Lemmatizing and POS-tagging Akkadian with BabyLemmatizer and Dictionary-Based Post-Correction

Aleksi Sahala
University of Helsinki, Finland
aleksi.sahala@helsinki.fi

Tero Alstola
University of Helsinki, Finland
tero.alstola@helsinki.fi

Jonathan Valk
University of Helsinki, Finland
jonathan.valk@helsinki.fi

Krister Lindén
University of Helsinki, Finland
krister.linden@helsinki.fi

Abstract

We present BabyLemmatizer, a hybrid lemmatizer and POS-tagger for Akkadian, the language of the ancient Assyrians and Babylonians, documented from 2350 BCE to 100 CE. In our approach the text is first POS-tagged and lemmatized with TurkuNLP trained with human-verified labels, and then post-corrected with dictionary-based methods to improve the lemmatization quality. The post-correction also assigns labels with confidence scores to flag the most suspicious lemmatizations for manual validation. We demonstrate that the presented tool achieves a Lemma+POS labeling accuracy of 94%, and a lemmatization accuracy of 95% in a held-out test set. We also apply the lemmatizer to a previously unlemmatized text corpus to test it in practice.

1 Introduction

Application of computational methods to historical text corpora provides interesting opportunities for studying large-scale phenomena that are difficult to perceive through close reading of texts. This often requires careful normalization of the language, because in many past societies spelling conventions were not fully standardized, and the corpora can contain documents written in several synchronic and diachronic variants of the language. The languages can also be morphologically complex, which further complicates even such fundamental tasks as searching for all attestations of a word in the corpus.

One way to normalize historical languages is lemmatization, which labels words with their dictionary forms regardless of their morphology and spelling. In this paper, we present a lemmatizer for Akkadian, an extinct language that was widely used as a lingua franca in ancient Mesopotamia.

The motivation for this tool emerges from close co-operation between the FIN-CLARIN coordinated Language Bank of Finland and the Centre of Excellence in Near Eastern Empires, a University of Helsinki-based research project focusing on the study of the Near East in the first millennium BCE. As part of this co-operation, the Language Bank of Finland collects corpora of ancient Mesopotamian texts written in the Akkadian language in the Korp concordance service.[1] Korp offers several useful functionalities for historians, from flexible search options to generation of statistics from text metadata as well as map views and timelines (Borin et al., 2012).

At present, Korp hosts a version of the Open Richly Annotated Cuneiform Corpus (Oracc),[2] which comes with human-verified lemmatization. The next Akkadian corpus to be included in Korp is Achemenet,[3] which has not been manually lemmatized. The only Akkadian lemmatizer currently available (Tinney, 2019) requires extensive human supervision. To minimize the need for human intervention, our aim is to lemmatize the Achemenet corpus by first training the TurkuNLP's (Kanerva et al., 2021) universal lemmatizer using the available Oracc data, and then applying simple dictionary-based post-correction scripts.

Sahala developed BabyLemmatizer and wrote sections 1, 3, 4, 5, and 6. Valk and Alstola corrected the lemmatizations in manual evaluation. Valk wrote section 2 and Alstola section 2.1. Lindén was the team leader. The current version of BabyLemmatizer is available at https://github.com/asahala/BabyLemmatizer.

[1] https://korp.csc.fi/

[2] http://oracc.org/

[3] http://www.achemenet.com/

Aleksi Sahala, Tero Alstola, Jonathan Valk and Krister Lindén 2023. Lemmatizing and POS-tagging Akkadian with BabyLemmatizer and Dictionary-Based Post-Correction. *Selected papers from the CLARIN Annual Conference 2022. Ed. by Tomaž Erjavec and Maria Eskevich.* Linköping Electronic Conference Proceedings 198, pp. 111–119. DOI: https://doi.org/10.3384/ecp198011

2 The Akkadian Language

Akkadian is an extinct East Semitic language (Hasselbach-Andee, 2021). It is attested in hundreds of thousands of inscriptions primarily from modern Iraq, but also from other sites across the Middle East. The earliest evidence of Akkadian comes in the form of personal names in texts from the Early Dynastic III Period (ca. 2600–2450 BCE). The oldest exemplars of continuous Akkadian text hail from the Sargonic Period (2350–2170 BCE), when the language was adopted for official purposes by the Akkadian Empire. After this period, the language is generally attested in one of its two main dialects: Assyrian (2000–600 BCE) and Babylonian (2100 BCE–100 CE) (Kouwenberg, 2012), both of which can be divided into different stages of development. From the second millennium BCE onward, there is also a literary dialect of Akkadian known as Standard Babylonian (Hess, 2020). Although most historical speakers of Akkadian appear to have lived in modern Iraq, the language served as a scholarly and diplomatic lingua franca in the Middle East for much of the second millennium BCE (Vita, 2020). Vernacular Akkadian died out in the first millennium BCE. Nevertheless, Akkadian continued to be used as a language of scholarship into the first centuries of the Common Era (Geller, 1997).

The corpus of Akkadian texts is vast, numbering approximately 10 million published words (Streck, 2010). This number will only grow as more Akkadian texts in museum collections are published and others are recovered from the Middle Eastern soil. Making this corpus available for computational analysis offers tremendous opportunities for future research. Yet Akkadian presents serious difficulties for automatic reading. Like other Semitic languages, Akkadian morphology employs nonconcatenative root-pattern morphotactics in stem formation and concatenative morphotactics in the attachment of various grammatical affixes to the stems. For example, the verbal form *ludlul* ”let me praise (it)!” consists of the first person singular precative suffix {lu} attached to the preterite stem {dlul}, which is formed from the root *dll* of the verb *dalālu* ”to praise”. Although the morpheme boundaries are transparent in this example, various morphophonological processes often obscure the underlying structure of the word, complicating recognition of the root radicals (von Soden, 1995). These difficulties are apparent in the following derived surface forms of the verb *warû* ”to go up”: *umda”ir* ”he commanded”, *umtēr* ”I assign”, *īrama* ”he proceeded to” (von Soden, 1995).

Another layer of complexity emerges from the use of the cuneiform script to write Akkadian (Streck, 2021). The cuneiform writing system first developed toward the end of the fourth millennium BCE to represent the Sumerian language, which is unrelated to Akkadian. It was only in the 24th century BCE that cuneiform was adapted to represent Akkadian. The cuneiform script is logo-syllabic. Signs usually represent either a syllable or a logogram. But signs can also represent determinatives, grammatical markers, and phonetic complements. Determinatives mark words as belonging to categories that include male and female personal names, divine names, geographical names, the material of an object, and types of animals. Grammatical markers are attached to logograms and convey information like whether a noun is plural. Phonetic complements can be appended to logograms to suggest to the reader the intended grammatical form of the verb represented by a logogram.

In Akkadian transliteration, logograms are represented in capital letters and named after their base reading values in Sumerian rather than Akkadian. For this reason, the character level relationship between the graphemic and phonemic forms of logographic spellings is typically suppletive. Many logograms are also ambiguous and can have different readings in different contexts. For example, the Sumerian logogram IGI (depicting an eye) can indicate any form of the words *īnu* ”eye”, *pānu* ”front”, *mahru* ”before”, and *amāru* ”to see”. Because Sumerian does not have the same phonemic repertoire as Akkadian, the cuneiform script with its inherited Sumerian values does not align perfectly with the needs of the Akkadian language. The syllabic values of cuneiform signs often collapse phonemically distinct Akkadian consonants like the dentals /t/, /ṭ/, and /d/, and the velars /g/, /k/, and /q/. The cuneiform sign with the syllabic value /ig/ can, for example, also represent the syllables /ik/ and /iq/, while the sign /ud/ can also represent /ut/ and /uṭ/. Many cuneiform signs have multiple possible syllabic, logographic, and other values. The correct reading for any sign can only be determined contextually.

Akkadian has few fixed spelling conventions. The Akkadian verbal form *iddin* ”(s)he gave it” can, for instance, be spelled syllabically as *id-din, i-din, id-di-in,* or *i-di-in*. Although Akkadian is generally

Figure 1: A bilingual (Sumerian and Akkadian) clay tablet from the Neo-Babylonian period written in the cuneiform script (9 x 9.8 x 2.9 cm). The Metropolitan Museum of Art 86.11.313.

written syllabically, scribes sometimes favored the use of Sumerian logograms, especially in certain genres of text. The verbal form *iddin* ”(s)he gave it” can therefore also be rendered with logographic and logo-syllabic spellings like SUM and SUM-*in*. The multiplicity of spellings and sign readings in Akkadian pose special challenges to lemmatization and morphological analysis. The training data for any lemmatizer must enumerate the full range of possible logographic, syllabic, and other sign readings, as well as account for the breadth of different spellings.

2.1 Digital Resources

For an extinct language, Akkadian is fairly well resourced, and texts comprising about 3–4 million tokens (words) in total have been digitized.[4] However, only a fraction of all Akkadian texts exist in a digital format, and even fewer texts have been lemmatized. According to an estimate by Streck (2010), the known Akkadian texts contain up to 10 million words, which indicates that the current digital corpora represent only about one third of the total word mass. Some larger text corpora are Archibab[5] with 22,500 Akkadian texts, the Cuneiform Digital Library Initiative (CDLI)[6] with 14,000 texts, Oracc with 13,000 texts, and Achemenet with 5,000 texts.[7]

These four projects highlight the current diversity in the standards of digitizing Akkadian texts. The texts in Oracc have been linguistically annotated and can be downloaded as JSON files, and they are thus well suited for computational analysis without much further processing. The Akkadian texts in Oracc comprise about two million lemmatized tokens in total. At the other end of the spectrum, the Achemenet texts are provided as transliterations with some metadata and occasional translations, and they can only

[4] This figure is our estimate. There are no surveys that report accurate estimates, nor studies that indicate how much overlap different resources have.

[5] https://www.archibab.fr/

[6] https://cdli.mpiwg-berlin.mpg.de/

[7] There is overlap between the corpora, but the number of duplicates is difficult to estimate. A complete but somewhat outdated survey of Akkadian digital resources is given in Charpin (2014). The number of texts in Oracc was counted in February 2023, and the number of texts in Archibab, CDLI, and Achemenet was retrieved from their websites in January 2023.

be accessed as HTML files published on the website. Many texts in Archibab have been lemmatized, but it is not possible to download the annotated data. The texts in CDLI are only transliterated, but they can be easily downloaded. CDLI is also the largest database of cuneiform language metadata, containing information about 350,000 cuneiform texts or their fragments, including 88,000 written in the Akkadian language.

The limited availability and uneven geographic and chronological distribution of lemmatized texts pose problems for the computational study of the Akkadian language. The majority of texts in Oracc have been written in the Neo-Assyrian period (934–612 BCE), whereas CDLI focuses on texts from the Old-Babylonian (2003–1595 BCE) and Neo-Assyrian periods, Archibab on texts from the Old-Babylonian period, and Achemenet on texts from the Persian period (539–331 BCE). Because a large quantity of lemmatized texts is readily available only from the Neo-Assyrian period, it is currently not possible to do diachronic or cross-cultural studies of the Akkadian language with computational methods. Using the lemmatizer described in this paper, we aim to alleviate this problem by creating a large corpus of lemmatized texts from the Neo-Babylonian (in our context, 626–539 BCE) and Persian periods. For this purpose, we have acquired a corpus of 3,000 texts from Achemenet and are in the process of collecting additional transliterated text corpora from our colleagues.

3 Previous Work

Due to the previously discussed complexity of the Akkadian morphology and script, lemmatization is considered a mandatory step in making any digital corpus of Akkadian searchable or suitable for computational analysis (Maiocchi, 2019). To date, however, only Oracc provides an extensive and downloadable dataset of lemmatized Akkadian texts, totalling about two million lemmatized words. Oracc is lemmatized using a dictionary-based tool known as L2 (Tinney, 2019), which populates new texts with lemmata and part of speech (POS) tags based on a labeled glossary extracted from previously lemmatized texts. Texts are then checked manually word-by-word, filling in lemmata for out-of-vocabulary words and resolving possible ambiguities.

In addition to L2, there are also other tools for Akkadian lemmatization (Sahala, 2021). The earliest proto-type was a two-level morphology by Laura Kataja and Kimmo Koskenniemi, which was also the first attempt to morphologically analyze and lemmatize Semitic discontinuative morphology (Kataja and Koskenniemi, 1988). A more practical dictionary-based lemmatizer was developed by Simo Parpola and Robert M. Whiting for the use of the Neo-Assyrian Text Corpus Project in the late 1980s, but the source code or description of the system has not been published.[8]

A recent morphology-based lemmatizer is Bamman's finite-state morphology for Old Assyrian (Bamman, 2012), capable of lemmatizing and morphologically analyzing Old Assyrian letters from transliteration. The only finite-state morphology for Babylonian is BabyFST (Sahala et al., 2020). However, at its current state, it is not capable of disambiguating the morphological analyses or lemmatization, and it requires the input text to be normalized into phonological transcription (e.g. *inaddin* ”he gives”) instead of transliteration (e.g. SUM or *i-na-din, i-na-dì-in*), which is the standard way of rendering Akkadian texts in the Latin alphabet.

The system presented in this paper aims to improve four aspects of the existing Akkadian lemmatizers:

1. Handling out-of-vocabulary (OOV) words that are problematic for dictionary-based methods.

2. Handling ambiguity by producing exactly one analysis per word form.

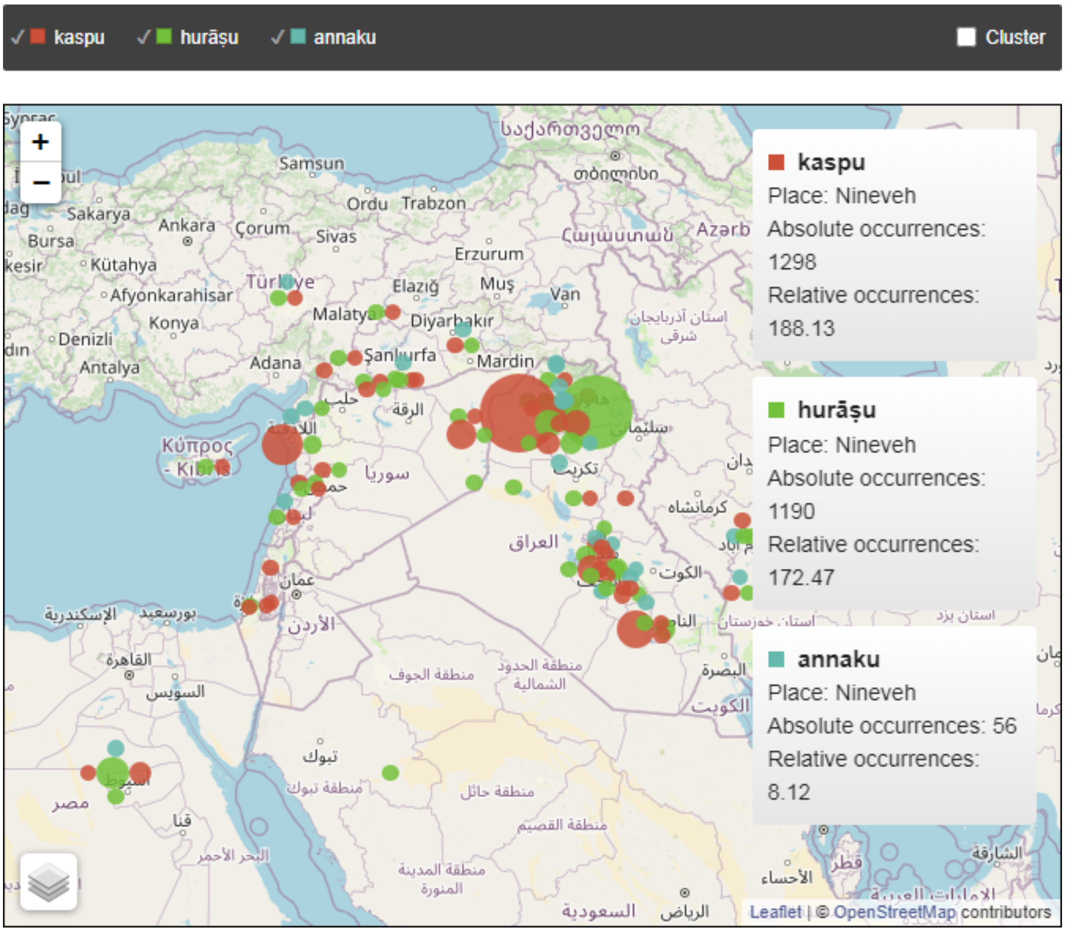
3. Providing a simple self-evaluation of the lemmatization results to minimize the need for manual processing.

4. Offering a possibility to train a specific model for the required task, such as annotation of a certain domain of texts that belong to a specific time period, genre, or dialect.

[8] Personal communication with Parpola.

3.1 Relevance to CLARIN

The Language Bank of Finland aims to improve the accessibility and usability of cuneiform corpora by hosting them in Korp. Currently Korp hosts a version of the Oracc corpora[9] providing some additional means to work with the cuneiform texts, such as statistics based on text metadata, plots of the provenance of texts on a map (see Figure 2), and search by lemma and POS-tagging by preprocessing previously unannotated texts with the BabyLemmatizer. Currently, Achemenet, a linguistically unannotated corpus of 3,000 Neo- and Late Babylonian texts (from the mid-first millennium BCE), is being added to Korp.



🟥 **kaspu**
Place: Nineveh
Absolute occurrences: 1298
Relative occurrences: 188.13

🟩 **hurāṣu**
Place: Nineveh
Absolute occurrences: 1190
Relative occurrences: 172.47

🟩 **annaku**
Place: Nineveh
Absolute occurrences: 56
Relative occurrences: 8.12

Figure 2: A screenshot of Korp's map feature showing provenances of texts where Akkadian words *kaspu* ”silver”, *hurāṣu* ”gold” and *annaku* ”tin” are mentioned in Oracc.

4 Description of BabyLemmatizer

BabyLemmatizer[10] is a hybrid lemmatizer that utilizes predictive features of neural networks to handle out-of-vocabulary words and dictionary-based lemmatization for previously known word forms. The neural lemmatization is based on a character level representation of the Akkadian transliteration, whereas the dictionary-based method is based on word form tokens. During the lemmatization process, the input stream follows standard Oracc transliteration guidelines[11] with the exception of converting determinatives into capital letters as if they were logograms. This conversion aims to reinforce the logographic nature of determinatives and to guide the neural network to not confuse them with similar signs that represent phonetic sequences. For example, capitalizing the determinative of geographical locations {ki}

[9] http://urn.fi/urn:nbn:fi:lb-2022031705
[10] https://github.com/asahala/BabyLemmatizer
[11] http://oracc.museum.upenn.edu/doc/help/languages/akkadian/akkadianstylesheet/

into {KI}, associates it more closely with the logogram KI ”land” than the syllabic reading *ki* of this same sign.

Currently the backbone of our tool is the Turku Neural Parser Pipeline (TurkuNLP) (Kanerva et al., 2018), a state-of-the-art neural lemmatizer built around Dozat’s POS-tagger and parser (Dozat et al., 2017). We first train a model for TurkuNLP using Oracc data to provide input text with raw lemmatization and POS-tagging, and then apply this model on the source text and run dictionary-based post-corrections on the result to improve the lemmatization accuracy. In our system, the post-correction involves three distinct steps:

The first step overrides all predictions for in-vocabulary words to minimize the effect of mislearned character level relationships between spellings and their lemmata. We calculate the degree of ambiguity for all lemmatizations in the training data and create a *master glossary* of word forms that have a low degree of ambiguity, and use this dictionary to override all lemmatizations of in-vocabulary words. The degree of ambiguity for a word form is considered to be low, if any lemma+POS label constitutes over 60% of all the labels assigned to it in the training data. At this step, we leave ambiguous words untouched.

The second step aims to assign correct lemmata to words that are known to be ambiguous based on the training data, especially those written with logograms. We calculate co-occurrence probabilities for lemmata and their adjacent POS-tags in the training data, and then assign the most likely lemmata for all word forms in the text based on their POS contexts. We rely on POS-tags instead of surrounding lemmata due to the high POS-tagging accuracy of the TurkuNLP’s tagger (ca. 97% for Akkadian), and because the Akkadian corpus is fairly small, which makes using adjacent word-forms or lemmata infeasible. In addition to determining the most likely lemmatization for an ambiguous word form, this step also allows us to reconfirm that our close-to-unambiguous lemmata determined in the previous step are probably correct. This information is later used to score the confidence of our lemmatizations, explained in the next subsection.

Finally, we apply various other post-corrections to the data, such as removing the lemmatization from numbers and words that occur in badly damaged sections of the tablet. These parts are easy to detect, because in the Akkadian transliteration unreadable signs are indicated with the symbol x, as in *x-x-in-nu*. This is done to make the lemmatizations more consistent with Oracc conventions, which generally leave too badly damaged and thus unrecognizable word forms unlemmatized. We also heuristically detect some obvious lemmatization errors, such as verbs that show impossible or very unlikely dictionary form patterns. An example of this would be cases in which a word has been labeled as a verb in the POS-tagging process, but lemmatized as if it were a noun. This is possible, because many Akkadian nouns and adjectives derive from verbs. Nonetheless, these can only be flagged for human editors, rather than fixed automatically especially for word forms that do not exist in the training data.

4.1 Confidence Scoring

All lemmata are assigned with a confidence score based on the post-correction steps they have passed. This aims to help Assyriologists in finding the most likely incorrect lemmata from the text and maximize the efficiency of manual lemmatization corrections. OOV logograms and logo-syllabic spellings receive the lowest class of 0 as the relationship between logographic spellings and their lemmatizations is generally suppletive. Syllabic spellings of OOV words receive a confidence score of 1, as they are possible to predict but they cannot be verified by the post-correction process. Of the highest confidence classes, the score of 3 is given to all words that have passed the first post-correction step and thus have been considered to be unambiguous or close to unambiguous. This score is raised to 4, if the lemmatization also passes the second post-correction step and is thus verified to exist in a previously seen part-of-speech context. Remaining in-vocabulary words, namely those with high ambiguity that cannot be resolved by their POS context, are given a confidence score of 2.

5 Evaluation

For evaluation, we train ten models for the first millennium BCE Babylonian texts from Oracc comprising ca. 500,000 Akkadian words in total.[12] The texts represent a wide variety of genres, ranging from astronomical diaries and sign lists to royal inscriptions and legal texts. Everyday texts such as legal transactions are written in the Neo-Babylonian (Late Babylonian) dialect of Akkadian, while literature and royal inscriptions are written in Standard Babylonian, an archaic literary variety of the language (Hess, 2020). We use a text-wise 80/10/10 train/dev/test split and estimate the model's accuracy against two baseline models by using 10-fold cross-validation.[13] As the Oracc data is divided into several subprojects that contain texts that belong to similar genres, we do not shuffle the texts before building our data, but instead pick our 80/10/10 splits in order to ensure that all the sets have a somewhat balanced distribution of different text genres.

Our first **baseline** model is a dictionary-based lemmatizer and POS-tagger that labels the word forms in our test set with their most common lemmata and POS-tags seen in the training data. To measure the effect of our post-corrections, we use **TurkuNLP** without any post-correction scripts as the second baseline model. The results are presented in Table 1.

Model	**Lemma**	**POS**	**Lemma+POS**
Baseline	84.42 ±0.33	88.83 ±0.31	82.71 ±0.34
TurkuNLP	86.19 ±1.32	**97.32** ±0.10	85.31 ±1.31
BabyLemmatizer	**94.94** ±0.17	**97.32** ±0.10	**94.03** ±0.35

Table 1: Average accuracy (%) based on 10-fold cross-validation.

In addition to validating the lemmatization and POS-tagging accuracy, we examined the distribution of the confidence classes in our evaluation set. Table 2 presents lemmatization accuracies in different confidence classes, as well as the proportion of lemmata that are assigned to each confidence class in our evaluation setting.

Confidence score	**0**	**1**	**2**	**3**	**4**
Accuracy	30.66%	56.71%	69.57%	96.25%	98.40%
Lemma-%	0.86%	3.87%	0.49%	52.10%	42.67%

Table 2: Confidence score distribution after all post-correction steps.

The above confidence score distribution also reveals BabyLemmatizer's capability to handle out-of-vocabulary words. As the confidence scores of 0 and 1 are assigned for out-of-vocabulary syllabic spellings and logo-syllabic spellings respectively, we can observe that the system is able to assign correct lemmata for 30.66% of the OOV logo-syllabic spellings and for 56.71% of the OOV syllabic spellings.

5.1 Manual Evaluation

To test our lemmatizer in practice, we apply it to a sub-corpus of Achemenet comprising 107,778 words.[14] These are primarily Babylonian legal and administrative texts from the Persian period. Although this sub-corpus has a different genre and time period distribution than our previous test sets, the texts were not completely out-of-domain, since our training data included 371 legal documents from the Hellenistic period (late first millennium BCE) comprising 107,403 words, and 87 texts from the Persian period comprising 5,893 words. For administrative texts, our training data comprised only 34 texts totaling 1,734 words. We use a model trained with the same Oracc data and train/dev/test split as in our evaluation setting described above, with an added glossary of Akkadian personal names from the

[12] Every model uses the same hyperparameters.

[13] In this experiment we use the default network architectures for training TurkuNLP's lemmatizer and tagger.

[14] The texts were provided to us by the Achemenet project in December 2020.

Prosobab database (Waerzeggers and Groß et al., 2019). We then generate glossaries of the most common words that were assigned with the two lowest confidence classes and manually correct lemmata and POS-tags for word forms in the glossary file that have a frequency of >3 (for class 0) and >5 (for class 1) in the data. For two text groups in Achemenet (CT 55 and Bel-remanni) both of these frequencies were >2. There were 315 unique corrected word forms, comprising 3.87% of the unique word forms covering 4.77% (5,037) of the 107,778 words in the sub-corpus.

To measure the accuracy of the lemmatizer and the effect of our manual corrections, we randomly select texts from our lemmatization results amounting to ca. 1,000 tokens for manual evaluation. We first evaluate the initial lemmatization without any manual corrections to the glossaries as a baseline. Then we apply our corrections to the lemmatization results in two ways: first, as a part of our *master glossary* of unambiguous lemmata (used in step 1 of post-correction), and second, by adding our manual corrections to the training data for TurkuNLP to see how much the system can learn from the corrections. The training data is added by first lemmatizing the text with a corrected master glossary and then replacing all words with the lowest two confidence scores with underscores to prevent the neural network from learning likely erroneous lemmatizations. The results are shown in Table 3.

	Lemma	**POS**	**Lemma+POS**
Baseline	93.0%	94.6%	90.2%
Glossary Override	96.2%	96.0%	93.8%
Retrained NN	**96.6%**	**96.1%**	**94.5%**

Table 3: Improvement in accuracy after corrections.

As can be seen from Table 3, our Lemma+POS labeling accuracy improves 4.3% when manually correcting only 3.87% of the unique word forms. The final results can be considered satisfactory for our current needs, which are to make the corpus searchable in Korp and to use it for lexical analysis.

6 Conclusions

We presented a hybrid lemmatizer and POS-tagger for Akkadian, and demonstrated an increase of ca. 10% in Lemma+POS labeling accuracy compared with our baseline models. We also tested the lemmatizer on a previously unlemmatized Akkadian corpus with a different chronological and genre distribution than our training data. This test demonstrated that the system can reach a Lemma+POS labeling accuracy close to 95% after minor manual corrections.

As our future work, we plan to try a different input format for training the neural networks. Both the POS-tagger and the lemmatizer used in TurkuNLP split input words into sequences of characters, which is relevant for languages using alphabetic writing systems, but not necessarily for languages that use logo-syllabic writing. Our preliminary tests show that by splitting syllabic signs into sequences of characters but preserving logograms as tokens yields slightly more reliable results for both in-vocabulary and out-of-vocabulary words. We have also experimented with providing more context data (immediately adjacent POS-tags) for the lemmatizer already when the neural network is being trained, which also seems to improve the lemmatization accuracy.

We plan to integrate the morphological analyzer BabyFST to BabyLemmatizer, as BabyLemmatizer can be used to disambiguate morphological annotations at least on the lemmatization and POS-tagging level.

We also plan on training BabyLemmatizer for other cuneiform languages such as Sumerian and Urartian, as well as some Akkadian dialects (Assyrian) and sister languages such as Eblaite.

Acknowledgements

We gratefully acknowledge that the research for this article has been funded by FIN-CLARIN and the Academy of Finland (decision numbers 330727, 336673, and 341798). We thank the Open Richly Annotated Cuneiform Corpus (Oracc) for their efforts in making linguistically annotated cuneiform texts available online and the Achemenet project for providing us with their corpus of transliterated Babylonian texts. We thank Tommi Jauhiainen for providing feedback on a draft version of the article.

References

David Bamman. 2012. *NLP Lab Report: Akkadian-morph-analyzer, https://github.com/dbamman/akkadian-morph-analyzer*.

Lars Borin, Markus Forsberg, and Johan Roxendal. 2012. Korp – the corpus infrastructure of Språkbanken. In *Proceedings of LREC 2012*, pages 474–478, Istanbul. ELRA.

Dominique Charpin. 2014. Ressources Assyriologiques sur Internet. In *Bibliotheca Orientalis 71.*, October.

Timothy Dozat, Peng Qi, and Christopher D Manning. 2017. Stanford's graph-based neural dependency parser at the conll 2017 shared task. In *Proceedings of the CoNLL 2017 shared task: Multilingual parsing from raw text to universal dependencies*, pages 20–30.

Mark Geller. 1997. The First Wedge. In *Zeitschrift für Assyriologie 87*, pages 43–95.

Rebecca Hasselbach-Andee. 2021. Classification of Akkadian within the Semitic Family. In J. P. Vita, editor, *History of the Akkadian Language*, pages 119–146. Brill.

Christian W. Hess. 2020. *Standard Babylonian*. Wiley Sons, Ltd.

Jenna Kanerva, Filip Ginter, Niko Miekka, Akseli Leino, and Tapio Salakoski. 2018. Turku Neural Parser Pipeline: An End-to-End System for the CoNLL 2018 Shared Task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 133–142, Brussels, Belgium, October. Association for Computational Linguistics.

Jenna Kanerva, Filip Ginter, and Tapio Salakoski. 2021. Universal lemmatizer: A sequence-to-sequence model for lemmatizing universal dependencies treebanks. *Natural Language Engineering*, 27(5):545–574.

Laura Kataja and Kimmo Koskenniemi. 1988. Finite-state Description of Semitic Morphology: A Case Study of Ancient Accadian. In *Coling Budapest 1988 Volume 1: International Conference on Computational Linguistics*.

Bert Kouwenberg. 2012. Akkadian in General. In Weninger S., editor, *The Semitic Languages: An International Handbook*, pages 330–339. De Gruyter Mouton.

Massimo Maiocchi. 2019. Thoughts on Ancient Textual Sources in Their Current Digital Embodiments]. In S. Valentini and G. Guarducci, editors, *Between Syria and the Highlands: Studies in Honor of Giorgio Buccellati and Marilyn Kelly-Buccellati*, pages 262–268. CAMNES.

Aleksi Sahala, Miikka Silfverberg, Antti Arppe, and Krister Lindén. 2020. BabyFST-towards a finite-state based computational model of ancient babylonian. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3886–3894.

Aleksi Sahala. 2021. *Contributions to Computational Assyriology (PhD Thesis)*. University of Helsinki.

Michael P. Streck. 2010. Großes Fach Altorientalistik: Der Umfang des keilschriftlichen Textkorpus. In *Mitteilungen der Deutschen Orient-Gesellschaft zu Berlin 142, pages 35-58*, page 35–58.

Michael P. Streck. 2021. Akkadian and Cuneiform. In History of the Akkadian Language. In J. P. Vita, editor, *History of the Akkadian Language*, pages 66–74. Brill.

Steve Tinney. 2019. *L2: How it Works, http://oracc.org/doc/help/lemmatising/howl2works*.

Juan-Pablo Vita. 2020. *History of the Akkadian Language (2 vols)*. Brill.

Wolfram von Soden. 1995. *Grundriss der akkadischen Grammatik (3rd edition)*. Pontifical Biblical Institute, Rome.

Caroline Waerzeggers and Melanie Groß et al. 2019. *Prosobab: Prosopography of Babylonia (c. 620-330 BCE), https://prosobab.leidenuniv.nl*.