A Neural Model for Part-of-Speech Tagging in Historical Texts

Christian Hardmeier
Uppsala University
Dept. of Linguistics and Philology
751 26 Uppsala, Sweden
first.last@lingfil.uu.se

Abstract

Historical texts are challenging for natural language processing because they differ linguistically from modern texts and because of their lack of orthographical and grammatical standardisation. We use a character-level neural network to build a part-of-speech (POS) tagger that can process historical data directly without requiring a separate spelling normalisation stage. Its performance in a Swedish verb identification and a German POS tagging task is similar to that of a two-stage model. We analyse the performance of this tagger and a more traditional baseline system, discuss some of the remaining problems for tagging historical data and suggest how the flexibility of our neural tagger could be exploited to address diachronic divergences in morphology and syntax in early modern Swedish with the help of data from closely related languages.

1 Introduction

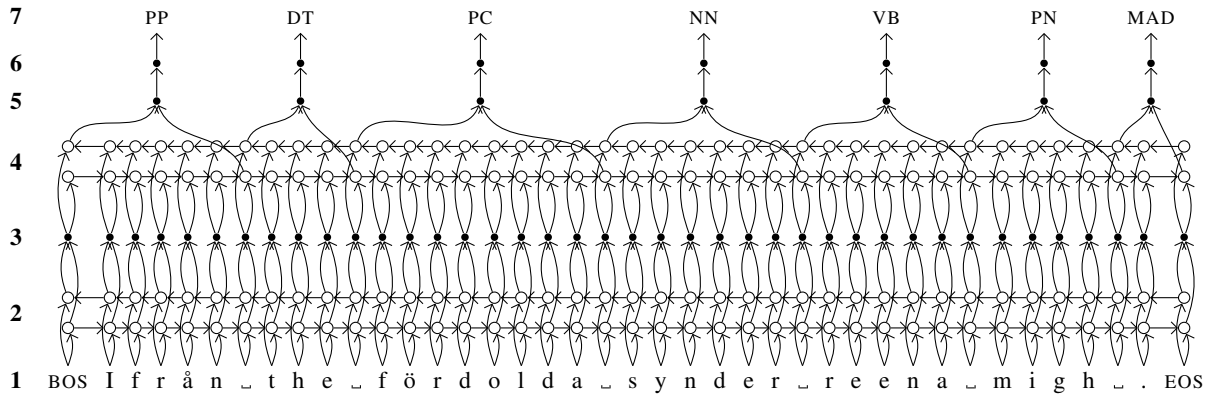
Most tools for automatic linguistic text annotation are based on supervised learning and trained on manually annotated text samples such as treebanks. This approach works best when the texts to be annotated are very similar to the language in the training corpora. The greater the differences, the more difficult it becomes to do automatic annotation with high accuracy. One application that poses particular challenges is automatic processing of historical texts. Language records from a few centuries ago are often still intelligible to modern readers, but they can nonetheless exhibit substantial divergence from later language use in terms of orthography, morphology, syntax, etc. Moreover, the languages we speak and write have undergone relatively recent processes of standardisation. Historically, there was much more variety in spelling and grammar both across and within texts, making the data sparseness problems we know from modern language processing even more acute. Standard approaches to deal with this challenge include manual or semi-automatic annotation of historical data sets to train language processing tools or automatic spelling normalisation to convert historical into modern spellings for the purpose of applying standard tools for modern language.[1] In this work, we present a neural network model to do part-of-speech (POS) tagging in historical texts. Our model uses a modern POS-tagged data set and a historical corpus with original and normalised spellings for training, but reads historical data without specific preprocessing at test time. We test the model on a Swedish verb identification and a German POS tagging task and analyse the output of the model to identify some remaining challenges to be addressed in future work.

2 Model Architecture

The core of our neural network is a POS tagger. The network takes as input a sentence in the form of a sequence of characters. For each character, it computes a representation in the form of a dense, approximately 50-dimensional vector that captures information about the character and its preceding and following context. The vector representations occurring at word boundaries are then used to predict a POS tag for each of the words in the sentence. At training time only, the model contains additional components

[1] For an overview of the relevant literature, we refer the reader to the recent PhD thesis by Pettersson (2016).

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 922–931, Osaka, Japan, December 11-17 2016.



1 Character embeddings **2** Bidirectional GRU layer **3** Linear combination layer **4** Bidirectional GRU layer
5 Linear combination **6** Hidden layer with tanh activation **7** Softmax over POS tag vocabulary

Figure 1: Neural network architecture

	Swedish	German
Alphabet size	97	105
Character embeddings (**1**)	50	
First bidirectional GRU layer (**2**)	100	
Second bidirectional GRU layer (**4**)	51	
Final hidden layer (**6**)	300	100 *or* 300
POS tagset size (**7**)	29	58

Table 1: Neural network layer sizes

to ensure that the context-dependent vector-space representations created by the model are similar for historical data in original and normalised form.

Our character-level POS tagging model is shown in Figure 1. It is inspired by the work of Ling et al. (2015). The input of the model is a sentence split into characters. No normalisation or preprocessing is done at this point, and the input vocabulary consists of all Unicode code points encountered in the historical training set or its normalised form. In addition to the uppercase and lowercase letters of the modern alphabet, this also includes various forms of punctuation and letters with different diacritics, some of which are specific to the transcriptions of historical texts. The input characters are first transformed into dense character embeddings using a lookup table with trainable weights (**1**). Then, the entire sequence is scanned with a bidirectional recurrent neural network (Schuster and Paliwal, 1997) composed of gated recurrent units (GRUs; **2**) (Cho et al., 2014). The output states of the GRUs are passed through a linear layer and fed as inputs into another GRU layer (**4**). Up to this point, we are still processing the data at the character level and taking into consideration the context of the entire sentence. Unlike the model by Ling et al. (2015), our tagger never creates cacheable word embeddings that are independent of the surrounding words. We expect that this optimisation, which is used to speed up tagging in the Ling et al. model, would be less effective for historical than for modern text because of the greater spelling variability.

The transition to the word level, a prerequisite for predicting word-level POS tags, is done in the next step by combining, for each word, the final state reached by the forward and backward part of the bidirectional layer **4** after processing the word in question. These states are combined linearly (**5**), fed into a hidden layer using the hyperbolic tangent activation function (**6**) and passed on to a final softmax layer that outputs a probability distribution over the POS tagset (**7**).

The layer sizes of our network are shown in Table 1. Owing to memory limitations of the hardware we trained our systems on (Nvidia K20 GPUs with 5 GB of RAM), we could not test larger layer sizes systematically. Increasing the size of the hidden layer **6** from 100 to 300 brought a consistent improvement of 1–2 percentage points in F-score or accuracy for all experiments on Swedish. For German, the results were less conclusive, and the overall best model has a hidden layer of size 100.

Swedish				German			
Modern POS-tagged corpus		*Sent.*	*Tokens*	*Modern POS-tagged corpus*		*Sent.*	*Tokens*
Stockholm-Umeå corpus	Training	73,243	1,153,545	NEGRA corpus	Training	19,602	337,702
	Validation	500	7,287		Validation	500	8,415
	Test	500	5,924		Test	500	8,979
Historical corpus				*Historical corpus*			
Gender and Work corpus	Training	540	28,237	GerManC	Training	2,048	43,298
	Validation	60	2,590		Validation	186	4,216
	Development	600	33,544				
	Test	300	14,672		Test	216	4,845

Table 2: Corpus data overview

3 Model Training

The situation we consider in our experiments is one in which we have access to a POS-tagged training corpus of modern language as well as an unrelated corpus of historical texts in original and modern spelling, but not a POS-tagged training corpus of historical text. This corresponds to the actual situation for Swedish. Our historical training corpus for German does in fact contain a small amount of gold-standard POS annotations. In this paper, these are not used other than for comparison and evaluation.

The training objective we optimise our models for is to *maximise POS tagging performance on the tagged corpus* whilst *ensuring that the RNN states generated from historical texts in original spelling are similar to those arising from the corresponding normalised forms.* To achieve this, we compute two types of training error at every training step. The first is obtained by feeding a training example from the modern POS-tagged training set into the neural network shown in Figure 1. The *POS training error* E_{POS} of the training example is defined as the cross-entropy of the predicted tag distribution with respect to the gold-standard solution. For the second, we take a training example from the historical corpus and separately calculate the context-dependent word representations of the original historical text and its normalised form using layers **1** to **5** of the neural network, but omitting the hidden layer **6** and the final softmax layer. The *normalisation training error* E_{norm} of the training example is the squared error between the representation generated from the historical spellings and the representation of the normalised forms. The training examples used for the calculation of the two error types are independent from each other and paired randomly. The overall training objective is a weighted combination of the two error types:

$$E_{\text{total}} = \lambda E_{\text{POS}} + (1-\lambda) E_{\text{norm}} \tag{1}$$

To train our model, we apply minibatch stochastic gradient descent with a learning rate of 0.01, together with gradient clipping (Pascanu et al., 2013) to a maximum ℓ_2 norm of 10. The minibatch size was set to 30. We found that this batch size tended to give better results than smaller batches. Larger values could not be tested because of memory restrictions of our computer systems. The input sentences from both the POS-tagged corpus and the historical training corpus are cut at word boundaries into segments of approximately 25 words. To improve training efficiency, minibatches are formed from segments of similar length. The systems are trained on 100,000 to 200,000 minibatches, which corresponds to a wall-time limit of approximately 48 hours per training run. The error on the validation set is checked after every 3,000 batches. The set of parameters selected for evaluation purposes is the one that achieved the lowest validation error during training.

4 Tasks, Data Sets and Baseline System

We apply our model to two different tasks known from the literature. For Swedish, we address the problem of verb identification in historical texts. This task was introduced by Pettersson and Nivre (2011). It is motivated by its use in a historical research project named *Gender and Work* (Fiebranz et al., 2011). The goal of the *Gender and Work* project is to study the activities that men and women, respectively, carried out for a living in early modern Sweden (1550–1800). One of the core methods used in this project

was the systematic identification of verb phrases describing such activities in historical documents. In the course of the project, Pettersson and her colleagues developed data sets and methods to support the automatic annotation of such verb phrases (Pettersson, 2016). For German, the availability of a corpus of historical texts with gold-standard POS annotations allows us to tackle the more general task of POS tagging for historical texts.

For each language, we need a modern corpus annotated with POS tags and a corpus of historical texts in original and normalised spelling. Additionally, we need historical data with verb annotations or POS tags to evaluate our systems. Table 2 shows an overview of the corpora used in our experiments. For Swedish, we closely follow the setup of the experiments of Pettersson (2016). As a modern resource annotated with POS tags, we use version 2.0 of the Stockholm-Umeå corpus (SUC), a fairly large balanced collection of Swedish texts from the 1990's (Gustafson-Capková and Hartmann, 2006). We removed the last 1,000 sentences of the corpus to be used, in equal parts, as validation and test sets. As a historical training corpus, we have the *Gender and Work* corpus (Fiebranz et al., 2011; Pettersson, 2016). The split of this corpus into different data sets corresponds to the experiments of Pettersson (2016). The *training* and *validation* sets (corresponding to the training and tuning sets of Pettersson's spelling normalisation experiments) are used for neural network training and validation. The *development* set (corresponding to Pettersson's spelling normalisation evaluation set, which she subsequently used as a development set for verb phrase identification) was used as a test set during development. Finally, the *test* set (corresponding to Pettersson's verb phrase evaluation set) was used as a held-out set for the final evaluation of our model.

For German, our modern POS-tagged resource is the NEGRA corpus (Skut et al., 1997). As for SUC, we removed the last 1,000 sentences for validation and testing. Our historical data for German comes from the gold-standard portion of the GerManC corpus (Scheible et al., 2011), a corpus of early modern German (1650–1800) annotated with normalised spelling, lemmas and POS tags. The manually annotated gold standard part of this corpus consists of 24 documents. We set aside two of the more recent documents each for validation ("Ursprung", 1772; "Wolfenbüttel 1", 1786) and testing ("Gottesdienst", 1770; "Anton Reiser", 1790) and use the rest as training data.

Our baseline systems are modelled on the best-performing approach of Pettersson (2016) and consist of a pipeline that first normalises the spelling of the historical texts to be as similar as possible to modern orthography and then applies standard natural language processing tools trained on modern resources. The spelling normalisation component is a character-based statistical machine translation (SMT) system (Pettersson et al., 2013) implemented with the Moses toolkit (Koehn et al., 2007). It is a phrase-based SMT model with phrase length 10, disabled reordering and a 10-gram language model with modified Kneser-Ney smoothing (Chen and Goodman, 1998). The feature weights are tuned with minimum error-rate training (Och, 2003) to optimise the character error rate of the output. The default values of the Moses training pipeline and decoder are used for all other settings. After spelling normalisation, we run the HunPos tagger (Halácsy et al., 2007) for verb identification and POS tagging. Our HunPos models for Swedish and German are trained on exactly the same modern data sets as our own neural network tagger. For German, we also have the possibility to train HunPos on historical text with gold-standard POS tags from the GerManC corpus as another point of comparison.

5 Results

Table 3 shows the results of our two best-performing neural network taggers together with some comparative figures. The POS weight λ refers to the parameter in the error function in Equation 1. With equal weights for the POS tagging error and the normalisation error, our system reaches an F-score of 0.8668 on the development set and 0.8427 on the test set. Precision is higher than recall on the development set, but on the test set they are fairly balanced. Increasing the POS weight to 0.8 leads to an improvement to 0.8695 on the development set, which also carries over to the test set and gives us an F-score of 0.8529, about one percentage point over the result with equal weights. Decreasing the POS weight to 0.2 gives lower scores (not reported here).

The most interesting point of comparison is, of course, the HunPos system with SMT normalisation that emerged as the best model from the study of Pettersson (2016). Our own implementation of this

	SUC tagging	Historical verb identification					
		Development set			Test set		
POS weight	Accuracy	Precision	Recall	F-score	Precision	Recall	F-score
$\lambda = 0.5$	0.9625	0.8927	0.8424	0.8668	0.8454	0.8400	0.8427
$\lambda = 0.8$	0.9637	0.8909	0.8490	0.8695	0.8612	0.8448	0.8529
$\lambda = 1.0$	0.9534	0.8013	0.6517	0.7188	0.7623	0.6566	0.7055
HunPos with SMT normalisation	–	0.8773	0.8776	0.8775	0.8477	0.8729	0.8601
HunPos without normalisation	0.9772	0.7683	0.6173	0.6846	0.7202	0.6130	0.6623

Table 3: Results for the Swedish verb identification task

		POS tagging accuracy			Historical verb identification					
		NEGRA	GerManC		Development set			Test set		
POS weight	Layer **6** size		*dev*	*test*	P	R	F	P	R	F
$\lambda = 0.8$	100	0.9695	0.8382	0.8615	0.8780	0.9000	0.8889	0.9022	0.9008	0.9015
$\lambda = 0.8$	300	0.9692	0.8036	0.8520	0.8386	0.9091	0.8724	0.8796	0.8768	0.8782
$\lambda = 0.5$	300	0.9667	0.8157	0.8594	0.8612	0.9023	0.8812	0.8994	0.8864	0.8928
$\lambda = 1.0$	300	0.9661	0.8183	0.8444	0.8281	0.8318	0.8299	0.8325	0.8192	0.8258
HunPos trained on NEGRA										
with SMT normalisation		–	0.8577	0.8625	0.9116	0.8909	0.9011	0.8981	0.8880	0.8930
without normalisation		0.9952	0.8107	0.8353	0.8338	0.7295	0.7782	0.8643	0.7744	0.8169
HunPos trained on GerManC		0.7737	0.9082	0.9154	0.9044	0.8818	0.8930	0.8820	0.8608	0.8713

Table 4: Results for German POS tagging and verb identification

system achieves an F-score of 0.8601 on the test set, about half a percentage point better than the result of 0.855 reported by Pettersson for her corresponding system. The difference could be due to the feature weight settings of the SMT normalisation model or to some other minor difference in training parameters. Compared with those results, the scores achieved by our neural network tagger are very close, but still slightly lower. This corroborates Pettersson's finding that SMT normalisation is a very strong method for processing Swedish historical texts.

The other two contrastive systems reported in Table 3 are trained towards tagging modern Swedish without specific accommodations for historical text. Our character-based neural network tagger trained with a λ weight of 1.0 (at an F-score of 0.7055) seems to be slightly more robust to the unexpected historical spellings than HunPos (at 0.6632), but as expected, both models perform substantially worse in this setting.

The results of our experiments with German are in Table 4. The table includes POS tagging accuracies for the modern (NEGRA) and historical (GerManC) corpora. For better comparison with Swedish, it also includes precision, recall and F-score values for a historical verb identification task. These results were derived from the POS tagging results by considering only those word classes that would be tagged as verbs in the Swedish verb identification setup (i. e., finite, infinite and imperative forms of main, auxiliary and modal verbs, but not participles since they have a separate tag in the SUC tag set).

Unlike for Swedish, we do not see a consistent advantage from enlarging the final hidden layer **6** in the German experiments, and indeed the best overall score on the GerManC corpus is achieved with a layer **6** size of 100 in combination with a λ weight of 0.8. The development score of our best system is 2.8 percentage points above the HunPos baseline without normalisation, which already performs quite well on this task, but still 1.9 points below the baseline with SMT normalisation. On the test set, the neural system performs almost on a par with the SMT normalisation baseline; the small remaining difference corresponds to only 5 additional mistagged tokens out of 4,845. Increasing the size of layer **6** to 300 without additional regularisation results in development accuracy scores on the order of the HunPos baseline without normalisation. On the test set, these systems still outperform the unnormalised system and achieve scores that are only 0.4–1 percentage points lower than those of the comparison systems.

When we evaluate the experiments as a verb identification task, we see mixed results. Spelling normalisation, either in the form of a normalisation error term during neural network training or as a separate preprocessing step, has a clear advantage, but all normalisation-aware systems achieve fairly similar F-scores around or just under 90 %. Pitting our best-performing system against the baseline with SMT normalisation, we find that the latter has an advantage on the development set, but the former wins on the test set. Compared to the results on Swedish, the performance on German is even more similar. On the whole, we can conclude that both system types are clearly viable approaches to this task.

It is interesting to observe that the additional constraint we impose on the neural network tagger by requiring that its internal representation of historical spellings should be similar to that of modern text does not have a negative effect on its tagging performance for modern text. Indeed, both of the adapted Swedish systems in Table 3, while still about 1.5–2.5 percentage points below the performance of HunPos, achieve higher scores on the SUC test set than the tagger trained without the additional constraint. For German, tagging performance on modern text lags more behind the HunPos benchmark and the effect of adding the normalisation error is smaller, but still slightly positive. We have not studied this result in detail, but one could speculate that the normalisation error term adds a form of regularisation to the model that improves its performance on the original domain.

6 Qualitative Observations and Discussion

To gain a clearer picture of the strengths and weaknesses of our tagging models, we subjected the output of the models on the final test sets to a manual qualitative study. The study was done informally by looking through the original text, the spelling produced by the SMT normalisation step, the gold-standard annotations and the annotations generated by our own best system and by the HunPos baseline with SMT normalisation in parallel. For each language, we checked approximately 20 % of the test set data. For German, we additionally consulted the confusion matrices for the test set annotations generated by the baseline with SMT normalisation and by our best neural network tagger.

6.1 Swedish

For Swedish, we find very few qualitative differences between the baseline tagger and our neural system. By and large, both systems seem to struggle with the same difficulties, and they often make the same errors in parallel. In the Swedish verb identification task, by far the most common source of errors was a confusion between the tags for common nouns, NN, and verbs, VB. We encountered both nouns marked as verbs and vice versa, and both errors were frequent in both systems, making it difficult to draw conclusions about the properties of a specific system from these observations. Another type of error that occurred frequently in both systems was a confusion between verbs (VB) and participles (PC), which is understandable since participles are inflected verb forms and the tested systems are explicitly designed to be tolerant towards orthographical details. Other frequent sources of errors included confusions of verbs with adverbs (AB), adjectives (JJ) and proper nouns (PM). These occurred a bit more frequently in the output of the neural tagger, but they were well attested in the HunPos output as well.

One peculiarity that is specific to our neural network tagger is that it is much more likely to output the tag UO (foreign word) than HunPos (262 instances versus 11 in the test set). In general, it does this in quite reasonable ways, for instance for tagging the Latin words *pastor in* in a list of parish priests. However, since the foreign words in the SUC training corpus are mostly in English, a language scarcely attested in our early modern corpus, it sometimes overgenerates the UO tag in incorrect contexts, for instance for the word *tree* ‘three’ (modern spelling *tre*). More seriously, it occasionally seems to interpret the presence of the letter *w*, which is frequent in early modern Swedish, but missing in the modern Swedish alphabet except for its occurrence in foreign-language words, as a cue for generating the tag UO. To address these problems, we might consider augmenting the training data with sentences from relevant foreign languages, to familiarise the model with foreign words it might encounter, or with artifically generated historical spelling, to make it more robust to the expected spelling variance.

Our clear impression from the inspection of the tagging output for the historical Swedish found in the *Gender and Work* corpus is that the most important potential gains for this type of text and task are

unlikely to be realised by tweaking the implementation of the tagger, but require a more targeted approach to handle the specific differences between historical and modern language. The work in this paper, as well as the baselines we compare with, primarily addresses orthographical differences between historical and modern text. It is well known and acknowledged in the literature, however, that the diachronic differences in language development affect all parts of language, including not only orthography but also morphology, syntax, the lexicon and so forth. Pettersson (2016) provides a good overview of different linguistic properties that are relevant for historical language processing. In the case of verb identification in historical Swedish, the errors made by our systems suggest that least morphology and syntax should be considered for better results.

In terms of Swedish morphology, a very significant development that has taken place between the early modern period and now is the decline of verb inflection. In contemporary Swedish, verbs are not inflected for person. However, the complete disappearance of person inflection is relatively recent; until the first half of the 20th century, Swedish verbs had different forms for singular and plural, and in the early modern texts in our test set, we find a separate form for first person plural (as opposed to third person, second person not being attested in the sample we inspected). The tense forms of the first person plural have an ending in *-om*, which is easily confused with similar endings of other word classes by a purely orthographical approach:

> . . . och *worom* [VB, JJ] begiärandes / at klara Gudz ord måtte blifwa predikat kring om alt Riket.
> . . . and *were* desirous / that the clear word of God should be preached in all the country.

> . . . och *hördom* [NN, NN] thesligest theras predikan och disputatien som samma nya tro sagdes predika / och *funnom* [PP, VB] doch i sanningen thet rykte oredeliga fört wara. . .
> . . . and *heard* also the sermons and teachings of those who were said to preach that new faith / and *found* in reality that this rumour was spread dishonestly. . .

The three words in italics are all first person plural verbs; the POS tags in brackets were assigned to them by the neural tagger and the SMT-normalised baseline, respectively. In the first case, the HunPos tagger assigns the tag JJ because the word *worom* is incorrectly normalised to the adjective *varm* 'warm' by the spelling normaliser. In the second example, both taggers select an incorrect noun tag, presumably because they recognise *-dom* as a derivational suffix for abstract nouns (as in *visdom* 'wisdom'). The third example is incorrectly tagged as a preposition by the neural tagger, possibly because of the similarity of its ending with Swedish prepositions like *inom* 'within' or *förutom* 'except'. HunPos tagged it correctly even though it did not get transformed into a correct modern word form by the spelling normaliser and must therefore have been treated as an unknown word by the tagger.

Another morphological phenomenon that occurs very frequently in our texts is the derivational suffix *-liga* that is used to form adverbs, as in the word *oredeliga* 'dishonestly' in the previous example. In modern Swedish, the corresponding suffix is *-ligen*. In principle, this transformation is accessible to spelling normalisation, but the problem is that *-liga* could also plausibly be a plural ending of an adjective or a common noun derived from an adjective. Moreover, *-a* is the ending of the infinitive or third person plural of a verb. Accordingly, both taggers frequently assign adjective, noun or verb tags to these adverbs.

The syntax of early modern Swedish is strongly influenced by German. In particular, it is very common for subordinate clauses to have verb-final word order, as in both clauses of the second example above. In this particular example, the verbs were tagged correctly by HunPos, but the neural tagger failed to parse the clause *som samma nya tro sagdes predika* 'who were said to preach that new faith' correctly. In this indirect speech construction, the noun phrase *samma nya tro* 'the same new faith' is the object of the verb *predika* 'to preach', which in turn is governed by the passive verb *sagdes* 'were said to'. The clause could be rendered in modern Swedish as *som sades predika denna nya tro*. The neural tagger chooses to interpret the verb *predika* as the homonymous noun *predika* 'sermon', an interpretation that makes perfect sense in the light of contemporary Swedish grammar, which does not allow a direct object to precede the governing verb as early modern Swedish did.

While the two tagging approaches sometimes make different choices for individual examples, both of them are clearly affected by the problems outlined above. An interesting fact about early modern Swedish is that many of its historical features, despite having disappeared completely from the modern form of the language, are still attested in other, closely related contemporary languages. In particular, morphological features similar to those of early forms of Swedish can be found in present-day Icelandic, and modern German still exhibits some of the syntactic patterns that were common in early modern Swedish. We believe that the versatility of vector-space embeddings will make it possible to exploit resources from those languages to train models for historical forms of Swedish by integrating them in a similar way as we integrated the historical and modern data resources in this work. In this sense, the neural method has a clear advantage of flexibility over a pipeline approach with an explicit spelling normalisation stage.

6.2 German

The German test data is rather different from the Swedish test set, mostly because it is from a later period. The two texts we selected for testing in German are from the late 18th century. This is the age of authors like Goethe and Schiller, whose works had a lasting influence on the German language. While the writing style of that epoch may seem a bit archaic to a speaker of modern German, it is still perfectly readable and much closer to present-day German in terms of syntax and morphology than the texts of the *Gender and Work* corpus are to present-day Swedish.

In the German data, we can find some distinctive tagger-specific patterns. A recurring problem in the HunPos output is the incorrect assignment of an adjective tag ADJA to an attributive possessive pronoun that should be tagged PPOSAT. This invariably concerns the pronoun *unser* 'our', which historically and dialectally can have oblique forms with elided *e* such as *unsrem* (dative). These forms do not get translated into their modern standard spellings like *unserem* and are therefore not recognised by the modern tagger. The neural tagger handles these forms without any problems. HunPos also has a tendency to mix up common nouns (NN) with adjectives (ADJA or ADJD), whereas the neural tagger is more prone to confuse common nouns with proper nouns (NE).

A large class of errors that we find in the output of both taggers is the confusion of finite verbs (VVFIN, VAFIN and VMFIN) with infinite verbs (VVINF and corresponding tags for auxiliaries and modals) and, to a lesser extent, participles (VVPP etc.). The underlying problem for most of these examples is the homonymy of first and third person plural forms and infinitives. One of the texts in the test set is a homily that extensively mixes general exhortations in the form of infinitives with first person plural verbs, as in the following example:

> Es heißt nicht Werke der Barmherzigkeit deswegen *thun*, weil wir begangne Bosheiten, die wir nicht aufrichtig *bereuen*, dadurch auszulöschen … *glauben*.
>
> It does not mean to *do* acts of charity because we *believe* we can thus eliminate sins that we have committed … and do not sincerely *regret*.

Here, *thun* is an infinitive, while the following verbs are first person plural forms, but both taggers frequently confuse the two. Unfortunately, it is difficult to evaluate these examples correctly because the gold standard itself is inconsistent. The GerManC gold standard was produced semi-automatically with automatic annotation followed by manual error correction (Scheible et al., 2011). Since the infinitives and first person plural forms are homonymous and freely mixed in the text, disambiguating them is difficult for a tagger and not entirely trivial even for a human. Looking through the homily mentioned above, we quickly found more than 25 instances of incorrectly tagged verb forms that had probably escaped the manual correction pass. It is therefore unclear to what extent the gold standard can be trusted for this specific distinction in this specific text type.

7 Conclusion

In this paper, we have presented a new method for POS tagging historical texts with a character-based recurrent neural network. Our neural tagger can be trained on a combination of a modern tagged corpus and a historical corpus in original and normalised spelling. At training time, we use a two-part error

function that combines optimisation for POS tagging performance with a criterion to ensure that historical and modern spellings are represented similarly by the neural network. The trained model can then be used to process historical data directly, without explicit spelling normalisation and achieves a level of performance that is very close to that of a state-of-the-art solution with explicit SMT-based normalisation. In a manual study of the output of our own tagger and that of a baseline with explicit spelling normalisation, we have identified the most important remaining problems for the tasks under consideration. While the 18th century German texts exhibited general tagging problems that were more reminiscent of domain adaptation than peculiar to the historical nature of the texts, the older Swedish texts clearly suffer from specific problems due to language development. We suggest that our neural tagging approach opens up new ways for tackling these problems with the help of data from other, closely related languages. This is an approach that we plan to explore further in future work.

Acknowledgements

The author would like to thank Eva Pettersson for providing the data sets used in the experiments. This work was supported by the Swedish Research Council under grant 2012-916. The computations were performed on resources provided by the Swedish National Infrastructure for Computing (SNIC) at the Lunarc centre under project SNIC 2016/1-238.

References

Stanley F. Chen and Joshua Goodman. 1998. An empirical study of smoothing techniques for language modeling. Technical report, Computer Science Group, Harvard University, Cambridge (Mass.).

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha (Qatar), October. Association for 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.

Sofia Gustafson-Capková and Britt Hartmann. 2006. Manual of the Stockholm Umeå Corpus version 2.0. https://spraakbanken.gu.se/parole/Docs/SUC2.0-manual.pdf.

Péter Halácsy, András Kornai, and Csaba Oravecz. 2007. HunPos – an open source trigram tagger. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics. Companion Volume: Proceedings of the Demo and Poster Sessions*, pages 209–212, Prague (Czech Republic), June. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Annual Meeting of the Association for Computational Linguistics: Demonstration session*, pages 177–180, Prague (Czech Republic).

Wang Ling, Tiago Luís, Luís Marujo, Ramón Fernandez Astudillo, Silvio Amir, Chris Dyer, Alan W. Black, and Isabel Trancoso. 2015. Finding function in form: Compositional character models for open vocabulary word representation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1520–1530, Lisbon (Portugal), September. Association for Computational Linguistics.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo (Japan).

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on Machine Learning*, pages 1310–1318, Atlanta (Georgia, USA).

Eva Pettersson and Joakim Nivre. 2011. Automatic verb extraction from historical Swedish texts. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 87–95, Portland (Oregon, USA), June. Association for Computational Linguistics.

Eva Pettersson, Beáta Megyesi, and Jörg Tiedemann. 2013. An SMT approach to automatic annotation of historical text. In *Proceedings of the Workshop on Computational Historical Linguistics at NODALIDA*, pages 54–69, Oslo (Norway).

Eva Pettersson. 2016. *Spelling Normalisation and Linguistic Analysis of Historical Text for Information Extraction*, volume 17 of *Studia Linguistica Upsaliensia*. Acta Universitatis Upsaliensis, Uppsala.

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*, pages 124–128, Portland (Oregon, USA), June. Association for Computational Linguistics.

Mike Schuster and Kuldip K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Wojciech Skut, Brigitte Krenn, Thorsten Brants, and Hans Uszkoreit. 1997. An annotation scheme for free word order languages. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, pages 88–95, Washington (District of Columbia, USA), March. Association for Computational Linguistics.