Multi-task learning for historical text normalization: Size matters

Marcel Bollmann, Anders Søgaard, Joachim Bingel
Dept. of Computer Science
University of Copenhagen
Denmark
{marcel,soegaard,bingel}@di.ku.dk

Abstract

Historical text normalization suffers from small datasets that exhibit high variance, and previous work has shown that multi-task learning can be used to leverage data from related problems in order to obtain more robust models. Previous work has been limited to datasets from a specific language and a specific historical period, and it is not clear whether results generalize. It therefore remains an open problem, *when* historical text normalization benefits from multi-task learning. We explore the benefits of multi-task learning across 10 different datasets, representing different languages and periods. Our main finding—contrary to what has been observed for other NLP tasks—is that multi-task learning mainly works when target task data is very scarce.

1 Introduction

Historical text normalization is the problem of translating historical documents written in the absence of modern spelling conventions and making them amenable to search by today's scholars, processable by natural language processing models, and readable to laypeople. In other words, historical text normalization is a text-to-text generation, where the input is a text written centuries ago, and the output is a text that has the same contents, but uses the orthography of modern-day language. In this paper, we limit ourselves to word-by-word normalization, ignoring the syntactic differences between modern-day languages and their historic predecessors.

Resources for historical text normalization are scarce. Even for major languages like English and German, we have very little training data for inducing normalization models, and the models we induce may be very specific to these datasets and not scale to writings from other historic periods—or even just writings from another monastery or by another author.

Bollmann and Søgaard (2016) and Bollmann et al. (2017) recently showed that we can obtain more robust historical text normalization models by exploiting synergies across historical text normalization datasets and with related tasks. Specifically, Bollmann et al. (2017) showed that multi-task learning with German grapheme-to-phoneme translation as an auxiliary task improves a state-of-the-art sequence-to-sequence model for historical text normalization of medieval German manuscripts.

Contributions We study *when* multi-task learning leads to improvements in historical text normalization. Specifically, we evaluate a state-of-the-art approach to historical text normalization (Bollmann et al., 2017) with and without various auxiliary tasks, across 10 historical text normalization datasets. We also include an experiment in English historical text normalization using data from Twitter and a grammatical error correction corpus (FCE) as auxiliary datasets. Across the board, we find that, unlike what has been observed for other NLP tasks, *multi-task learning only helps when target task data is scarce*.

2 Datasets

We consider 10 datasets from 8 different languages: German, using the Anselm dataset (taken from Bollmann et al., 2017) and texts from the RIDGES corpus (Odebrecht et al., 2016)[1]; English, Hungarian, Icelandic, and Swedish (from Pettersson, 2016); two versions of Slovene using different alphabets (Bohorič or Gaj; from Ljubešić

[1] https://korpling.org/ridges/

Proceedings of the Workshop on Deep Learning Approaches for Low-Resource NLP, pages 19–24
Melbourne, Australia July 19, 2018.

Dataset/Language		Time Period	Tokens			Source of Splits
			Train	Dev	Test	
DE_A	German (Anselm)	14^{th}–16^{th} c.	233,947	45,996	45,999	Bollmann et al. (2017)
DE_R	German (RIDGES)	1482–1652	41,857	9,712	9,587	–
EN	English	1386–1698	147,826	16,334	17,644	Pettersson (2016)
ES	Spanish	15^{th}–19^{th} c.	97,320	11,650	12,479	–
HU	Hungarian	1440–1541	134,028	16,707	16,779	Pettersson (2016)
IS	Icelandic	15^{th} c.	49,633	6,109	6,037	Pettersson (2016)
PT	Portuguese	15^{th}–19^{th} c.	222,525	26,749	27,078	–
SL_B	Slovene (Bohorič)	1750–1840s	50,023	5,841	5,969	Ljubešić et al. (2016)
SL_G	Slovene (Gaj)	1840s–1899	161,211	20,878	21,493	Ljubešić et al. (2016)
SV	Swedish	1527–1812	24,458	2,245	29,184	Pettersson (2016)

Table 1: Historical datasets used in our experiments

et al., 2016); as well as Spanish and Portuguese texts from the Post Scriptum corpus (Vaamonde, 2017)[2].

For the Anselm dataset, we concatenate the individual training, development, and test sets from Bollmann et al. (2017) to obtain a single dataset. For RIDGES, we use 16 texts and randomly sample 70% of all sentences from each text for the training set, and 15% for the dev/test sets. The Spanish and Portuguese datasets consist of manually normalized subsets of the Post Scriptum corpus; here, we randomly sample 80% (train) and 10% (dev/test) of all sentences per century represented in the corpus. Dataset splits for the other languages are taken from Pettersson (2016) and Ljubešić et al. (2016).

We preprocessed all datasets to remove punctuation, perform Unicode normalization, replace digits that do not require normalization with a dummy symbol, and lowercase all tokens.

Table 1 gives an overview of all historical datasets, the approximate time period of historical texts that they cover, as well as the size of the dataset splits. Note that, to the best of our knowledge, the Spanish, Portuguese, and German RIDGES datasets have not been used in the context of automatic historical text normalization before.

Table 2 additionally gives some examples of historical word forms and their gold-standard normalizations from each of these datasets.[3]

3 Experimental setup

Model We use the same encoder–decoder architecture with attention as described in Bollmann et al. (2017).[4] This is a fairly standard model consisting of one bidirectional LSTM unit in the encoder and one (unidirectional) LSTM unit in the decoder. The input for the encoder is a single historical word form represented as a sequence of characters and padded with word boundary symbols; i.e., we only input single tokens in isolation, not full sentences. The decoder attends over the encoder's outputs and generates the normalized output characters.

Hyperparameters We use the same hyperparameters across all our experiments: The dimensionality of the embedding layer is 60, the size of the LSTM layers is set to 300, and we use a dropout rate of 0.2. We use the Adam optimizer (Kingma and Ba, 2014) with a character-wise cross-entropy loss. Training is done on mini-batches of 50 samples with early stopping based on validation on the individual development sets. The hyperparameters were set on a randomly selected subset of 50,000 tokens from each of the following datasets: English, German (Anselm), Hungarian, Icelandic, and Slovene (Gaj).

Multi-task learning Bollmann et al. (2017) also describe a multi-task learning (MTL) scenario where the encoder–decoder model is trained on two datasets in parallel. We perform similar experiments on pairwise combinations of our datasets.

[2] `http://ps.clul.ul.pt`

[3] Note that normalization guidelines differ between the datasets, and normalizations do not always constitute a modern word form—e.g. in the case of extinct lexemes—or the correct inflected form in the given context.

[4] The implementation is taken from: `https://bitbucket.org/mbollmann/acl2017`

DE$_A$	deſe wort ſpricht vnſer liber here iheſus criſtus czu eyme iczlychen menſchen
	diese wort spricht unser lieber herr jesus christus zu einem ieteslichen menschen
DE$_R$	ſeind ſÿ doch alle auſz den vier elementen gemiſchet vnd eins feüchter deñ das ander
	sind sie doch alle aus den vier elementen gemischt und eins feuchter denn das andere
EN	whan your graciouse erthely persoune from your inward spirit ys dessolued
	when your gracious earthly person from your inward spirit is dissolved
ES	anque tomeys mui mucho travajo tengola guardada pa quando dios sea servido
	aunque toméis muy mucho trabajo téngola guardada para cuando dios sea servido
HU	o zauoc ėſmė felèmèluē kèzdėnc̣ ſirńoc èlmēnèc èzèkèt tolga ez a noemi azeꝛt iouo
	ő szavuk ismét felemelvén kezdének sírniuk elmenjek ezeket toldja ez a noémi azért jöve
IS	þá sem hanz gödverk voru i og þá vrdu hanns gödverk miklu þýngre enn ill
	þá sem hans góðverk voru í og þá urðu hans góðverk miklu þyngri en ill
PT	cõ a poenetencia que lhe derão pera avisar aos snres do sancto oficio
	com a penitência que lhe deram para avisar aos senhores do santo ofício
SL$_B$	ter ne bodi nevéren zhe ſe zherna perſt premozhi tezhe od nje rjav mòk
	ter ne bodi neveren če se črna prst premoči teče od nje rjav mok
SL$_G$	in privéže na vsak konec niti drobtino kruha in verže vse kokóšem breskevno vkuhanje lovre
	in priveže na vsak konec niti drobtino kruha in vrže vse kokošim breskvino vkuhanje lovre
SV	blef av rätten afsagdt det en syyn och rådhgångh nu nästkommande wårdagh hållas
	blev av rätten avsagt det en syn och rådgång nu nästkommande vårdag hållas

Table 2: Examples of input tokens (first line) and reference normalization (second line) for each of the historical datasets.

The question we ask here is whether training on pairs of datasets can improve over training on datasets individually, which pairings yield the best results, and what properties of the datasets are most predictive of this. In other words, we are interested in *when* multi-task learning works.

In the multi-task learning setting, the two datasets—or "tasks"—share all parts of the encoder–decoder model except for the final prediction layer, which is specific to each dataset. This way, most parts of the model are forced to learn language-independent representations. This is different from Luong et al. (2015) and related work in machine translation, where typically only the encoder or the decoder is shared. We do not explore these alternatives here.

During training, we iterate over both our datasets in parallel in a random order, with each parameter update now being based on 50 samples from *each* dataset. Since datasets are of different sizes, we define an epoch to be a fixed size of 50,000 samples. Validation is performed for both datasets after each epoch, and model states are saved independently for each one if its validation accuracy improved. This means that even if the ideal number of epochs is different for the datasets, only the best state for each dataset will be used in the end. Training ends only after the validation accuracy for *each* dataset has stopped improving.

Sparse data scenario The training sets in our experiments range from ca. 25,000 to 230,000 tokens. Generally, historical text normalization suffers from scarce resources, and our biggest datasets are considered *huge* compared to what scholars typically have access to. Creating gold-standard normalizations is cumbersome and expensive, and for many languages and historic periods, it is not feasible to obtain big datasets. Therefore, we also present experiments on reduced datasets; instead of taking the full training sets, we only use the first 5,000 tokens from each one.

In this case, for multi-task learning, we combine the small target datasets with the *full* auxiliary datasets. This procedure mimics a realistic scenario: If a researcher is interested in normalizing a language for which no manually normalized resource exists, they could conceivably create a small batch of manual normalizations for this language and then leverage an existing corpus in another language using multi-task learning.

Dataset	Full		Sparse	
	Single	MTL	Single	MTL
DE_A	88.00	87.78	65.99	71.93
DE_R	86.05	87.81	70.04	74.25
EN	93.95	93.46	75.43	81.02
ES	94.41	94.32	82.50	86.59
HU	89.43	88.56	49.21	54.86
IS	84.83	86.67	69.52	72.73
PT	93.45	93.36	78.61	81.97
SL_B	90.12	91.81	82.39	86.35
SL_G	94.79	94.53	89.54	91.03
SV	88.48	89.90	79.24	82.14

Table 3: Normalization accuracy (in percent) using the full or sparse training sets, both for the single-task setup and the best-performing multi-task (MTL) setup.

4 Results

We evaluate our models using normalization accuracy, i.e., the percentage of correctly normalized word forms. Table 3 compares the accuracy scores of our single-task baseline models and for multi-task learning, in both the full and the sparse data scenario. For multi-task learning, we report the test set performance of the best target-auxiliary task pair combination, as evaluated on development data. Figure 1 visualizes the results for all pairwise combinations of the multi-task models; here, we show the error reduction of multi-task learning over our single-task baseline to better highlight by how much the MTL setup changes the performance.

Full datasets We make two observations about the results for the full data scenario (the left side of Fig. 1): (i) the usefulness of multi-task learning depends more on the dataset that is being evaluated than the one it is trained together with; and (ii) for most datasets, multi-task learning is *detrimental* rather than beneficial.

One hypothesis about multi-task learning is that its usefulness correlates with either synergistic or complementary properties of the datasets. In other words, it is conceivable that the performance on one dataset improves most with an MTL setup when it is paired with another dataset that is either (i) very similar, or (ii) provides an additional signal that is useful for, but not covered in, the first dataset. The results in Figure 1 show that there can indeed be considerable variation depending on the exact dataset combination; e.g., the error reduction on Slovene (Bohorič) ranges from 5% (when paired with the Gaj dataset) to 33.2% (when paired with Swedish). At the same time, the question whether multi-task learning helps at all seems to depend mostly on the dataset being evaluated. With few exceptions, for most datasets, the error rate either always improves or always worsens, independently of the auxiliary task.

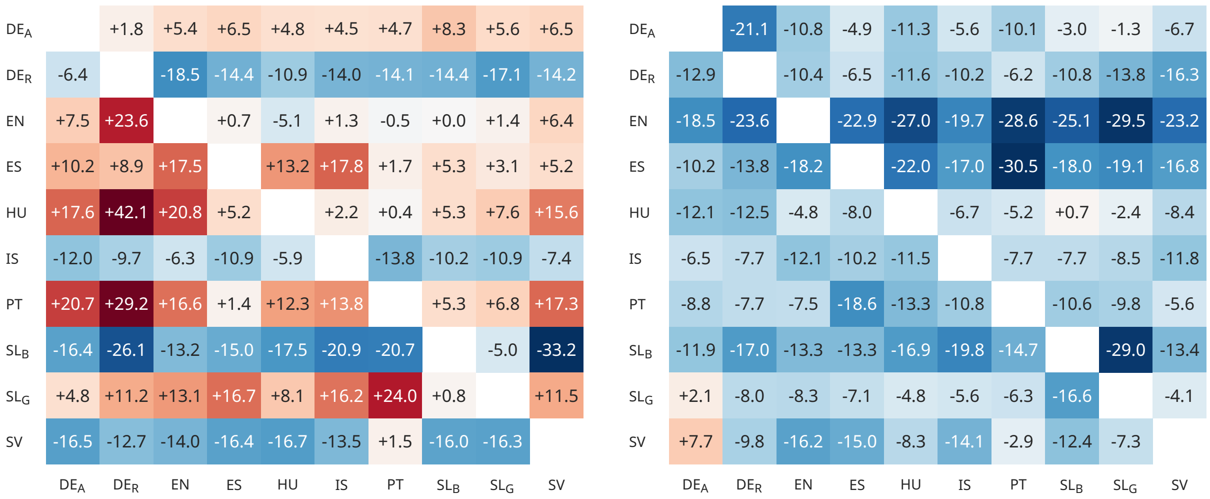
Auxiliary data	Accuracy
None	75.43
Best above	81.02
Twitter	81.72
FCE	78.53

Table 4: Normalization accuracy for English (sparse): Single and MTL from Table 3; and with non-historical auxiliary datasets (Twitter & FCE).

Considering the dataset statistics in Table 1, it appears that the *size of the training corpus* is the most important factor for these results. The four corpora that consistently benefit from MTL—German (RIDGES), Icelandic, Slovene (Bohorič), and Swedish—also have the smallest training sets, with about 50,000 tokens or less. For other tasks, different patterns have been observed (Martínez Alonso and Plank, 2017; Bingel and Søgaard, 2017); see Sec. 5.

Sparse datasets In the sparse data scenario where only 5,000 tokens are used for training (right side of Fig. 1), MTL almost always leads to improvements over the single-task training setup. This further confirms the hypothesis that *multi-task learning is beneficial for historical text normalization when the target task dataset is small.*

English with non-historical auxiliary data We also conduct a follow-up experiment on the (sparse) English dataset using a Twitter normalization dataset (Han and Baldwin, 2011) and a grammatical error corpus (Yannakoudakis et al., 2011) as auxiliary data. The results are presented in Table 4. Surprisingly, the Twitter dataset is actually more helpful than the best historical dataset; but of course, it is also in-language, unlike the historical datasets.



	DE$_A$	DE$_R$	EN	ES	HU	IS	PT	SL$_B$	SL$_G$	SV
DE$_A$		+1.8	+5.4	+6.5	+4.8	+4.5	+4.7	+8.3	+5.6	+6.5
DE$_R$	-6.4		-18.5	-14.4	-10.9	-14.0	-14.1	-14.4	-17.1	-14.2
EN	+7.5	+23.6		+0.7	-5.1	+1.3	-0.5	+0.0	+1.4	+6.4
ES	+10.2	+8.9	+17.5		+13.2	+17.8	+1.7	+5.3	+3.1	+5.2
HU	+17.6	+42.1	+20.8	+5.2		+2.2	+0.4	+5.3	+7.6	+15.6
IS	-12.0	-9.7	-6.3	-10.9	-5.9		-13.8	-10.2	-10.9	-7.4
PT	+20.7	+29.2	+16.6	+1.4	+12.3	+13.8		+5.3	+6.8	+17.3
SL$_B$	-16.4	-26.1	-13.2	-15.0	-17.5	-20.9	-20.7		-5.0	-33.2
SL$_G$	+4.8	+11.2	+13.1	+16.7	+8.1	+16.2	+24.0	+0.8		+11.5
SV	-16.5	-12.7	-14.0	-16.4	-16.7	-13.5	+1.5	-16.0	-16.3	

	DE$_A$	DE$_R$	EN	ES	HU	IS	PT	SL$_B$	SL$_G$	SV
DE$_A$		-21.1	-10.8	-4.9	-11.3	-5.6	-10.1	-3.0	-1.3	-6.7
DE$_R$	-12.9		-10.4	-6.5	-11.6	-10.2	-6.2	-10.8	-13.8	-16.3
EN	-18.5	-23.6		-22.9	-27.0	-19.7	-28.6	-25.1	-29.5	-23.2
ES	-10.2	-13.8	-18.2		-22.0	-17.0	-30.5	-18.0	-19.1	-16.8
HU	-12.1	-12.5	-4.8	-8.0		-6.7	-5.2	+0.7	-2.4	-8.4
IS	-6.5	-7.7	-12.1	-10.2	-11.5		-7.7	-7.7	-8.5	-11.8
PT	-8.8	-7.7	-7.5	-18.6	-13.3	-10.8		-10.6	-9.8	-5.6
SL$_B$	-11.9	-17.0	-13.3	-13.3	-16.9	-19.8	-14.7		-29.0	-13.4
SL$_G$	+2.1	-8.0	-8.3	-7.1	-4.8	-5.6	-6.3	-16.6		-4.1
SV	+7.7	-9.8	-16.2	-15.0	-8.3	-14.1	-2.9	-12.4	-7.3	

Figure 1: Percentage change of error of MTL over single-task models; rows are targets, columns auxiliary data. *Left:* full data; *right:* sparse data. Blue scores are improvements, reds increases in error.

5 Related work and conclusion

There has been considerable work on multi-task sequence-to-sequence models for other tasks (Dong et al., 2015; Luong et al., 2015; Elliott and Kádár, 2017). There is a wide range of design questions and sharing strategies that we ignore here, focusing instead on under what circumstances the approach advocated in (Bollmann et al., 2017) works.

Our main observation—that the size of the target dataset is most predictive of multi-task learning gains—runs counter previous findings for other NLP tasks (Martínez Alonso and Plank, 2017; Bingel and Søgaard, 2017). Martínez Alonso and Plank (2017) find that the label entropy of the auxiliary dataset is more predictive; Bingel and Søgaard (2017) find that the relative differences in the steepness of the two single-task loss curves is more predictive. Both papers consider sequence tagging problems with a small number of labels; and it is probably not a surprise that their findings do not seem to scale to the case of historical text normalization.

Acknowledgments

This research was supported by ERC Starting Grant LOWLANDS No. 313695 as well as by Trygfonden. Parts of this research were carried out by the first author at the Ruhr-Universität Bochum, Germany, supported by Deutsche Forschungsgemeinschaft (DFG), Grant DI 1558/4.

References

Joachim Bingel and Anders Søgaard. 2017. Identifying beneficial task relations for multi-task learning in deep neural networks. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 164–169. Association for Computational Linguistics.

Marcel Bollmann, Joachim Bingel, and Anders Søgaard. 2017. Learning attention for historical text normalization by learning to pronounce. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 332–344. Association for Computational Linguistics.

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: Technical Papers*, pages 131–139. The COLING 2016 Organizing Committee.

Daxiang Dong, Hua Wu, Wei He, Dianhai Yu, and Haifeng Wang. 2015. Multi-task learning for multiple language translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1723–1732. Association for Computational Linguistics.

Desmond Elliott and Àkos Kádár. 2017. Imagination improves multimodal translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 130–141. Asian Federation of Natural Language Processing.

Bo Han and Timothy Baldwin. 2011. Lexical normalisation of short text messages: Makn sens a #twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 368–378. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Nikola Ljubešić, Katja Zupan, Darja Fišer, and Tomaž Erjavec. 2016. Normalising Slovene data: historical texts vs. user-generated content. In *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, volume 16 of *Bochumer Linguistische Arbeitsberichte*, pages 146–155, Bochum, Germany.

Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2015. Multitask sequence to sequence learning. volume abs/1511.06114.

Héctor Martínez Alonso and Barbara Plank. 2017. When is multitask learning effective? semantic sequence prediction under varying data conditions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 44–53. Association for Computational Linguistics.

Carolin Odebrecht, Malte Belz, Amir Zeldes, Anke Lüdeling, and Thomas Krause. 2016. RIDGES Herbology: designing a diachronic multi-layer corpus. *Language Resources and Evaluation*, pages 1–31.

Eva Pettersson. 2016. *Spelling Normalisation and Linguistic Analysis of Historical Text for Information Extraction*. Doctoral dissertation, Uppsala University, Department of Linguistics and Philology, Uppsala.

Gael Vaamonde. 2017. Userguide for digital edition of texts in P. S. Post Scriptum. Technical report. Translated by Clara Pinto.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189. Association for Computational Linguistics.