Evaluating historical text normalization systems: How well do they generalize?

Alexander Robertson
School of Informatics
University of Edinburgh
alexander.robertson@ed.ac.uk

Sharon Goldwater
School of Informatics
University of Edinburgh
sgwater@inf.ed.ac.uk

Abstract

We highlight several issues in the evaluation of historical text normalization systems that make it hard to tell how well these systems would actually work in practice—i.e., for new datasets or languages; in comparison to more naïve systems; or as a preprocessing step for downstream NLP tools. We illustrate these issues and exemplify our proposed evaluation practices by comparing two neural models against a naïve baseline system. We show that the neural models generalize well to unseen words in tests on five languages; nevertheless, they provide no clear benefit over the naïve baseline for downstream POS tagging of an English historical collection. We conclude that future work should include more rigorous evaluation, including both intrinsic and extrinsic measures where possible.

1 Introduction

Historical text normalization systems aim to convert historical wordforms to their modern equivalents, in order to make historical documents more searchable or to improve the performance of downstream NLP tools. In historical texts, a single word type may be realized with several different orthographic forms, which may not correspond to the modern form. For example, the modern English word *said* might be realized as *sayed*, *seyd*, *said*, *sayd*, etc. Spellings change over time, but also vary within a single time period and even within a single author, since orthography only became standardized in many languages fairly recently.

Over the years, researchers have proposed normalization methods based on rules and/or edit distances (Baron and Rayson, 2008; Bollmann, 2012; Hauser and Schulz, 2007; Bollmann et al., 2011; Pettersson et al., 2013a; Mitankin et al., 2014; Pettersson et al., 2014), statistical machine translation (Pettersson et al., 2013b; Scherrer and Erjavec, 2013), and most recently neural network models (Bollmann and Søgaard, 2016; Bollmann et al., 2017; Korchagina, 2017). However, most of these systems have been developed and tested on a single language (or even a single corpus), and many have not been compared to the naïve but strong baseline that only changes words seen in the training data, normalizing each to its most frequent modern form observed during training.[1] These issues make it hard to tell which methods generalize across languages and corpora, and how they compare to each other. Moreover, researchers have rarely examined whether their systems actually improve performance on downstream tasks.

This paper brings together best practices for evaluating historical text normalization systems, highlighting in particular the need to report results on unseen tokens and to consider the naïve baseline. We focus our evaluation on two recent neural models: one that has been previously tested only on a German collection that is not widely available (Bollmann et al., 2017), and one that is adapted from work on morphological re-inflection, but has not been used for historical text normalization (Aharoni et al., 2017). Both are encoder-decoder models; the former with soft attention, and the latter with hard monotonic attention.

We present results on five languages, for both seen and unseen words and for various amounts of training data. The soft attention model performs surprisingly poorly on seen words, so that its overall performance is worse than the naïve baseline and several earlier models (Pettersson et al., 2014). However, on unseen words (which we argue are what matters), both neural models do well.

Unfortunately, these positive results did not

[1] Some authors have focussed on *unsupervised* normalization, where the naïve baseline is to leave words unchanged (Mitankin et al., 2014; Hauser and Schulz, 2007). We consider only *supervised* systems in the remainder of this paper.

translate into improvements when we tested the English-trained models on a downstream POS tagging task using a different historical collection spanning a similar time range. Normalizing the text gave better tag accuracy than not normalizing, but neither neural model convincingly outperformed the naïve normalizer. Although these results are disappointing, the clear evaluation standards laid out here should benefit future work in this area.

2 Task setting and issues of evaluation

We follow previous work in training our systems on pairs (h, m) of historical tokens and their gold standard modern forms.[2] Note that at test time, most of the h tokens will have been seen before in the training data (due to Zipf's law), and for these tokens it is very difficult to beat a baseline that normalizes each h to the most common m seen for it in training.[3] Thus, in practice, normalization systems should typically only be applied to *unseen* tokens. It is therefore critical to report both dataset statistics and experimental results for unseen tokens.

Unfortunately, some recent papers have only reported accuracy on all tokens, and only in comparison to other (non-baseline) systems (Bollmann and Søgaard, 2016; Bollmann et al., 2017; Korchagina, 2017). These figures can be misleading if systems underperform the naïve baseline on seen tokens (which we show does happen in practice). To see why, suppose 80% of test tokens were seen in training, and the baseline gets 90% of them right, while system A gets 80% and system B gets only 70%. Meanwhile the baseline gets only 50% of unseen tokens right, whereas systems A and B get 70% and 90%, respectively. A's accuracy is higher *overall* than B's (78% vs 74%), but *both* systems underperform the baseline (82%). More importantly, the best system (90% accuracy overall) is achieved by applying the baseline to seen tokens, and the system that generalizes best (B) to unseen tokens; it is irrelevant that A scores higher overall than B.

Stemming from the reasoning above, we argue that a full evaluation of any spelling normalization system requires more complete dataset statistics and experimental results. In describing the training and test sets, researchers should not only report the number of **types** and **tokens**, but also the percentage of **unseen tokens** in the test (or dev) set and the percentage of **training items** (h, m) **where** $h = m$. This last statistic measures the degree of spelling variation, which varies considerably between corpora.

As for reporting results, we have argued that accuracy should be reported separately for **seen vs unseen tokens**, and overall results compared to the **naïve memorization baseline**. Since historical spelling normalization is typically a low-resource task, systems should also ideally be tested with **varying amounts of training data** to assess how much annotation might be required for a new corpus (Pettersson et al., 2014; Bollmann and Søgaard, 2016; Korchagina, 2017). Finally, since these systems may be deployed on corpora other than those they were trained on, and used as preprocessing for other tasks, we advocate reporting **performance on a downstream task and/or different corpus**. To our knowledge the only previous supervised learning system to do so is Pettersson et al. (2013b).

3 Models

We focus on two neural encoder-decoder models for spelling normalization, comparing them against the memorization baseline and to previous results from Pettersson et al. (2014). The first model (Bollmann et al., 2017)[4] uses a fairly standard architecture with a bi-directional LSTM encoder and an LSTM decoder with soft attention (Xu et al., 2015), and is trained using cross-entropy loss.

The second model is a new approach to spelling normalization, which adapts the morphological reinflection system of Aharoni et al. (2017).[5] The reinflection model generates the characters in an inflected wordform $(y_{1:n})$, given the characters of its lemma $(x_{1:m})$ and a set of corresponding morphological features (f). Rather than using a soft attention mechanism that computes a weight vector over the entire sequence, this model exploits the generally monotonic character alignment between $x_{1:m}$ and $y_{1:n}$ and attends to only a single encoded input character at a time during decoding.

Architecturally, the model uses a standard bi-directional encoder. The decoder steps through the characters of the input and considers jointly the output of the previous step, the morphological features, and the currently attended encoded input. It outputs

[2] It would be possible to train on full texts rather than isolated tokens, which could improve results for ambiguous forms. However, previous models have not addressed this setting, nor do we, leaving this for future work.

[3] Our version breaks ties by choosing the first m observed.

[4] https://bitbucket.org/mbollmann/acl2017

[5] https://github.com/roeeaharoni/morphological-reinflection

	Tokens	h typ	m typ	%nc	%uns
Eng	148/16/17k	19.4k	10.6k	73.9	8.6
Ger	39/5/5k	9.0k	8.4k	84.8	14.8
Hun	137/17/17k	45.5k	25.8k	15.4	24.1
Ice	52/6/6k	9.7k	8.5k	48.0	11.3
Swe	28/2/34k	8.3k	6.5k	65.9	22.4

Table 1: Dataset statistics: the number of tokens in train/dev/test sets; *h*istorical and *m*odern word types and % of "no-change" tokens ($h = m$) in the training sets; and the % of dev set tokens that are unseen in training.

either a character or an advance symbol (to advance the focus of attention for the next time step). It is trained on an oracle sequence of write/advance actions $s_{1:q}$ which are generated from an automatic alignment of the input and output sequences. The model maximizes $p(s_{1:q}|x_{1:m}, f)$. For details, see Aharoni et al. (2017).

We adapt the model to our purpose by removing the morphological features f, maximising only $p(s_{1:q}|x_{1:m})$. The monotonic assumption is well-suited to our task, since fewer than 0.4% of edit operations require non-monotonic alignments (i.e. character transpositions) in any of our datasets.

Other than removing the need for morphological features from the hard attention model, and increasing the number of training epochs to 50 for both models, we did no further hyperparameter tuning, since our goal was to assess the "off-the-shelf" performance of these systems.

4 Experiments

We use the same datasets as Pettersson et al. (2014), with data from five languages over a range of historical periods.[6] We use the same train/dev/test splits as Pettersson; dataset statistics are shown in Table 1. Because we do no hyperparameter tuning, we do not use the development sets, and all results are reported on the test sets.

Each system was tested as recommended above, with accuracy reported separately on seen and unseen items, and for different training data sizes. To evaluate the downstream effects of normalization, we applied the models to a collection of unseen documents and then tagged them with the Stanford POS tagger, which comes pre-trained on modern English. The documents are from the Parsed Corpus of Early English Correspondence (PCEEC) (Taylor et al., 2006), comprised of 84 letter collections from the 15th-17th centuries. (Our English normalization training data is from the 14th-17th centuries.) PCEEC contains roughly 2.2m manually POS-tagged tokens but no spelling annotation. Because it uses a large and somewhat idiosyncratic set of POS tags, we converted these to better match the Stanford tags before evaluating (though the match still isn't perfect; accuracy would be higher in all cases if the tag sets were identical). Baselines are provided by tagging the unnormalized text and the output of the naïve normalization baseline.

Results: normalization accuracy Table 2 gives test set results for all models, broken down into seen and unseen items where possible. [7] The split into seen/unseen highlights the fact that neither of the neural models does as well on seen items as the baseline; indeed the soft attention model is considerably worse in English and Hungarian, the two largest datasets.[8] The result is that this model actually underperforms the baseline when applied to all tokens, although a hybrid model (baseline for seen, soft attention for unseen) would outperform the baseline. Nevertheless, the hard attention model performs best on unseen tokens in all cases, often by a wide margin, and also yields competitive overall performance.

We also compared the accuracy of the two neural models at different training data sizes starting from 1k tokens. On *seen* tokens, the baseline was best in all cases except for 1k tokens in Hungarian and Icelandic (where the soft attention model was slightly better) and the largest two data sizes in German (where the hard attention model was slightly better). This supports our claim that learned models should typically only be applied to *unseen* tokens.

Accuracy on unseen tokens is shown in Figure 1. Note that the set of unseen items gets smaller

[6] English: Markus (1999); German: Scheible et al. (2011); Hungarian: Simon (2014); Icelandic: Rögnvaldsson et al. (2012); Swedish: Fiebranz et al. (2011). For details of their dates and contents, see Pettersson et al. (2014).

[7] We obtained our datasets from Pettersson et al. but our baseline results are slightly different from what they report. The differences (theirs–ours) are -0.1, 0.2, 0.4, 1.2, 0.6 for Eng, Ger, Hun, Ice, Swe respectively. This could be due to differences in tie-breaking methods, or to another unknown factor. These differences suggest using caution in directly comparing their non-baseline results to ours.

[8] When we varied the training data sizes, we found that the soft attention model actually gets *worse* on seen tokens in all languages as the training data increases beyond a relatively small size. We have no good explanation for this, and it's possible that tuning the parameters would help.

	English			German			Hungarian			Icelandic			Swedish		
	A	S	U	A	S	U	A	S	U	A	S	U	A	S	U
Hybrid	92.9			95.1			76.4			**84.6**			90.8		
GIZA++ un	**94.3**			**96.6**			79.9			71.8			**92.9**		
GIZA++ bi	92.4			95.5			80.1			71.5			92.5		
Mem. baseline	91.5	**96.9**	30.5	94.1	96.9	30.5	73.6	**96.0**	2.9	80.3	**86.8**	28.3	85.4	**98.1**	41.4
Soft attention	89.9	93.7	46.9	94.3	98.1	72.4	79.8	89.4	49.6	83.1	85.9	60.1	89.7	97.2	63.8
Hard attention	93.0	96.6	**52.4**	96.5	**99.3**	**80.5**	**88.0**	95.3	**65.0**	83.5	86.2	**61.4**	90.7	97.9	**65.7**

Table 2: Tokens normalized correctly (%) for each dataset. Upper half: results on (A)ll tokens reported by Pettersson et al. (2014) for a hybrid model (apply memorization baseline to seen tokens and an edit-distance-based model to unseen tokens) and two SMT models (which align character unigrams and bigrams, respectively). Lower half: results from our experiments, including accuracy reported separately on (S)een and (U)nseen tokens.

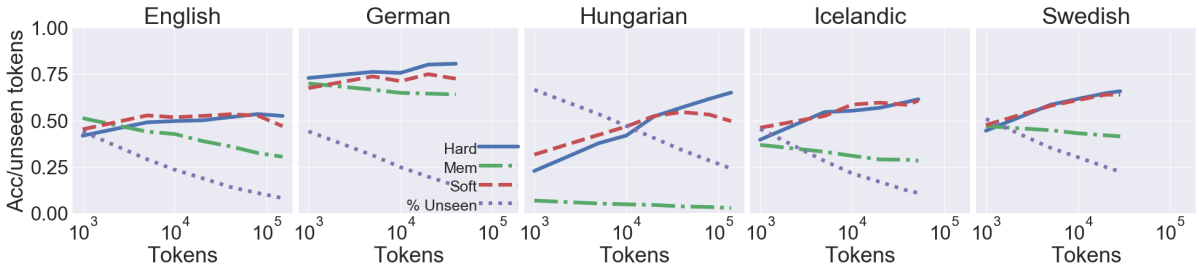

Figure 1: Proportion of unseen tokens, and normalization accuracy on those tokens, as training data size is varied.

and presumably more difficult as training data size increases, so the baseline gets worse. In contrast, the neural models are able to maintain or increase performance on this set. We expected that the bias toward monotonic alignments would help the hard attention model at smaller data sizes, but it is the soft attention model that seems to do better there, while the hard attention model does better in most cases at the larger data sizes. Note that Bollmann et al. (2017) trained their model on individual manuscripts, with no training set containing more than 13.2k tokens. The fact that this model struggles with larger data sizes, especially for seen tokens, suggests that the default hyperparameters may be tuned to work well with small training sets at the cost of underfitting the larger datasets.

Results: POS tagging Based on our results above, we tested the neural models by applying them only to unseen tokens in the PCEEC, and normalizing seen tokens using the naïve baseline in all cases. The PCEEC is a heterogeneous collection, so baseline tagger accuracy on the unnormalized text ranges from 52.0% to 82.6%, with an average of 71.0% (σ: 6.8). Figure 2 shows the effects of normalizing using the different methods.

Although normalizing provides a clear benefit, in most cases the neural models are no better than normalizing using the baseline method. The exception

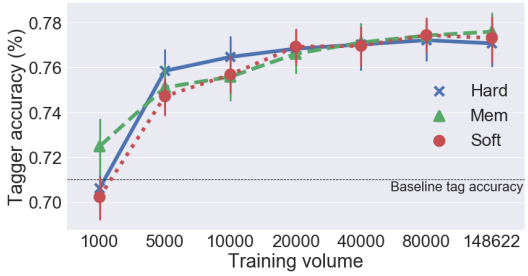

Figure 2: Average POS tagging accuracy on the unnormalized PCEEC texts (bottom of plot) and using three different normalization methods, as a function of the amount of data used to train the normalization systems.

is at 5k and 10k training items, where a two-tailed t-test shows that the hard attention model is significantly better than the other methods ($p < 0.01$). We also tried preprocessing both the normalization and tagging datasets by lowercasing all tokens; this resulted in small improvements in most cases (about 1 point) but any remaining differences were to the benefit of the baseline method.

Our findings differ from those of Pettersson et al. (2013b), who reported that their SMT-based system did work better than the baseline normalizer for POS tagging in Icelandic and verb identification in Swedish. Our contrasting findings could derive either from our use of different models or different datasets; nevertheless, they highlight the fact that

intrinsic improvements do not always translate into extrinsic ones.

5 Conclusion

We have highlighted some important issues in the evaluation of historical text normalization systems: in particular, the need to report accuracy on unseen tokens and to compare performance to a naïve memorization baseline. Following these recommendations, we evaluated two neural models, one of which is new to this task. Across five languages, both models greatly outperformed the baseline on unseen tokens, with the soft attention model doing a bit better for smaller data sizes, and the hard attention model doing a bit better for larger ones. However, these improvements did not translate into clearly better POS tagging downstream.

Despite these mixed results, we hope that the evaluation guidelines presented here will help promote work in this area, in order to eventually provide better tools for working with historical text collections.

6 Acknowledgements

We thank Pettersson, Megyesi, and Nivre for the datasets, and Aharoni, Goldberg, and Ramat-Gan and Bollmann, Bingel, and Søgaard for making their code available. This work was supported in part by the EPSRC Centre for Doctoral Training in Data Science, funded by the UK Engineering and Physical Sciences Research Council (grant EP/L016427/1) and the University of Edinburgh.

References

Roee Aharoni, Yoav Goldberg, and Israel Ramat-Gan. 2017. Morphological inflection generation with hard monotonic attention. In *Proceedings of ACL*.

Alistair Baron and Paul Rayson. 2008. VARD2: A tool for dealing with spelling variation in historical corpora. In *Postgraduate Conference in Corpus Linguistics*.

Marcel Bollmann. 2012. Automatic normalization of historical texts using distance measures and the Norma tool. In *Proceedings of the Second Workshop on Annotation of Corpora for Research in the Humanities (ACRH-2), Lisbon, Portugal*.

Marcel Bollmann, Joachim Bingel, and Anders Søgaard. 2017. Learning attention for historical text normalization by learning to pronounce. In *Proceedings of ACL 2017*.

Marcel Bollmann, Florian Petran, and Stefanie Dipper. 2011. Applying rule-based normalization to different types of historical texts - an evaluation. In *Language and Technology Conference*. Springer, pages 166–177.

Marcel Bollmann and Anders Søgaard. 2016. Improving historical spelling normalization with bi-directional LSTMs and multi-task learning. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics*.

Rosemarie Fiebranz, Erik Lindberg, Jonas Lindström, and Maria Ågren. 2011. Making verbs count: the research project "Gender and Work" and its methodology. *Scandinavian Economic History Review* 59(3):273–293.

Andreas W Hauser and Klaus U Schulz. 2007. Unsupervised learning of edit distance weights for retrieving historical spelling variations. In *Proceedings of the First Workshop on Finite-State Techniques and Approximate Search*. pages 1–6.

Natalia Korchagina. 2017. Normalizing Medieval German texts: from rules to deep learning. In *Proceedings of the NoDaLiDa 2017 Workshop on Processing Historical Language*. Linköping University Electronic Press, 133, pages 12–17.

Manfred Markus. 1999. *Manual of ICAMET (Innsbruck Computer Archive of Machine-Readable English Texts)*. Leopold-Franzens-Universitat Innsbruck.

Petar Mitankin, Stefan Gerdjikov, and Stoyan Mihov. 2014. An approach to unsupervised historical text normalisation. In *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*. ACM, New York, NY, USA, DATeCH '14, pages 29–34.

Eva Pettersson, Beáta Megyesi, and Joakim Nivre. 2013a. Normalisation of historical text using context-sensitive weighted Levenshtein distance and compound splitting. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NoDaLiDa 2013); Oslo University; Norway. NEALT Proceedings Series 16*. Linköping University Electronic Press, 085, pages 163–179.

Eva Pettersson, Beáta Megyesi, and Joakim Nivre. 2014. A multilingual evaluation of three spelling normalisation methods for historical text. In *LaTeCH@ EACL*. pages 32–41.

Eva Pettersson, Beáta Megyesi, and Jörg Tiedemann. 2013b. An SMT approach to automatic annotation of historical text. In *Proceedings of the Workshop on Computational Historical Linguistics at NoDaLiDa 2013; Oslo; Norway. NEALT Proceedings Series 18*. Linköping University Electronic Press, 087, pages 54–69.

Eiríkur Rögnvaldsson, Anton Karl Ingason, Einar Freyr Sigurðsson, and Joel Wallenberg. 2012. The Icelandic parsed historical corpus (IcePaHC). In *LREC*. pages 1977–1984.

Silke Scheible, Richard J Whitt, Martin Durrell, and Paul Bennett. 2011. A gold standard corpus of Early Modern German. In *Proceedings of the 5th linguistic annotation workshop*. Association for Computational Linguistics, pages 124–128.

Yves Scherrer and Tomaž Erjavec. 2013. Modernizing historical Slovene words with character-based SMT. In *Proceedings of the 4th Biennial International Workshop on Balto-Slavic Natural Language Processing*. Association for Computational Linguistics, Sofia, Bulgaria, pages 58–62.

Eszter Simon. 2014. Corpus building from Old Hungarian codices .

Ann Taylor, Arja Nurmi, Anthony Warner, Susan Pintzuk, and Terttu Nevalainen. 2006. The York-Helsinki Parsed Corpus of Early English Correspondence (PCEEC). Department of Linguistics, University of York.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*. pages 2048–2057.