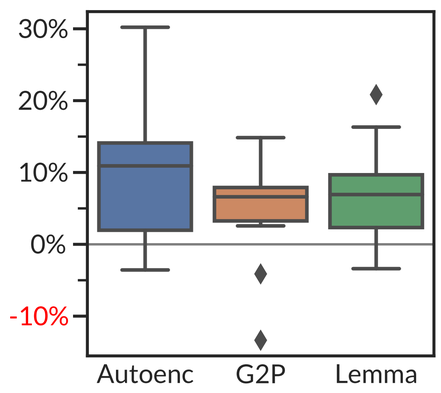
- We divide the model into six components and examine all possible combinations of *sharing* vs. *not sharing* each component.

  "Sharing" means that the same parameters (e.g. weight matrices) are used for main and auxiliary tasks, "not sharing" means that parameters are kept separate for each task.

Results



Normalization accuracy on English by number of shared components



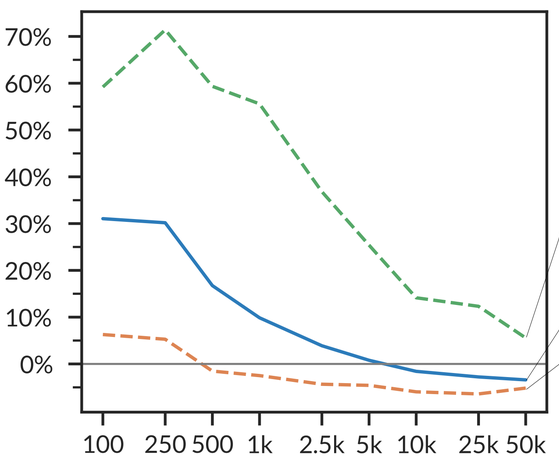
Error reduction by auxiliary task
(over all languages, for the best-performing model configuration)

We find empirically that **sharing more is better.**

- Sharing *all but one* component is best on average.

  The best-performing configuration is sharing *everything except the target embeddings*, but we see comparable results when keeping a task-specific decoder or prediction layer instead.

  Training data for the main task is limited to 1000 tokens here.



Identity subset:
tokens that should stay identical according to the reference normalization

Average over all tokens

Non-identity subset:
tokens that require changing characters in the normalization

Error reduction by training set size
(avg. over all languages, for the best-performing model configuration)

However, the beneficial effect of multi-task learning **goes away** with more (main task) training data.

- MTL mainly helps the model learn the *autoencoding part* of the normalization task, as the error reduction is highest for the subset of tokens that stay *identical* in the normalization.

References

- Marcel Bollmann, 2019. A Large-Scale Comparison of HIstorical Text Normalization Systems. In Proceedings of NAACL 2019.
- Aliya Deri & Kevin Knight, 2016. Grapheme-to-phoneme models for (almost) any language. In Proceedings of ACL 2016.

This research was supported by ERC Starting Grant LOWLANDS No. 313695.

Marcel Bollmann was partly funded from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 845995.