Evaluating a Modern Polish POS-Tagger on Historical Data

Maria Irena Szawerna
Master in Language Technology program
University of Gothenburg
gusszawma@student.gu.se

Abstract

The goal of this course paper is to evaluate the performance of a machine learning-based POS-tagger that was trained on modern Polish on historical data. While the issue of the reliability of utilizing POS-taggers on historical data has been previously discussed, most of the research focuses on languages the grammar of which differs from Polish substantially, meaning that their results need not be fully applicable in this case. The evaluation is conducted on a set of over 3000 manually annotated tokens from a piece of historical Polish writing (1899), identifying the problematic tokens and their context for subsequent qualitative analysis.

1 Introduction

While corpora, tools, and other resources exist for many modern languages, the scarcity of data, especially annotated data, plagues quantitative research into less prominent, non-standard, or historical languages or varieties. One of the approaches in computational linguistics has been trying to utilize or adapt existing tools for those underrepresented resources. Significant work has gone into the evaluation of a variety of different part-of-speech taggers on historical data and into exploring the procedures that can be undertaken in order to improve the performance of the aforementioned taggers.

In one of the seminal papers within this field, Rayson et al. (2007) tested the performance of a combined rule-based and statistical tagger on a selection of Early Modern English texts. The authors selected around 1000 tokens from each of the 8 texts that they utilized, and compared the performance of the tagger, with or without pre-processing of the text, to a manually post-edited gold standard. The authors report that the accuracy of the annotation fell from the benchmark of 96% down to between 82% and 88.5%. However, if spelling variation was accounted for, that accuracy could be increased to 89% and 93.2%, respectively. The authors acknowledge that the remaining difference between the performance on modern and historical data may result from differences in grammar, which can also complicate the use of a specific tagset by the tagger.

In a similar study, Scheible et al. (2011) investigated how an "off-the-shelf" German part-of-speech tagger called TreeTagger performed on Early Modern German data. They used more than 50,000 annotated tokens which represented a variety of genres and time periods, with around 2000 tokens for each category. The spelling was normalized, and the tagset was adapted to match the tagger. The authors report a 69.6% accuracy on non-normalized data, with an increase of performance to 79.7% when the spelling was normalized. They also note that not all of the normalization and annotation had a positive effect on the performance of the tagger.

The topic of POS-tagging historical German is continued by Bollmann (2013), who employs automatic normalization and explores handling the peculiarities of particular texts as a part of pre-processing the texts for automatic part-of-speech tagging. While for each text used in the study around 2000-5000 tokens were chosen, only 1000 from each are used for evaluation, while parts of the rest are used for training the pre-processing tools. The author notes that different texts perform differently, and their age seems to play a role in how well the tagger performed. The outliers in terms of the performance in this study are baseline 23.05% and 81.83% after normalization, and 83.41% and 95.56% after normalization, with

a variety of values in between. It appears that the more data the normalization tool has to learn from, the better it performs. Modernizing punctuation appears to be more effective than removing it. A more qualitative analysis reveals that extinct or highly uncommon word-forms, which cannot be easily normalized, are still problematic for the tagger, as well as a simple mismatch between the vocabulary domains between the training data and the evaluation set.

Approaching the topic from a slightly different angle, Hupkes and Bod (2016) explore the existing methods for part-of-speech tagging historical Dutch and possibilities for improvement. As their test data, the authors use slightly less than 3000 tokens from two corpora, that have been manually annotated with tags. For one of the texts, they also have a tagged modern translation of the appropriate sentences. The two taggers used in the study are MBT, a Markov-based model that was trained on a contemporary Dutch corpus, and Trigrams'n'Tags, a trigram-based tagger which has some methods for handling unknown tokens. The MBT tagger has a 96% accuracy on modern data, but only 60% on historical. From the analysis of the results, misspelled closed-class words and irregular capitalization are causing much of the confusion. Rule-based normalization increases the performance to 73-74%, and manual respelling shows even better results, with 82-89%, depending on the text. Replacing words using a pre-learned dictionary based on a parallel corpus shows a similar increase: 80-90% when only the unknown tokens are replaced, 78-92% when all tokens are replaced; the choice which tokens to replace can result in an increase or a decrease in performance depending on the source text. The authors also find it beneficial to retrain taggers using historical data, even when the annotations on that historical data are generated semi-automatically via word alignments with modern data.

While many of the papers focus on more recent - though still historical - variations of the language, the older ones are not left unexplored. In their paper, Adesam and Bouma (2016) focus on part-of-speech and morphological tagging of Old Swedish, as well as what methods can be used to handle the sparse and unusual data. The authors base their investigation on around 20,000 tokens from various Old Swedish texts, which were manually tagged and pre-processed to an extent. One issue that prevents fully automatized preprocessing is the difficulty in recognizing compounds when their constituent words are spelled apart. The tagger that the study utilizes is Marmot, a conditional random field tagger framework which is capable of handling the large number of tags that morphological tagging requires. The model reaches high performance (91.5% to 94.2%, depending on the training regime); however, when testing on other texts that accuracy drops down to 45% for POS-tags and 30% for morphological tags. The authors also investigate two methods of counteracting spelling variation: including the lemma of the word in question as a feature, and utilizing a set of simple rewriting rules to unify the spelling. Both of these approaches can be helpful, but it depends on the text that is being processed. The best final results of automatic tagging are 70% for POS-tags and 50% for the morphological ones.

Another issue that can impede part-of-speech tagging is raised by Yang and Eisenstein (2016) in their paper on tagging historical English: namely, meaning changes. The authors explore methods for handling this phenomenon, such as normalization and domain adaptation. The latter approach has several implementations, such as words being replaced by word embeddings, utilizing Structural Correspondence Learning (SCL), or Feature Embeddings (FeMa). The authors show that the latter is the most effective method, and that it can be complemented by old-fashioned spelling normalization.

What is clear from the prior research is that taggers trained on contemporary data oftentimes struggle with historical texts, for a variety of reasons, including, but not limited to, out-of-vocabulary items, variation in spelling, capitalization, and punctuation, as well as differences in morphology and syntax and semantic shifts. Large performance improvements can be observed when relatively simple pre-processing methods are used. Nevertheless, all of the previously discussed studies focus on languages from the Germanic family, which, in comparison with Slavic languages, have relatively rigid word order and relatively simple grammar[1], as well as different trends in word-formation. Given these differences, part-of-speech taggers may struggle with other problems, such as

[1]That is not to say that the grammar of those languages is simplistic - the grammars of these languages simply differ from each other substantially, meaning that certain tools or methods may need to be adapted to these conditions.

misreading a non-standard or archaic inflectional ending to signify a different part of speech. The goal of this project is to evaluate the performance of a maximum entropy model-based POS tagger for Polish that has been trained on contemporary Polish from the Universal Dependencies corpora on late 19th century memoirs from southeastern Poland and to identify the problematic tokens and whatever systematic issues about the nature of historical Polish these may reveal. Given that the text is not that old, the performance is expected to be around 70-80% accuracy, based on Rayson et al. (2007) and Scheible et al. (2011), and the problematic tokens are expected to fit within one of the aforementioned categories, as the text is not preprocessed.

In Section 2, the materials and methods used in the project are discussed in more detail, including potential issues that have been encountered. In Section 3, the results of the study are presented, and subsequently discussed in Section 4. Finally, Section 5 provides the conclusions and suggestions for future work on the topic.

2 Materials and Methods

2.1 The Part-of-Speech Tagger

The tagger used in this project is a maximum entropy-based tagger hosted by the University of Sheffield on their GATE Cloud platform. The tagger is available online and can be accessed using an API. It has limitations on how much data can be processed by it at once, but for this project the free quota was sufficient. The tagger has been trained on all Universal Dependencies corpora for Polish, with the exception of test corpora. It has been evaluated to perform with a 94.56% accuracy on the test set. The annotations that the tagger assigns to the input are rather general, as it only outputs the universal POS-tags, such as NOUN or ADP (The University of Sheffield, n.d.).

Universal Dependencies contains three corpora for Polish: LFG, a corpus of 130,000 tokens, with sources such as fiction, nonfiction, news, social, and spoken language; PUD, the parallel treebank of 18,000 tokens, containing news and nonfiction data, and PDB, a corpus of 350,000 tokens, containing news, fiction, and nonfiction (Universal Dependencies, n.d.). Taking the structure of the corpora into account, it is clear that the tagger was trained on a variety of genres - it is unclear, though, if any of the source texts fully matched the literary genre of the test data in this project. In addition, while it is not specified, the assumption is that the data that the tagger was trained on represents overwhelmingly contemporary and rather standard Polish: the sources of the data including newspapers and other writing that would be edited, as well as there being separate categories for older forms of languages (Old French, Old Church Slavonic, etc.).

2.2 Test Data

The data used for testing the tagger in this project comes from the memoirs of Juliusz Czermiński, who lived in the 19th century in the area of nowadays Eastern Poland or Western Ukraine. The original manuscript was composed in 1889, and later copied on a typewriter. The location of the original manuscript is unknown as of now. Recently, the typewriter copy has been re-typed into a .docx file by a descendant of the author, and has been made available for this project. In its entirety, the document consists of over 37,000 tokens, as counted using the Word Count option in Microsoft Word. Out of those, 3270 tokens, corresponding to 114 sentences, were selected and manually annotated using the Universal Dependencies' universal part of speech tags, following the guidelines as outlined on the UD website (Universal Dependencies, n.d.).

In the case of some uncertainty about the appropriate tag for a given token, Polish dictionaries, UD treebanks, and the UD website were consulted. In some cases words had to be interpreted in context in order for the best matching tag to be assigned, as they can play different roles in sentences given the same form. For example, "jeden" literally means "one," so it should be assigned the NUM tag as a cardinal number. However, in some contexts it can be used instead of "pierwszy" (first) as an ordinal number, and consequently it can have the ADJ tag assigned instead.

Naturally, there still exists the issue of the accuracy of the re-typed text in comparison with the original manuscript. Since the original is inaccessible, no corrections were made to elements that seemed to be mistyped as a result of the copying and not spelling variation present in the original manuscript. Such an editorial decision would perhaps be possible if a more comprehensive description of the spelling variation in the text existed, but in this case a choice was made to pre-

serve almost all the tokens the way they were in the .docx file. The exceptions included what UD calls "mobile inflection", or a certain type of participle (optionally) fused with an auxiliary - those are expected to be split into their constituent parts. In addition, due to how the tagger parses the text that is given to it, "100.000" had to be changed to "100000", as otherwise it would be split by that full stop (Universal Dependencies, n.d.). On the other hand, some of the words or phrases that nowadays would have been spelled as separate words seem to have been written together by the author, or close enough to be interpreted as such. Those have been left intact, unlike the aforementioned participle case. Nevertheless, it is important to acknowledge that this text is not the original and certain misspellings may originate from the transcription, not the author.

2.3 Python and Jupyter Notebook

Both the test data and the tagger were accessed using Python code within a Jupyter Notebook file, which allowed for an instant and easy access to the results and which simplified the development of the code by allowing only parts of it to be executed as desired. The code makes use of a number of libraries to enable or simplify certain procedures. It can be accessed in the repository, which is referenced at the end of the paper.

The notebook is structured in the following way: first, the necessary libraries are imported and certain variables are set; then, functions and classes specific to the problem are defined; finally, the code is utilized to obtain results, both for original tokens and all-lowercase tokens. The procedure of evaluating the tagger consists of opening the annotated text and retrieving its contents, extracting the original tokens and gold standard annotations, obtaining the tagger annotations and confidences, calculating a number of evaluation measures, including accuracy, Matthew's Correlation Coefficient, per-class precision and recall, generating a visualization of a confusion matrix, and finally constructing a DataFrame containing the mislabelled tokens together with other useful data for a later qualitative analysis.

2.4 Error Analysis

Finally, the results of the evaluation conducted in Jupyter Notebook were subjected to qualitative analysis, the goal of which was to determine what tags and tokens were the most problematic in the text, and whether any pattern could be noticed in that, hinting to some systematic issue that perhaps could be solved using appropriate pre-processing of the text. As mentioned in the introduction, the accuracy is expected to be around 70-80%. None of the papers used MMC, so no prediction is made about that measure, although it should not be far from accuracy. Given that spelling variation within closed class words can be very problematic for taggers, some issues are expected in that field, but other categories can also be influenced by factors such as out-of-vocabulary tokens. The qualitative analysis is expected to reveal to what extent spelling variation and other factors influenced mislabelled tokens - which seems to be a major factor for other languages. Aside from that, other issues may be identified at that stage. Replacing any uppercase characters with their lowercase counterparts is expected to improve the classification of any words that have been capitalized for unknown reasons, but should not have been; however, it is likely to impede the recognition of proper nouns, if capitalization is a feature that is being taken into account by the tagger.

3 Results

3.1 Quantitative Evaluation

The evaluation pipeline was run twice, on data with original capitalization and completely lowercase data. As can be seen in Table 1, the achieved accuracy slightly exceeds the expectations of 70-80%, which may be a consequence of relatively minor differences between modern Polish and the language of the test set, or a testament to how good the performance of the tagger is when it comes to unfamiliar tokens, or a combination of both. The all-lowercase data performed worse, indicating that capitalization does carry some meaning that is used by the tagger to determine the class of the word. The noticeable difference between Matthew's Correlation Coefficient and the accuracy indicates that some more numerous categories are inflating the accuracy. When corrected for the prevalence of particular labels, the performance drops, but not by much.

Data capitalization	Accuracy	MCC
Original capitalization	84.10%	81.96%
All lowercase	80.58%	77.95%

Table 1: Accuracy and MCC per trial.

POS tag	Raw freq.	Relative freq.
ADJ	238	7.27%
ADP	340	10.40%
ADV	85	2.60%
AUX	67	2.05%
CCONJ	164	5.02%
DET	152	4.65%
NOUN	818	25.02%
NUM	43	1.32%
PART	62	1.90%
PRON	143	4.37%
PROPN	217	6.64%
PUNCT	437	13.36%
SCONJ	69	2.11%
VERB	397	12.14%
X	38	1.16%

Table 2: Raw and relative frequencies per POS tag in the text.

POS tag	Capitalized	Lowercase
ADJ	**61.84%**	52.84%
ADP	95.65%	**95.94%**
ADV	**72.29%**	70.11%
AUX	**76.92%**	73.91%
CCONJ	**97.52%**	97.50%
DET	98.46%	**98.67%**
NOUN	**84.94%**	73.23%
NUM	**73.68%**	69.44%
PART	82.22%	**82.61%**
PRON	83.55%	**83.75%**
PROPN	**66.23%**	0.00%
PUNCT	100.00%	100.00%
SCONJ	**80.43%**	77.50%
VERB	**84.87%**	82.91%
X	**64.52%**	64.00%

Table 3: Precision per POS tag per trial.

One way to determine what labels the tagger was good at assigning and where it failed is looking at the precision and recall scores. Those are displayed per category and per type of data in Tables 3 and 4, with the highest score per tag in bold. Precision, as a measure of how many positives were true out of all the positives in a category, shows us what part of all the tokens with a certain label assigned to them actually should have had that label assigned. On the other hand, recall measures how many elements out of all that originally had a given label were correctly predicted. What is worth taking into the account is the relative frequency of the particular POS tags in the original data, as per Table 2.

In terms of precision, the highest scores (most pure classes) are associated with closed-class words, such as adpositions, coordinating conjunctions, determiners, pronouns, punctuation, but also, on a level comparable to pronouns, with frequent open-class words, such as nouns and verbs. The problematic parts appear to be adjectives, adverbs, auxiliaries, numerals, and proper nouns. This means that those were the classes that most of the mislabelled tokens ended up in. What is surprising, is the relatively low precision on adjectives and adverbs, which should have rather salient morphological markings. Perhaps, though, it was the variation in spelling or word order of the rest of the elements of the sentence that influenced this result. Proper nouns are also an interesting case, with a relatively high number of tokens being falsely categorized as such. However, in the all lowercase version of the data, no proper nouns were predicted correctly, implying that the tagger relies on capitalization for the detection of this category. Therefore, any capitalized words may have been misclassified as proper nouns. Overall, precision is better in the version of the test data where the original capitalization was retained.

POS tag	Capitalized	Lowercase
ADJ	**78.99%**	78.15%
ADP	97.06%	**97.35%**
ADV	70.59%	**71.76%**
AUX	74.63%	**76.12%**
CCONJ	**95.73%**	95.12%
DET	42.11%	**48.68%**
NOUN	79.95%	**89.98%**
NUM	**65.12%**	58.14%
PART	59.68%	**61.29%**
PRON	88.81%	**93.71%**
PROPN	**93.09%**	0.00%
PUNCT	100.00%	100.00%
SCONJ	**53.62%**	44.93%
VERB	90.43%	90.43%
X	**52.63%**	42.11%

Table 4: Recall per POS tag per trial.

As for recall, the best performing classes included, once again, closed-class words (adpositions, coordinating conjunctions, pronouns, punc-

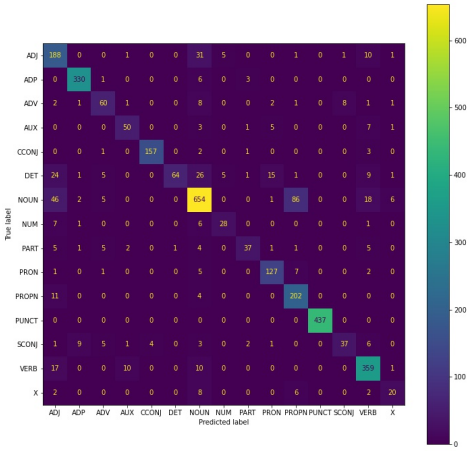

Figure 1: The confusion matrix for tokens with original capitalization.

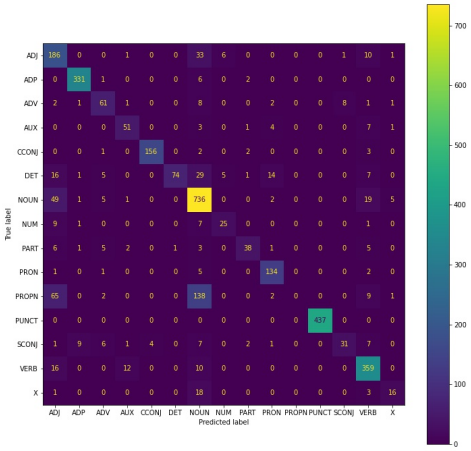

Figure 2: The confusion matrix for lowercase tokens.

tuation), but also open-class proper nouns (in the capitalized version of the data) and verbs. What is worth pointing out again, is that when capitalization was removed, the recall of proper nouns dropped to 0%. The worst performing classes - the ones where the fewest tokens were assigned their original tag correctly - include determiners, numerals, particles, and the X category. This is not entirely unexpected, as Polish does not really have determiners, and the words that are tagged as such include quantifiers and possessive pronouns, but can sometimes function as other categories. Similarly, words that can be classified as numerals and particles in other contexts may be assigned other classes, such as adjectives or conjunctions. Finally, the X category is given to abbreviations and words from foreign languages. Given how varied that makes this class, it is of little surprise that the tagger struggles with it. It seems that aside from the proper noun conundrum, capitalization does not clearly give better results, as better recall values seem to be spread out somewhat evenly between the two data versions.

In order to answer the question what kinds of tokens were wrongfully assigned to what class the most often, confusion matrices were generated for both types of data, included here as Figures 1 and 2. What definitely stands out for the data with original capitalization, is the number of nouns misclassified as proper nouns. As mentioned before, this is likely due to irregular capitalization of normal nouns. A fair bit of adjectives and determiners were misctegorized as nouns, perhaps due to spelling variations - and a number of nouns and determiners were, in turn, wrongfully assigned the label of adjectives. A few nouns were also misclassified as verbs. A certain number of X label tokens were assigned to nouns.

When it comes to the lowercase data, no proper nouns were correctly identified, and instead assigned to adjectives, nouns, and a few other classes. Similar miscategorization occurs between nouns, pronouns, and adjectives, as well as the X label. What can also be noticed is that for other badly performing classes there was no very clear preference for a wrong category, but instead those classes were not very numerous to begin with, so even a few misclassifications had a noticeable effect on the performance.

3.2 Qualitative Error Analysis

A qualitative analysis of the problematic tokens in the text with the original capitalization has revealed a few categories that are problematic for the tagger. As for adjectives, a number of them that were written with a final "y" instead of a "j" (in the singular feminine dative, accusative, or locative forms) was misclassified as nouns or verbs, which, looking strictly at the morphology, is not entirely unreasonable, as word-final "y" appears often in plural nouns or past non-masculine plural forms of verbs. The same applies, however,

for some adjectives in singular masculine nominative (or vocative) which end with that letter even in modern times, as in this case it signifies a vowel, not a consonant - those were still similarly miscategorized. A number of instances were misclassified as numerals, when they were, in fact, ordinal numbers, but written using Arabic numerals. Finally, one case where more than just the final /j/ sound was spelled using "y" was classified as X.

Most of the adpositions that were misclassified seem old-fashioned or contain spelling different from what is used nowadays, e.g. "wedle" instead of "według" ("according to"), "dokoła" instead of "dookoła" ("around"), "podpodczas" instead of "podczas" ("while"). Other ones, like "około" ("around") were misclassified as particles; according to Słownik Języka Polskiego (The Dictionary of the Polish Language) though, the word can be classified as either one or the other, with no clear indication of how context could help differentiate between the two (PWN Editorial Team, n.d.a).

As for adverbs, two cases of a word-final "y" lead to misclassification as noun (adverbs do have a word-final /j/ sound in the superlative form). Another nonstandard spelling, "wtencza" for "wtenczas" ("at that time") lead to the word being classified as verb, but the full form was later also miscategorized as an adposition, meaning that the tagger would have struggled with it anyway. Finally, many instances of "gdy" ("when") were misclassified as subordinating conjunctions; however, this word can play both roles in an utterance, with it being labelled as a subordinating conjunction when it is introducing a condition in the sentence, which was not the case in these examples.

In terms of auxiliaries, many problems stemmed from the semiauxiliary "to" ("is") being spelled the same way as the pronoun "to" ("it") and from the rule that in case the verb "być" ("to be")[2] is an existential one, and not a copula, it should be marked as a regular verb. Since there is nothing in the form of these words to differentiate between the two categories, the tagger struggled, as it seems to rely on the structure indeed.

Among coordinating conjunctions some of the issues can be explained by a possible dual nature of the word; most of them, though, display no clear reasons for the misclassification.

There were many issues with determiners being miscategorized. In some situations it is again the case of the same word form occuring also as a regular pronoun, and not only a possessive one; the trailing "y" seems to also be causing issues in many cases. Additionally, some nonstandard use can also be noted, which is definitely problematic for the tagger, such as "w one czasy" instead of "wówczas" ("back in those times"), "wieleż" instead of "wiele" ("many"), or "któremi" instead of "którymi" ("with which").

Many nouns were misclassified as proper nouns because of unexpected capitalization that seems to be used as a sign of respect by the author (e.g. writing "Dziad" instead of "dziad" ("grandfather," "old man") when talking about his grandfather). Some words seem to be confusing due to their archaic or highly domain-specific nature, while other ones end in what could be guessed to be a suffix by happenstance. Those are most often misclassified as verbs or adjectives.

Interestingly enough, while some ordinal numbers, which should be categorized as adjectives, were misclassified as numerals, the opposite is true for some of the cardinal numbers, which were most often miscategorized as adjectives or nouns. Some of those were spelled in a nonstandard way ("ośm" for "osiem" ("eight")), but the majority were spelled the same as nowadays. Perhaps this was a problematic situation due to numerals being essentially a theoretically infinite class of words, which means that some of them may not have appeared in the training data.

Particles do not show a clear trend as to what class they get miscategorized as, with the exception of "to," which, as mentioned before, can also be a pronoun or an auxiliary, leading to understandable confusion - and a similar case can be made for some other tokens (as the aforementioned "około" ("around") or "czy" ("whether"). Some variation in spelling is also present in this category, with "pono" instead of "ponoć" ("allegedly") and "jaszcze" instead of "jeszcze" ("still," "additionally") potentially complicating the task.

[2]While both "być" and "to" can be translated as "to be" and annotated as auxiliaries, they are not the same. "To" can also function as the pronoun "it," and it is from that meaning that some more fixed phrases evolved, such as "jest to" ("it is") where "jest" ("is") can be dropped, where it should be classified as a particle according to PWN Editorial Team (n.d.b). However, according to Universal Dependencies (n.d.) "copular uses" of "to" should be marked as auxiliaries, and marking both components of "jest to" as auxiliaries is attested for in the UD treebanks.

Error Type	Definition	Example	Prediction	Standard
unidentified	Errors the reason for which could not be pinpointed	*część **wsi*** (a part **of the village**)	ADJ	NOUN
capitalization	Errors due to nonstandard capitalization	*móy **Dziad*** (my **grandfather**)	PROPN	N
ending	Errors caused by a confusing word ending	*jego **fortuny*** (his **fortune**)	ADJ	NOUN
ambiguous	Words the class of which depends on the context	*Syn **jego*** (**his** son)	PRON	DET
archaic	Words misclassified due to archaic or nonstandard spelling	*Zostawiasz **mię*** (you leave **me**)	NOUN	PRON
y	Errors due to the frequent use of *y* instead of *j* for the /j/ sound	*Ojciec **móy*** (**my** father)	ADJ	DET
UD	Errors stemming from the UD annotation rules for auxiliaries	***Był** podobno* (**he was** supposedly)	AUX	VERB
surname	Errors due to Polish surnames inflecting in the adjectival paradigm	*kuzyn **Polanowskich*** (**Polanowscy's** cousin)	ADJ	PROPN
similar	Errors due to the similarity of a foreign word to a Polish one	***daruju** ukochanomu* (**I give** the beloved)	NOUN	X
misspelled	Errors due to a misspelling of the original word (different from nonstandard or archaic spelling)	*po **mim*** (after **him**)	NOUN	PRON

Table 5: Types and examples of errors.

Most of the mistakes that the tagger made when classifying pronouns stem from them being written with a capital letter at the start, once again, denoting respect, but confusing the annotation tool. Once more "to" was causing issues in this category too, for the reasons mentioned above.

Finally, proper nouns were also widely misclassified, sometimes as regular nouns, but often as adjectives. This quite reasonable, as many of those were surnames, which in Polish often inflect using the adjectival paradigms, but which are categorized as proper nouns.

Many subordinating conjunctions ended up with the wrong tags as well, being assigned a variety of other classes. In a number of cases that was likely due to the same word being able to appear with a different tag (interestingly enough, one of them, "więc" ("so"), can be both a subordinating and a coordinating conjunction), but in many it was probably because of the rarity or nonstandardness of the word, as in "dopokąd" ("until") or "toteż" ("so", "consequently"). In one case a compound of "bo" ("because") and "śmy" ("we were / we did"), a conjunction and auxiliary, was manually tagged as a conjunction, since that meaning was more salient to the annotator.

A big number of "być" verbs that were annotated according to the UD guidelines as content verbs were misclassified as auxiliaries. Some participles, which should also remain tagged as verbs,

Error Type	Raw freq.	Relative freq.
unidentified	156	30.00%
capitalization	95	18.27%
ending	95	18.27%
ambiguous	66	12.69%
archaic	41	7.88%
y	25	4.81%
UD	22	4.23%
surname	10	1.92%
similar	7	1.35%
misspelled	3	0.58%

Table 6: Frequency of errors in the text with original capitalization.

were miscategorized as adjectives, and the same goes for impersonal verb forms. Those verbs that were classified as nouns were most often instances of the author writing the verb and the preceding negation together, which was also left that way in the annotated text.

Finally, many of the words and abbreviations that were originally tagged as X were misclassified; a very prominent category here is the author's transcription of a sentence in Russian using Polish alphabet. Since the two languages are relatively similar, it is not unreasonable to expect that the tagger actually interpreted them as Polish words. A number of other words in Latin or German got classified as proper nouns due to them being spelled with a capital letter.

In the lowercase data many of the same issues persisted, as the lowercasing only influenced the situations where words were mistakenly classified as proper nouns based on capitalization. A number of them still were not properly classified, as for instance the noun "mandataryusz" ("mandatary") was assigned the verb category, for seemingly no reason - it is, however, not only a word with the old spelling of the sound /j/, but also a rather archaic term. Naturally, all the proper nouns were misclassified this time, in large part as nouns, but also as adjectives, as per the explanation concerning surnames.

Overall, while some of the erroneous classification can be explained away by capitalization, confusing word endings, variation in spelling, the ability of some words to function as multiple classes, and archaic vocabulary, some of the instances are hard to justify. Overall, 10 error categories were identified, and the errors were classified into them. It is worth keeping in mind that some of these categories are overlapping (e.g. *y* and *archaic*), and they are somewhat arbitrary. The statistics and examples for the error categories can be seen in Tables 6 and 5.

4 Discussion

On their own, the results of the quantitative evaluation show a surprisingly good performance of the tagger, comparable to the ones from Rayson et al. (2007), where the Early Modern English data that was not pre-processed in terms of spelling variation yielded an 81.94% accuracy and some of the results from Bollmann (2013), with the top result before normalization being 83.41%. On the other hand, much of the other research reported lower scores, and the results across different test data within the studies also vary. It is not unreasonable to assume that the tagger's high performance is partly due to the test data not being very old, and, as a consequence, not too perplexing. Another issue is that not all the taggers in the studies operate on the same algorithm, which makes direct comparisons difficult. Nevertheless, this particular maximum entropy-base tool performs on this test data only around 13 percentage points below the accuracy it has shown on its own test set. What has also been shown is that different classes of tokens can be more or less confusing for the tagger, and while some of them, on their own, showed really bad performance, they were not numerous classes, meaning that their influence on the final score, both accuracy and MCC, was not that drastic.

A qualitative analysis of the errors made by the tagger has approximated what it struggles with in the test data. Previous studies have shown that variation in spelling, capitalization, punctuation, differences in morphology and syntax, and semantic shifts are some of the factors that make accurate tagging of historical texts using modern taggers difficult. In the case of this particular tool, some of those issues, such as, seemingly, variation in spelling (most prominently "y" written instead of "j"), capitalization of regular nouns and pronouns, relative prevalence of impersonal verb forms, as well as rare and archaic vocabulary have negatively impacted the tagger's performance. However, there also seems to be an issue when it comes to words that can function as multiple parts of speech, depending on the context (e.g. "tak" be-

ing able to signify "yes" and "in such a way," "to" meaning "it" or "is," both as an auxiliary and main verb). Other tagger-specific problems include some words being classified seemingly only on the basis of their ending, which resembles an inflectional ending as well as simply misassigning a class without a salient reason for it. While such issues have not been mentioned in prior research, it is safe to assume that they could have been omitted, as they are tool-specific and not as much data-specific. While such an idea was posited in the introduction, variation in word order or morphology do not seem to influence the performance that much, with the core issues either being specific to the tagger, or universal when compared to other languages that have been tested this way.

Finally, comparing the results on the data with original capitalization and with all the tokens being lowercase highlighted the issue with proper noun detection, which appears to be based on capitalization. However, capitalization of nouns and pronouns as a sign of respect is present in the test data, complicating the situation. Aside from the proper nouns, capitalization or lack thereof did not appear to drastically change the performance.

5 Conclusions and Future Work

Throughout this project a modern Polish POS tagger has been successfully evaluated on historical data, and issues causing the drop in its performance have been identified, along with the influence of capitalization on the tagging performance. It has been shown that a modern Polish tagger can perform relatively well on non-standard, historical Polish data from the late 19th century. Many of the misclassified tokens were causing problems due to issues previously identified in the literature in the field; however, some problems seemed to be inherent to the tagger itself. Capitalization was shown to be essential in the classification (and misclassification) of proper nouns.

While it was beyond the scope of this project, it would be interesting to see the influence of other factors, such as punctuation, on the quality of tagging. Another possible future research project could be comparing the performance of multiple different taggers or tagging architectures on the data, or testing the same tagger on data from different time periods. Alternatively, one could juxtapose there results to those from tagging a very recent, nonstandard text, e.g. sourced from the web, to see to what extent the same issues are causing tagging problems, and perhaps to what extent they could be remedied using the same methods. Finally, developing some method for the preprocessing of texts from this time period for subsequent tagging could also be interesting.

References

Yvonne Adesam and Gerlof Bouma. 2016. Old Swedish Part-of-Speech Tagging between Variation and External Knowledge. In *Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*. Association for Computational Linguistics, Berlin, Germany, pages 32–42. https://doi.org/10.18653/v1/W16-2104.

Marcel Bollmann. 2013. POS Tagging for Historical Texts with Sparse Training Data. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*. Association for Computational Linguistics, Sofia, Bulgaria, pages 11–18. https://aclanthology.org/W13-2302.

Dieuwke Hupkes and Rens Bod. 2016. POS-tagging of Historical Dutch. In *LREC 2016: Tenth International Conference on Language Resources and Evaluation*. European Language Resources Association (ELRA), Paris, pages 77–82. https://hdl.handle.net/11245/1.535946.

PWN Editorial Team. n.d.a. około - definicja, synonimy, przykłady użycia. https://sjp.pwn.pl/slowniki/oko%C5%82o.html. Accessed: 29.12.2022.

PWN Editorial Team. n.d.b. to - definicja, synonimy, przykłady użycia. https://sjp.pwn.pl/szukaj/to.html. Accessed: 30.12.2022.

Paul Rayson, Dawn Archer, Alistair Baron, Jonathan Culpeper, and Nicholas Smith. 2007. Tagging the Bard: Evaluating the accuracy of a modern POS tagger on Early Modern English corpora .

Silke Scheible, Richard J. Whitt, Martin Durrell, and Paul Bennett. 2011. Evaluating an 'off-the-shelf' POS-tagger on early Modern German text. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*. Association for Computational Linguistics, Portland, OR, USA, pages 19–23. https://aclanthology.org/W11-1503.

The University of Sheffield. n.d. Universal dependencies POS tagger for pl / Polish. https://cloud.gate.ac.uk/shopfront/displayItem/tagger-pos-pl-maxent1. Accessed: 29.12.2022.

Universal Dependencies. n.d. UD for Polish. https://universaldependencies.org/pl/index.html. Accessed: 29.12.2022.

Universal Dependencies. n.d. Universal Dependencies. https://universaldependencies.org/. Accessed: 29.12.2022.

Yi Yang and Jacob Eisenstein. 2016. Part-of-Speech Tagging for Historical English. pages 1318–1328. https://doi.org/10.18653/v1/N16-1157.

A Appendix

Within this section links to the code and the data used in this project are provided. The tagger itself is listed under references.

1. The project repository: https://github.com/Turtilla/ltr-project
2. The project code and results: https://github.com/Turtilla/ltr-project/blob/main/ltr-project.ipynb
3. The annotated text used in the testing: https://github.com/Turtilla/ltr-project/blob/main/memoirs_annotated_3k.txt
4. The full text of the memoirs: https://github.com/Turtilla/ltr-project/blob/main/memoirs.txt
5. The confusion matrices in a higher quality: https://github.com/Turtilla/ltr-project/tree/main/images