Proceedings of 1st Workshop on Language Technologies for Historical and Ancient Languages, pages 114–118
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

JHUBC's Submission to LT4HALA EvaLatin 2020

Winston Wu, Garrett Nicolai
Johns Hopkins University, University of British Columbia
Baltimore, USA, Vancouver, Canada
wswu@jhu.edu, garrett.nicolai@ubc.ca

Abstract

We describe the JHUBC submission to the EvaLatin Shared task on lemmatization and part-of-speech tagging for Latin. We view the task as a special case of morphological inflection, and adopt and modify a state-of-the-art system from this task. We modify a hard-attentional character-based encoder-decoder to produce lemmas and POS tags with separate decoders, and to incorporate contextual tagging cues. We observe that although the contextual cues both POS tagging and lemmatization with a single encoder, the dual decoder approach fails to leverage them efficiently. While our results show that the dual decoder approach fails to encode data as successfully as the single encoder, our simple context incorporation method does lead to modest improvements. Furthermore, the implementation of student-forcing, which approximates a test-time environment during training time, is also beneficial. Error analysis reveals that the majority of the mistakes made by our system are due to a confusion of affixes across parts-of-speech.

Keywords: evalatin, morphology, encoder-decoder, lemmatization, pos-tagging

1. Introduction

In this paper, we describe our system as participants in the EvaLatin Shared Task on lemmatization and part-of-speech (POS) tagging of Latin (Sprugnoli et al., 2020). Latin represents an interesting challenge for POS taggers — unlike English, its substantial inflectional morphology leads to significant data sparsity, resulting in large numbers of out-of-vocabulary (OOV) words for type-based taggers. Additionally, its word order is much more fluid than languages like English, handicapping n-gram taggers such as HMMs that rely on language modeling to produce tag sequences.

We consider lemmatization to be a special case of morphological reinflection (Cotterell et al., 2017), which takes as input one inflected form of a word and produces another, given the desired morpho-syntactic description (MSD) of the output form. Likewise, POS-tagging is a special case of morphological tagging but with a greatly reduced tagset.

Beginning with the state-of-the-art neural morphological generator of Makarov and Clematide (2018), we make several small modifications to both its input representation and its learning algorithm to transform it from a context-free generator into a contextual tagger. These modifications are described in Section 2. We also experiment with a neural machine translation system with no modifications.

Our results indicate that out-of-the-box tools already perform at a very high level for Latin, but that small boosts in performance can be observed through simple modifications and ensembling of different learning algorithms. We discuss our results in more detail in Section 5.

2. System Description

Since 2016, SIGMORPHON has hosted a series of Shared Tasks in morphological inflection (Cotterell et al., 2016; Cotterell et al., 2017; Cotterell et al., 2018; McCarthy et al., 2019). Increasingly, the tasks have become dominated by neural encoder-decoder architectures with heavy copy-biasing. Originally borrowed from the neural machine translation (NMT) community (Bahdanau et al., 2014), the systems have converged around hard-attentional transducers over edit actions (Aharoni and Goldberg, 2017).

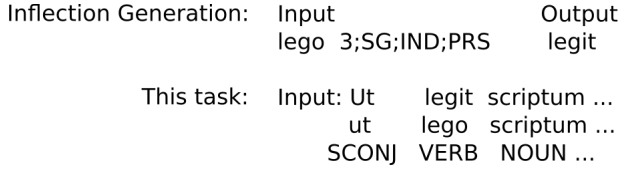

Figure 1: The difference between inflection generation and contextual tagging.

2.1. System 1: Seq-to-seq morphological analysis

As our starting point, we take the system of Makarov and Clematide (2018), the highest performing system in the 2018 shared task. Note, however, that the inflection task is quite different from this one. In the 2018 task, participants were provided with an input lemma and MSD and were required to produce an inflected word out of context. Our task is in many ways the exact opposite: given a word *in context*, we must produce a lemma and a POS tag. Figure 1 illustrates this difference.

Our first task is to convert the initial system from a generator to a lemmatizer. This step is trivial: we simply specify the MSD for every input word as "LEMMA", producing a context-free lemmatizer. We expand to a context-free morphological analyzer by appending the POS to the end of the output — where the initial system would produce "lego" given `legit LEMMA`, our system now produces "lego+VERB". We refer to this system in future sections as the single-decoder without context (SDNC).

We introduce context into the system through a modification to the MSD, appending the two previous POS tags to the MSD. Given the example sequence in Figure 1, the input for "scriptum" would be `scriptum LEMMA;-2:SCONJ;-1:VERB`. We refer to this system as the single-decoder with context (SDC).

During training, it is common to feed the gold POS tags into the system as context, but at test time, the system must rely on its own predictions and may fall prey to overfitting, as it has trouble recovering from an incorrectly-predicted tag. To help mitigate this issue, we also introduce a sys-

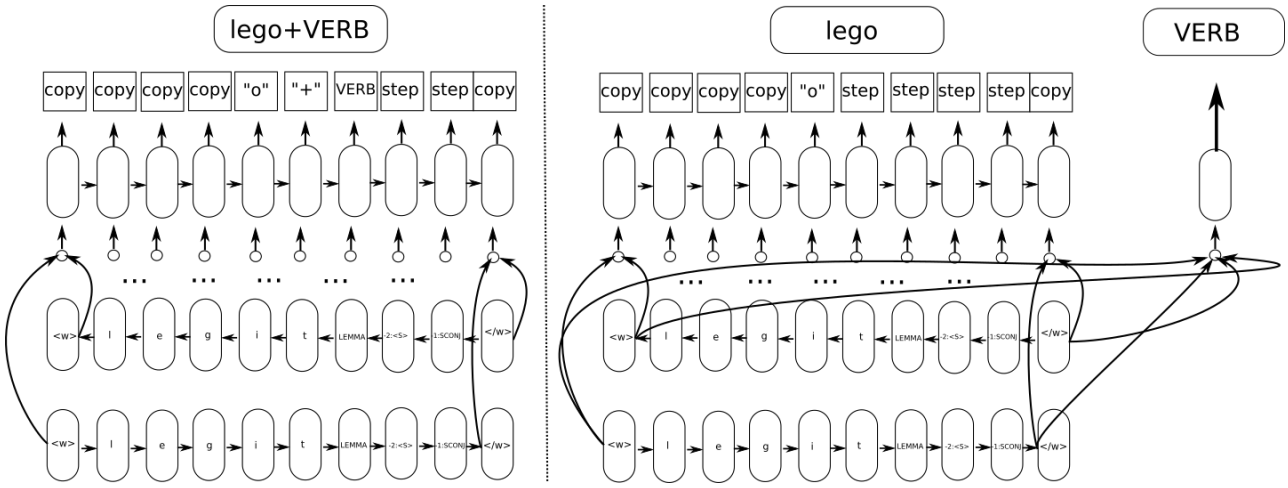

Figure 2: On the left - the single decoder architecture of Makarov et al.; On the right, the dual decoder architecture we introduce. Some connections have been removed to avoid clutter.

tem that learns via student-forcing, where the tags on the MSD are not the gold POS tags, but rather the predictions produced by the decoder. We refer to this system as the single-decoder with student forcing (SDSF).

Our most significant modification to the baseline system involves altering the architecture to produce lemmas and tags separately. By separating the decoders, we simplify the task of each decoder, allowing each decoder to specialize in its particular task. Each decoder has its own attention mechanism that allows it to focus on the parts of the input most significant to its task. The architecture is illustrated in Figure 2.

In both the single and dual decoder models, a bidirectional LSTM encoder reads in the input sequence (`legit LEMMA -2:<s> -1:<SCONJ>`) character-by-character[1]. In the single decoder, a hard attention mechanism feeds a decoder that generates edit actions (either "copy", "step", or "insert-x"), before producing the final output: `lego+VERB`. The dual decoder produces the lemma in the same way, but uses a second decoder with a global attention mechanism to produce a single POS tag.

2.2. System 2: Neural Machine Translation

Our second system submission is meant to serve as a strong baseline to compare with System 1. Treating the lemmatization and POS tagging tasks as a sequence prediction problem, we employ an off-the-shelf neural machine translation toolkit OpenNMT (Klein et al., 2017) with modifications to the data preprocessing. For both tasks, the input is the Latin word with its previous and next words in the sentence (including sentence boundary tokens). We train a SentencePiece (Kudo and Richardson, 2018) model with a vocabulary size of 8000 and apply it on both the input and output for lemmatization, and only the input for POS tagging. An example is shown in Table 1.

2.3. Ensembling

In addition to producing multiple individual systems, we ensemble each system, using a linear combination of each system's confidence scores from the decoder[2]. To aid the ensemble, we produce 10-best lists for each system, which requires a small modification to the beam search: each decoder produces a 10-best list of hypotheses, which are then combined with a linear combination of their confidence scores, with ties going to the prediction with the higher lemma score.

Input:	_cum _dolore _in f i d e l it at is
Output (lemma):	dolor
Output (POS):	NOUN

Table 1: Data format for System 2 after processing with SentencePiece.

3. Experimental setup

We train our models on a 90% balanced subset of the provided training data, reserving 10% of the sentences in each document as a validation set. We train the single- and dual-decoder models identically. The encoders and decoders consists of a single layer with 200 hidden units, an embedding size of 100 for actions and characters, and 20 for POS tags. We train with a batch size of 32, using AdaDelta, a ReLU non-linearity function, and 50% dropout. All models are trained for a maximum of 60 epochs, with patience of 10 epochs.

For the NMT system, we use the default parameters of OpenNMT, which include a 2 layer encoder and decoder with 500 hidden units and an embedding size of 500. There is no difference in architectures for the lemmatization and POS tagging tasks. We train with a batch size of 64 using Adam, with 30% dropout, with a patience of 3 epochs.

4. Results

We now present the official test results of our systems in the three sub-tasks: classical, cross-genre, and cross-time. Our official submissions correspond to the Ensemble and

[1] MSDs are atomic.

[2] An incompatibility with OpenNMT's decoder prevents us from including the NMT system in the ensemble.

NMT baseline. The classical task presents test data by the same authors as were used in training, and consists of letters, speeches, and treatises. The cross-genre task tests on the Odes of Horace, also written in classical Latin but of a different genre (lyric poetry), while the cross-time task evaluates on a treatise by St. Thomas Aquinas written in the Ecclesiastical Latin of the 13th century.

System	Setting	Lemma	POS
Single	No Context	94.32	93.38
Dual	No Context	93.94	93.20
Single	Teacher	94.36	*93.87*
Dual	Teacher	93.61	92.73
Single	Student	*94.59*	93.8
Dual	Student	93.45	92.74
Ensemble	–	**94.76**	**94.15**
NMT	–	94.22	92.98

Table 2: Test Accuracy on Classical Task

System	Setting	Lemma	POS
Single	No Context	83.98	87.00
Dual	No Context	82.47	86.51
Single	Teacher	84.67	87.53
Dual	Teacher	82.42	86.39
Single	Student	*84.74*	*87.92*
Dual	Student	82.32	86.85
Ensemble	–	**85.49**	**88.40**
NMT	–	82.69	82.93

Table 3: Test Accuracy on Cross-Genre Task

System	Setting	Lemma	POS
Single	No Context	85.38	80.32
Dual	No Context	84.87	78.5
Single	Teacher	85.77	*82.49*
Dual	Teacher	85.36	80.06
Single	Student	***85.81***	81.58
Dual	Student	84.26	78.21
Ensemble	–	85.75	80.78
NMT	–	83.76	**82.62**

Table 4: Test Accuracy on Cross-Time Task

We observe that for all three sub-tasks, the single-encoder model outperforms our dual-decoder extension, for both lemmatization and POS-Tagging. It may be that lemmatization and POS-tagging provide complementary information that benefits a joint decoder, and splitting the decoders shifts much of the joint learning to the encoder, which is not able to learn a sufficient representation to accomodate the separate decoding mechanisms.

Encouragingly, the contextual information appears to have been captured by the encoder. POS-tagging and lemmatization both benefit from knowing the POS-tag of the previous POS tags in the sentence. We provide some discussion of this phenomenon in Section 5. We also observe that lemmatization benefits slightly from a student-forcing scenario.

	NN	VB	JJ	NNP	RB	AUX
NN	13333	182	356	63	50	0
VB	105	12037	114	9	12	69
JJ	204	140	4099	91	86	0
NNP	51	3	46	2437	6	0
RB	18	3	72	10	4188	0
AUX	0	273	0	0	0	480

Table 5: POS Confusion Matrix: open classes (y=gold)

Not surprisingly, ensembling multiple systems leads to small gains over any individual system. The sole exception occurs in the Cross-Time track, which sees the ensemble struggle to surpass the individual systems. We hypothesize that the low overall accuracy on this track harms the ensemble, as models produce hypotheses more consistent with classical Latin. A system that produces a correct medieval analysis is out-voted by the other systems.

5. Discussion

We now begin a detailed discussion of the types of errors made by our systems. As a test case, we consider the classical track; the types of errors encountered here are simply exacerbated in the other tracks.

We first consider the **open classes** of words: nouns, verbs, and adjectives. These classes demonstrate prolific inflectional morphology, and account for 82.3% of the lemmatization errors of our ensembled system. Of the remaining errors, 73% of false lemmatizations concern subordinating conjunctions or pronouns. Pronouns and conjunctions are regularly tagged as adverbs — they are incorrectly tagged as such nearly 10% of the time. All told, more than 90% of our system's errors can be attributed to either the open classes, or to closed words incorrectly tagged and lemmatized as such.

Table 5 shows the errors that our system makes on the open classes. Unsurprisingly, there is much confusion between auxiliary and main verbs. Given that these are often the finite form of a verb, the results suggest that our character-based model is heavily attending to the affixes of the word for POS-tagging. Likewise, we observe this phenomenon between common nouns, proper nouns, and adjectives, which must agree grammatically and often decline similarly. Perhaps the biggest surprise comes from the confusion between verbs and nouns/adjectives, which have very different inflectional systems, but account for nearly a quarter of all open-class errors.

Closer inspection reveals that nominal-verbal confusion comes about from incorrect affix-boundary identification. For example, the noun *evocatis* should be lemmatized as *evocati*, but is instead tagged as a verb, and lemmatized as *evoco*. The *-atis* ending is a common verbal suffix denoting the 2nd person plural, and indeed, the noun *evocati* "a veteran soldier called back to service" is derived from the verb *evoco* "to call out/summon" and in dictionaries is often listed as a subentry of *evoco*. In the other direction, *meritum* should be analyzed as a conjugation of the verb *mereor*, but is instead analysed as the noun *meritum*. *-tum* is a common

	SCONJ	PRON	ADP	DT	RP	CC	NUM	INT	X
SCONJ	1235	182	80	0	43	13	0	0	0
PRON	1	4191	0	26	0	0	0	0	0
ADP	61	0	3646	0	0	0	0	0	0
DT	1	36	0	3644	0	0	95	0	0
RP	19	0	0	0	732	0	0	0	0
CC	5	15	0	0	0	3981	0	0	0
NUM	0	2	12	0	0	0	197	0	0
INT	0	0	0	0	0	0	0	24	0
X	0	0	0	0	0	0	0	0	500

Table 6: POS Confusion Matrix: closed classes (y=gold)

nominal suffix, and *meritum* is the perfect passive participle of *mereor*, which itself belongs to a rare class of deponent verbs. We see that many of our verb misclassifications occur when the verb is inflected as a participle, which in Latin resemble and decline as ordinary adjectives.

Table 6 shows similar statistics for the **closed classes**. Outside of the aforementioned errors, we see some confusion between conjuctions and pronouns and adpositions, as well as between determiners and numbers. The latter is understandable, as the word *unus* and its inflections can be both determiner or number. For the former, many subordinating conjunctions share a suffix with relative pronouns (qui, quae, quod) and interrogative pronouns (quis, quod) and their inflections. One commonly misclassified word is *quod*, which can be translated as "because" (SCONJ) or "which" (PRON) depending on the context. Several subordinating conjunctions also function as adpositions depending on context, including *cum*, which is translated as "when" (SCONJ) or "with" (ADP). Accurately determining the function and translation of these words often requires first analyzing the verb, which may appear many words later in the sentence. A larger context window may allow our systems to more accurately analyze such words.

5.1. System variants

We next investigate the types of errors that are corrected by our system variants. As the single decoder dominates the dual decoder, we will focus our investigation on its variants in the classical task. When we add context to the model, we note a 7.5% relative error reduction on POS tagging. Many of the correct POS tags occur in the closed word classes.

As hinted above, several common Latin function words such as *ante* "before", *cum* "with/when", and the inflections of *unus* are ambiguous with respect to the part of speech. *Ante*, for example, can be an adverb, meaning "ago", such as in the sentence: *multis ante mensibus in senatu dixit* ... – "He said many months *ago* in the senate ..." However, it occasionally also operates as an adposition, as in English - *volui si possem etiam ante Kalendas Ianuarias prodesse rei publicae* – "I wished, if I could, to be useful to the state even *before* the first of January." Often, *ante* is used in its adverbial form when it follows an adjective or adverb, but as an adposition when it follows a verb. Knowing the prior contextual parts of speech can help disambiguate it, such as in the test sentence: *venisti paulo ante in senatum* – "You came a little while *ago* into the senate" – where the non-contextual model predicts an adposition, but the contextual system corrects it to an adverb.

The teacher-forcing model is heavily dependent on the quality of the contextual tags. At test time, the tags produced by the system will occasionally be incorrect, cascading to incorrect lemmatization and subsequent tagging. Contrary to the POS analysis, we see that it is the open word classes that benefit most from the student-forcing. POS accuracy stays stable, but the relative lemmatization error drops by 4%. The lemmatization model learns to rely less on the previous POS tags, which may now be incorrect, and to focus more on the shape of the word; nouns and verbs, in particular, seem to benefit the most from this model. Consider the form *speret*, which is the 3rd person singular present active subjunctive of the verb *spero* "to hope". Under the teacher forcing model, it is lemmatized as "*speo", likely following the deletion rule of other verbs like "nocere → noceo". In this particular POS context, "ere → eo" is much more common than "eret → ro" — the subjunctive is simply rarer than the indicative — so the model uses the contextually conditioned transition. Under the student forcing paradigm, the model makes less use of the POS context for lemmatization, and is able to correct the error.

Finally, we take a look at the dual decoder and why it fails with respect to the single decoder model. Comparing similar systems, we note that the dual decoder and single decoder are nearest in accuracy when no context is considered, and that adding context and noise degrades the dual decoder even as it improves the single encoder. We investigate some possible reasons why in this section.

The dual decoder fails to correctly apply contextual cues much more often than the single decoder model. For example, when *quod* is used as a pronoun, it should be lemmatized as *qui*. However, when used as a conjunction, it should remain as *quod*. The single decoder correctly identifies this difference, but the dual decoder invariably lemmatizes *quod* to the majority class *qui*. It would appear that although both decoders share an encoder and an embedding space, the lemmatizing decoder disregards contextual information for lemmas.

For part-of-speech tagging, somewhat surprisingly, the dual decoder also fails to leverage contextual information, even degrading as context is fed into the system. We are at a loss to describe such a phenomenon, and the errors describe no clear pattern. It is possible that the encoder is not strong enough to embed complementary information such that separate decoders can leverage it in different ways. In the future, we will investigate increasing the representational power of the encoder in the dual-decoder model.

6. Conclusion

We have described and analyzed the JHUBC submission to the 2020 EvaLatin Shared Task on Lemmatization and POS-Tagging. Viewing the task as an extension of morphological analysis, we adapted a strong morphological generator to the tasks, with a high level of success – contextual cues can be fed to the tagger via an extended tag vocabulary, and student-forcing can help the system recover from errors at test time. Our best systems perform well across a series of related tasks, and we feel that our system provides a strong, intuitive system for future comparison.

7. Bibliographical References

Aharoni, R. and Goldberg, Y. (2017). Morphological inflection generation with hard monotonic attention. In

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2004–2015, Vancouver, Canada, July. Association for Computational Linguistics.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Cotterell, R., Kirov, C., Sylak-Glassman, J., Yarowsky, D., Eisner, J., and Hulden, M. (2016). The SIGMORPHON 2016 shared task—morphological reinflection. In *Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 10–22. Association for Computational Linguistics.

Cotterell, R., Kirov, C., Sylak-Glassman, J., Walther, G., Vylomova, E., Xia, P., Faruqui, M., Kübler, S., Yarowsky, D., Eisner, J., et al. (2017). CoNLL-SIGMORPHON 2017 shared task: Universal morphological reinflection in 52 languages. *arXiv preprint arXiv:1706.09031*.

Cotterell, R., Kirov, C., Sylak-Glassman, J., Walther, G., Vylomova, E., McCarthy, A. D., Kann, K., Mielke, S., Nicolai, G., Silfverberg, M., et al. (2018). The CoNLL–SIGMORPHON 2018 shared task: Universal morphological reinflection. *arXiv preprint arXiv:1810.07125*.

Klein, G., Kim, Y., Deng, Y., Senellart, J., and Rush, A. (2017). OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada, July. Association for Computational Linguistics.

Kudo, T. and Richardson, J. (2018). Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Makarov, P. and Clematide, S. (2018). Neural transition-based string transduction for limited-resource setting in morphology. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 83–93, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

McCarthy, A. D., Vylomova, E., Wu, S., Malaviya, C., Wolf-Sonkin, L., Nicolai, G., Kirov, C., Silfverberg, M., Mielke, S. J., Heinz, J., et al. (2019). The SIGMORPHON 2019 shared task: Morphological analysis in context and cross-lingual transfer for inflection. *arXiv preprint arXiv:1910.11493*.

Sprugnoli, R., Passarotti, M., Cecchini, F. M., and Pellegrini, M. (2020). Overview of the evalatin 2020 evaluation campaign. In Rachele Sprugnoli et al., editors, *Proceedings of the LT4HALA 2020 Workshop - 1st Workshop on Language Technologies for Historical and Ancient Languages, satellite event to the Twelfth International Conference on Language Resources and Evaluation (LREC 2020)*, Paris, France, May. European Language Resources Association (ELRA).