My grat destress – POS tagging 18th-century English letters

Lassi Saario-Ramsay[1], Janine Siewert[1,2], Lidia Pivovarova[1], Akseli Kettunen[1], Inga Kokkonen[1], Tanja Säily[1],
[1]University of Helsinki, Finland, [2]University of Augsburg, Germany,
Correspondence: tanja.saily@helsinki.fi

Abstract

This paper investigates the POS annotation of letters from women and the lower classes in 18th-century England (Late Modern English) as a low-resource NLP task. The dataset of 30 letters with manually corrected annotation created for this task is made available upon request. Using this dataset, we compare the commonly used statistical CLAWS tagger with shallow and deep neural approaches.

1 Introduction

Although POS-tagging present-day standard English is basically a solved task, historical varieties still pose some challenges. One hitherto neglected aspect is the impact of sociolinguistic variation on tagging accuracy: Saario et al. (2021, 119–120) show that while the accuracy may be acceptable on average, in a sociohistorical corpus it varies by the gender and education level of the writers, which is problematic if we wish to study the language use of women and the lower social ranks. In this paper, we compare the accuracies of three POS-tagging approaches on a socially representative sample of 18th-century English letters, treating this as a low-resource NLP task. The approaches include a statistical tagger, shallow neural models—both with and without pre-training, and large language models with and without fine-tuning.

Corpora of Early and Late Modern English are most often tagged with the CLAWS tagger (the Constituent Likelihood Automatic Word-tagging System; see cla). CLAWS was developed for Present-day English, where it may reach accuracies of 96–97%, and it is the *de facto* standard in the field (Saario et al., 2021). However, the earlier the period, the lower the accuracy: for instance, Schneider et al. (2016, 258) find that even with prior spelling normalization, the accuracy of CLAWS varies in the ARCHER corpus from 87.8% in the 17th century to 93.2% in the 18th and 95.0% in the 19th (see also Rayson et al. 2007). Schneider et al. (2016) attempt to improve the output by using an ensemble approach and find that combining two taggers automatically, when their methods are sufficiently different, raises tagging performance by 0.78%, whereas adding a human to the loop leads to further improvements.

While many of the current Modern English corpora are still CLAWS-tagged, recent work on POS-tagging evaluation has focused on language models and the Penn tagset used in the Penn historical corpora. Manjavacas and Fonteyn (2022, 5–6) compare different BERT models and find that their historically pre-trained MacBERTh model performs the best in the *Penn-Helsinki Parsed Corpus of Early Modern English*, although its accuracy on the test dataset still remains below 90%. Using pre-trained ELMo embeddings and a simplified Penn tagset, Kulick et al. (2022) achieve accuracies of up to 98.30% in *Early English Books Online*.

Our research complements the work outlined above in that we test modern language models on the widely used CLAWS7 tagset.[1] Moreover, previous work is based on corpora that mostly represent published texts, which were produced by a very small section of society. In contrast, our dataset consists of personal letters, which could be written by anyone who was literate, enabling us to analyse sociolinguistic variation in tagging accuracy. The two main research questions we aim to address here are: (1) To what degree are sociolinguistic differences reflected in the tagging accuracy? (2) Can neural approaches attain a comparable accuracy to the CLAWS tagger without the additional effort of spelling normalization?

2 Data

Our historical dataset comes from the *Corpora of Early English Correspondence* (CEEC), which

[1] https://ucrel.lancs.ac.uk/claws7tags.html

were compiled for the purposes of historical sociolinguistics and aim to represent the language use of men and women from the lowest to the highest social ranks in the period 1400–1800. In particular, our letter samples have been extracted from the *Corpus of Early English Correspondence Extension* (CEECE), which focuses on original-spelling letters from the eighteenth century, or Late Modern English. The samples comprise two sets of 15 letters (c. 400 tokens per letter) representing a wide social spectrum from paupers to noblemen and women; see Appendix A. One of these sets is used for training and the other for testing. The automatic annotation of these has been manually corrected by two of the authors for training and testing purposes, and this gold-standard dataset is available upon request by email.

In addition, we use the *British National Corpus Sampler* (BNC, 2005), a two-million word corpus of Present-day English that has been POS-tagged using the CLAWS7 tagset and subsequently hand-corrected. This corpus contains written and spoken material from the 1990s. It was selected because it provides a large amount of hand-corrected tagged data in our tagset of interest. Note that CLAWS7 is a rather fine-grained standard, with more than a hundred different tags.

3 Automatic annotation results

3.1 Statistical baseline and manual corrections

The first steps towards providing the CEECE with POS tagging were taken over a decade ago. First, the most frequent variant spellings were normalized using the VARD software (Baron, 2026). Some further normalization was carried out more or less manually, focusing on abbreviations, reflexive pronouns, indefinite pronouns and related adverbs (Saario and Säily, 2020, § 3; Saario et al., 2021, 110–112). The partially normalized CEECE was then tokenized and POS-tagged by the CLAWS tagger in the CLAWS7 tagset. Example 1 shows the original (a), normalized (b) and tagged version (c); "Kind" is tagged incorrectly as a noun.

1. (a) I receved your Cind Letter
   (b) I received your Kind Letter
   (c) I_PPIS1 received_VVD your_APPGE Kind_NN1 Letter_NN1

The accuracy of tagging was checked in a sample of 15 letters and estimated to be c. 94.5%. However, many of the perceived tagging errors are actually due to errors in tokenization. Since the errors were counted token by token, errors in tokenization also affected the very way in which the accuracy of tagging was calculated (Saario and Säily, 2020; Saario et al., 2021).

Later, a gold standard for both tokenization and tagging was produced from the said sample and from another, similar sample of 15 letters. The gold standard was first prepared by one annotator and then checked by another annotator. Unclear cases were discussed and resolved together. The tokens and tags from the CLAWS output were then aligned with those of the gold standard, allowing us to calculate the performance of CLAWS in a more principled way. For each token in the gold standard, including punctuation marks, we checked whether CLAWS had assigned the same tag to the same token. If CLAWS had assigned a different tag or if there was no matching token in the CLAWS output due to misalignment in tokenization, this counted as one incorrectly tagged token. The accuracy of tagging is now estimated to be 94.23% for the first sample and 94.28% for the second one.

3.2 Shallow neural models

As examples of shallow neural models, we use Stanza (Qi et al., 2020) and MaChAmp (van der Goot et al., 2021), both with their default settings and three training setups: 1) training on the BNC alone, 2) training on the BNC and fine-tuning on the first 15 letter sample of the CEECE, and 3) training only on the 15 letter sample of the CEECE. The 15 letter sample was split into 10 letters for the train set and 5 for the development set. The best model was selected based on the development set. The results are presented in Table 1. We present a gender and a social rank of the sender, according to CEECE classification. The detailed metadata are shown in Appendix A. It can be seen that the writing by a lower-class female—PAUPER_011—yields the lowest results by all methods, though correlation between the scores and the rank is not straightforward and probably influenced by the train-dev split in the training set.

The surprisingly low accuracy of models pretrained on the BNC results from a mismatch in the annotation of punctuation in the BNC versus the CEECE. While this was apparently fixed by the CEECE-finetuning in case of MaChAmp, punctuation was tagged as <UNK> by Stanza pretrained on the BNC and finetuned on the CEECE. Interestingly, when assuming correct punctuation in the

Document	G	Rank	Stanza			MaChAmp		
			BNC only	BNC → CEECE	CEECE only	BNC only	BNC → CEECE	CEECE only
BENTHAJ_016	M	Professional	80.83	81.94	**84.17**	81.94	**94.44**	93.33
BLOMEFI_034	M	Clergy (low)	79.29	79.55	**82.83**	83.84	**94.95**	93.43
BURNEY_040	M	Professional	77.34	77.34	**78.91**	77.60	**92.45**	90.10
CARTER_015	F	Clergy (low)	85.49	**86.53**	83.68	90.41	**95.34**	92.49
DUKES_089	M	Nobility	80.60	**82.37**	81.11	85.39	**93.45**	90.68
HADDOC2_008	M	Professional	73.74	70.11	**81.01**	76.82	**91.90**	91.62
HATTON2_048	F	Gentry (low)	70.91	76.59	**81.59**	77.73	**86.82**	85.68
PAUPER_011	F	Other	74.66	**78.28**	**78.28**	73.30	**76.02**	74.66
PINNEY_056	M	Merchant	80.89	**81.99**	71.75	85.87	**86.43**	**86.43**
PRIDEA2_035	M	Clergy (up)	79.40	80.22	**84.62**	81.32	**91.76**	89.29
SANCHO_034	M	Other	69.64	69.84	**82.19**	72.87	**93.32**	91.50
SWIFT_033	F	Nobility	77.72	82.08	**85.23**	85.96	**93.70**	91.77
SWIFT_036	M	Nobility	78.53	82.65	**86.76**	84.41	92.94	**94.12**
TWINING_006	F	Clergy (low)	80.21	**80.48**	79.68	81.82	**92.25**	91.98
YOUNG_026	M	Clergy (low)	70.81	75.38	**84.77**	76.14	**93.91**	92.64
Overall	–	–	77.17	78.85	**81.89**	81.06	**91.76**	90.39

Table 1: Per-letter accuracy of Stanza and MaChAmp under different fine-tuning settings. The metadata columns give sender gender and rank; F = female, M = male. Bold indicates the best score for each model and document, underlining shows the best overall score.

accuracy calculation, MaChAmp BNC outperforms MaChAmp BNC + CEECE on 6 out of 15 letters. The finetuning might thus primarily improve the annotation of punctuation, while the quality could suffer for other tags. Perhaps the train set was too small, leading to a loss of generalization ability.

3.3 Large language models

We use the BNC and CEECE to fine-tune QWEN2.5-72B-INSTRUCT. First, the model was fine-tuned on the BNC and then fine-tuning continued using the CEECE. We used a detailed prompt that defines the output format and includes the full definition of all CLAWS7 tags[2].

The BNC was randomly split into training and validation samples, and the best model was selected using validation performance. For the CEECE corpus we used leave-one-out evaluation to determine the optimal number of training epochs, which was 5 in our case. Then we used all 15 letters to train the final model for 5 epochs and applied it to the testset of the other 15 manually annotated letters.

We also experimented with smaller models from the same Qwen2.5 family, as shown in Table 2. For most model sizes, fine-tuning on the letter improves accuracy by approximately three percentage points (pp). However, the smallest 0.5B model degrades from fine-tuning, suggesting that it is less able to generalize from the idiosyncrasies of non-standard texts. The best-performing model is the largest one, but the difference between 72B and 7B models is only 0.7pp, indicating that a substantially smaller model may be sufficient for this task in practice.

Model size	Epoch					
	0	1	2	3	4	5
0.5B	88.42	74.72	80.90	84.98	86.57	86.57
1.5B	90.56	90.35	92.73	91.99	92.20	92.20
3B	92.33	92.14	93.20	93.20	93.29	93.39
7B	91.94	92.77	94.97	94.97	95.13	95.18
14B	92.22	94.90	95.07	94.71	94.83	95.18
32B	92.01	93.91	94.82	94.98	95.02	94.95
72B	92.29	92.78	93.73	94.21	94.83	**95.25**

Table 2: Accuracy by model size and leave-one-out fine-tuning epoch. Epoch 0 corresponds to direct inference using a model fine-tuned on the BNC only.

Table 3 shows the models' performance on the testset. As can be seen from the table, the base, out-of-the-box model yields rather miserable performance, since it is unable to follow the instructions given in the prompt and stick to the CLAWS7 tagset. Fine-tuning on 15 letters only is not sufficient to learn this task, so using the BNC is crucial. The best results are obtained by first fine-tuning on the BNC and then continuing with the CEECE.

Manual investigation showed that one of the main difficulties is Ditto (multi-word) tags that are used to annotate idiomatic expressions treated as a

[2]The full prompt is shown in Appendix B. Fine-tuning used LoRA with HuggingFace PEFT, applying adapters to all linear layers, with rank $r = 48$, scaling parameter $\alpha = 96$, dropout 0.05, and the default AdamW optimizer in the HuggingFace Trainer. The learning rate was $1e{-}4$ for BNC and $1e{-}5$ for the letters. Documents were split into 4096-token chunks.

eval	Fine-tuning dataset			
	-	CEECE only	BNC only	BNC → CEECE
strict	72.58	84.34	92.15	**92.26**
–ditto	72.90	84.53	**93.38**	93.35
–punct	70.89	83.22	91.78	**93.09**

Table 3: Accuracy for the QWEN2.5-72B-INSTRUCT model fine-tuned on different combinations of data and tested on the letter test set.

Document	Acc	-Punct	-Ditto	Level 1
BENTHAJ_016	93.89	95.60	94.72	95.83
BLOMEFI_034	92.93	92.61	92.93	93.69
BURNEY_040	88.80	94.94	90.89	91.15
CARTER_015	96.37	96.43	97.15	97.93
DUKES_089	94.21	93.44	94.21	95.21
HADDOC2_008	93.58	92.58	94.13	96.37
HATTON2_048	91.14	90.23	91.59	94.09
PAUPER_011	84.16	84.55	87.33	90.50
PINNEY_056	88.37	88.80	90.03	92.52
PRIDEA2_035	95.05	94.50	95.05	95.33
SANCHO_034	85.22	95.94	86.23	87.45
SWIFT_033	97.58	97.33	98.06	99.03
SWIFT_036	94.41	94.16	95.00	96.18
TWINING_006	95.45	95.15	97.06	97.86
YOUNG_026	91.37	87.02	95.43	95.94

Table 4: Per-document results for the best LLM model on the test set. Level-1 evaluation concerns only the first letter in the CLAWS7 tag, roughly corresponding to coarse-grain part of speech.

unit. For example, "at least" is tagged as an adverb (`at_RR21 least_RR22`), "so that" as a subordinating conjunction (`so_CS21 that_CS22`). A more relaxed evaluation where the numbers in the ditto tags are ignored yields 1pp improvement, as shown in the second row of Table 3. Table 4 shows per-document results for the best model.

Another problem found in the LLM results is related to the punctuation tags: the model is struggling to follow the conventions and frequently uses arbitrary tags for the punctuation. To further investigate this issue, we conduct evaluation ignoring all punctuation. In this evaluation, the overall score went down, as the bottom row of Table 3.

However, per-document results indicate a more nuanced picture: for some letters evaluation results calculated without punctuation are much higher than the strict evaluation—e.g., BURNEY_040, SANCHO_034—which means that most of the errors for these documents were made in this domain.

Document	CLAWS	Shallow	LLM
BENTHAJ_016	**95.28**	94.44	93.89
BLOMEFI_034	**96.46**	94.95	92.93
BURNEY_040	**94.53**	92.45	88.80
CARTER_015	**96.89**	95.34	96.37
DUKES_089	**95.97**	93.45	94.21
HADDOC2_008	**94.13**	91.90	93.58
HATTON2_048	**93.41**	86.82	91.14
PAUPER_011	77.38	76.02	**84.16**
PINNEY_056	**88.37**	86.43	**88.37**
PRIDEA2_035	**96.43**	91.76	95.05
SANCHO_034	**94.94**	93.32	85.22
SWIFT_033	95.16	93.70	**97.58**
SWIFT_036	**96.47**	92.94	94.41
TWINING_006	**95.99**	92.25	95.45
YOUNG_026	**95.43**	93.91	91.37
Overall	**94.28**	91.76	92.26

Table 5: Accuracy of tagging by document and overall.

	Accuracy			Support	
Document	CLAWS	Shallow	LLM	Docs	Tokens
Sender Gender					
Female	93.13	89.97	**93.73**	5	1,834
Male	**94.83**	92.62	91.55	10	3,848
Sender Rank					
Nobility	**95.83**	93.39	95.48	3	1,150
Gentry	**93.41**	86.82	91.14	1	440
Clergy	**96.24**	93.68	94.20	5	1,914
Professional	**94.65**	92.92	92.01	3	1,102
Merchant	**88.37**	86.43	**88.37**	1	361
Other	**89.51**	87.97	84.90	2	715
Year of writing					
Pre-1740	**94.54**	91.56	93.49	8	3,056
Post-1740	**93.98**	92.00	90.82	7	2,626

Table 6: Accuracy of tagging by contextual variables. Support indicates the number of documents and tokens in each category.

4 Analysis and discussion

We compare the performance of CLAWS with the best shallow neural model (MaChAmp BNC+TCEECE) and the best large language model (LLM) on the test sample of 15 letters. As Table 5 shows, CLAWS was the most accurate tagger, outperforming the shallow neural model for all letters and the LLM for 12 letters out of 15. For instance, in *send me sum relefe so That with industry imay bee able to git my bread* (PAUPER_011), only CLAWS tagged "so That" correctly as a subordinating conjunction with ditto tags (`so_CS21 That_CS22`), whereas the shallow model tagged the tokens as an adverb and a determiner (`so_RR That_DD1`), and the LLM as `so_CS That_CS`, which lacks ditto markup but is otherwise correct.

	CLAWS		Shallow		LLM	
Tag group	Prec.	Rec.	Prec.	Rec.	Prec.	Rec.
Punctuation	99.83	**99.66**	98.81	99.49	**100.00**	83.08
A-	**100.00**	**98.93**	96.58	96.38	99.14	98.29
C-	**95.28**	**93.80**	90.11	87.08	93.19	91.99
D-	94.65	**95.68**	89.20	84.86	**96.67**	94.05
I-	**96.82**	**96.31**	95.42	95.25	93.09	92.27
J-	93.96	88.30	88.04	86.17	**94.89**	**92.20**
M-	92.65	86.30	**93.24**	**94.52**	92.96	90.41
N-	89.81	89.63	89.70	89.43	**95.48**	**91.54**
P-	99.66	**99.14**	98.25	96.40	**100.00**	97.95
R-	**90.65**	**90.35**	75.29	82.32	82.46	86.17
V-	92.18	93.97	88.54	90.95	**98.38**	**94.55**
Misc	97.00	**89.81**	95.96	87.96	**97.47**	89.35

Table 7: Precision and recall by tag, summarized at the level of major tag groups (cf. Saario et al., 2021, 120). Bold indicates the best score in each row separately for precision and recall.

The greatest improvement in tagging accuracy is observed in PAUPER_011. This letter was authored by a poor woman and includes rather non-standard language (e.g. “my grat destress”, “made no youse”). The LLM exceeded CLAWS’s score by 6.79pp, even though it did not get any help from normalization. CLAWS, too, only received a limited amount of help because many of the spelling variants in the letter were too infrequent to be affected by our semi-automated normalization. For example, in *All my frends thare is Ded which mad me not Go*, CLAWS tagged “Ded” (‘dead’) as a singular proper noun (NP1), while the LLM correctly tagged it as an adjective (JJ).

The biggest drop in accuracy is also associated with the lowest social ranks, namely SANCHO_034 (-9.72pp from CLAWS to LLM). The author of this letter was a son of two slaves who nevertheless got an education and was friends with higher-class people. The problem here was his extensive use of dashes, which the LLM failed to tag entirely despite a similar letter in the training data: *You have missed the truth by a mile - aye and more - It was not neglect - I am too proud for that - it was not forgetfulness, Sir, I am not so ungrateful - it was not idleness, the excuse of fools - nor hurry of business, the refuge of knaves - It is time to say what it was*.

Table 6 shows how the results are distributed across sender gender, sender rank, and year of writing. While CLAWS continues to prevail in almost every category, there is social variation in its accuracy, which is lower for women and the lower social ranks of merchants and other non-gentry. Lower-class letters remain problematic to the LLM despite the improvement over CLAWS in the pauper letter, but it does slightly improve the tagging accuracy of women’s letters (+0.6pp).

Table 7 summarizes the precisions and recalls for groups of tags. CLAWS is the most precise tagger for articles (A-), conjunctions (C-), prepositions (I-), and adverbs (R-). The shallow model has the best precision for numerals (M-) and the LLM for punctuation, determiners (D-), adjectives (J-), nouns (N-), pronouns (P-), and verbs (V-). For most tag groups, the most precise model is also the one with best recall. However, CLAWS still has the best recall for punctuation, determiners and pronouns, although the LLM has a better precision. These results indicate that using ensemble methods could be a fruitful way forward (cf. Schneider et al., 2016).

5 Conclusion

Although CLAWS won the present comparison, it would not have won had its input not been partially normalized (Saario et al., 2021, 122–123), and the LLM in fact outperformed it on women’s letters, particularly the pauper letter. We expect that the performance of both shallow and deep models could be improved by normalization. However, given that the neural approaches often achieve fairly close and in some cases even identical or better accuracy than CLAWS, the manual work that would otherwise be spent on spelling normalization might be more meaningfully invested into producing a larger amount of gold standard POS annotation for training and testing. Moreover, the issues with punctuation and ditto tags could be alleviated through either prompt engineering or an ensemble system. Future research could also extend the benchmarking to other open-weight LLMs as well as commercial models.

References

CLAWS part-of-speech tagger for English.

2005. The BNC Sampler, XML version. Distributed by Oxford University Computing Services on behalf of the BNC Consortium.

Alistair Baron. 2026. VARD 2.

CEEC. Corpora of Early English Correspondence. Compiled by Terttu Nevalainen, Helena Raumolin-Brunberg, Samuli Kaislaniemi, Jukka Keränen, Mikko Laitinen, Minna Nevala, Arja Nurmi, Minna

Palander-Collin, Tanja Säily and Anni Sairio at the Department of Languages, University of Helsinki.

Seth Kulick, Neville Ryant, and Beatrice Santorini. 2022. Parsing early Modern English for linguistic search. In *Proceedings of the Society for Computation in Linguistics 2022*, pages 143–157, online. Association for Computational Linguistics.

Enrique Manjavacas and Lauren Fonteyn. 2022. Adapting vs. pre-training language models for historical languages. *Journal of Data Mining & Digital Humanities*, NLP4DH.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

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. In *Proceedings of Corpus Linguistics 2007, 27–30 July, University of Birmingham, UK*. Article 192.

Lassi Saario and Tanja Säily. 2020. POS tagging the CEECE: A manual to accompany the *Tagged Corpus of Early English Correspondence Extension* (TCEECE).

Lassi Saario, Tanja Säily, Samuli Kaislaniemi, and Terttu Nevalainen. 2021. The burden of legacy: Producing the *Tagged Corpus of Early English Correspondence Extension* (TCEECE). *Research in Corpus Linguistics*, 9(1):104–131.

Gerold Schneider, Marianne Hundt, and Rahel Oppliger. 2016. Part-of-speech in historical corpora: Tagger evaluation and ensemble systems on ARCHER. In *Proceedings of the 13th Conference on Natural Language Processing (KONVENS 2016)*, pages 256–264. Ruhr-Universität Bochum.

Rob van der Goot, Ahmet Üstün, Alan Ramponi, Ibrahim Sharaf, and Barbara Plank. 2021. Massive choice, ample tasks (MaChAmp): A toolkit for multi-task learning in NLP. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 176–197, Online. Association for Computational Linguistics.

A Metadata for the CEECE training and test datasets

Letter ID	Sender Name	Gender	Rank	Year	Token Count
BENTHAJ_056	Alleyne Fitzherbert	Male	Nobility	1792	389
BURNEY_039	Charles Burney	Male	Professional	1784	410
BURNEYF_013	Frances (Fanny) Burney	Female	Professional	1779	420
CARTER_023	Elizabeth Carter	Female	Clergy (Lower)	1740	426
CLIFT_027	Joanna Clift	Female	Other	1795	364
DUKES_052	Charles Lennox	Male	Nobility	1741	370
FLEMIN2_133	George Fleming	Male	Gentry (Lower)	1691	380
GARRICK_032	David Garrick	Male	Gentry (Lower)	1753	454
GIBBON_007	Edward Gibbon	Male	Gentry (Lower)	1758	331
PEPYS3_035	Samuel Pepys	Male	Professional	1680	340
SANCHO_024	(Charles) Ignatius Sancho	Male	Other	1778	377
SWIFT_059	Elizabeth Germain née Berkeley	Female	Nobility	1731	368
TWINING_010	Elizabeth Twining née Smythies	Female	Clergy (Lower)	1765	423
WEDGWOO_023	Josiah Sr Wedgwood	Male	Professional	1769	423
WENTWO2_146	Richard Wardman	Male	Other	1734	430

Table 8: CEECE training set document metadata sorted by letter ID.

Letter ID	Sender Name	Gender	Rank	Year	Token Count
BENTHAJ_016	Jeremy Bentham	Male	Professional	1772	360
BLOMEFI_034	Francis Blomefield	Male	Clergy (Lower)	1748	396
BURNEY_040	Charles Burney	Male	Professional	1784	384
CARTER_015	Elizabeth Carter	Female	Clergy (Lower)	1739	386
DUKES_089	Charles Lennox	Male	Nobility	1746	397
HADDOC2_008	Richard Haddock	Male	Professional	1709	358
HATTON2_048	Elizabeth Hatton née Scroggs	Female	Gentry (Lower)	1690	440
PAUPER_011	Ann Clark	Female	Other	1768	221
PINNEY_056	Nathaniel Pinney	Male	Merchant	1706	361
PRIDEA2_035	Humphrey Prideaux	Male	Clergy (Upper)	1710	364
SANCHO_034	Ignatius Sancho	Male	Other	1779	494
SWIFT_033	Mary Butler née Somerset	Female	Nobility	1723	413
SWIFT_036	John Carteret	Male	Nobility	1724	340
TWINING_006	Elizabeth Twining née Smythies	Female	Clergy (Lower)	1764	374
YOUNG_026	Edward Young	Male	Clergy (Lower)	1730	394

Table 9: CEECE testset document metadata sorted by letter ID.

B Prompt used for LLM fine-tuning

```
You are a CLAWS7 part-of-speech tagger for 18th-century letters.

Your job:
- Read the input text and output token-level CLAWS7 tags.
- Keep original token order.
- Output one tag table and nothing else.

Hard output rules (must follow exactly):
- Output ONLY the caption line, the header line, and table rows.
- No reasoning, no preface, no markdown, no explanations.
- Never output <think> or any meta text.
- No blank lines.
- First line must be exactly:
  TAG TABLE
- Second 2 lines must be exactly:
  | token | tag | comment |
  |----------------------|
- After that, output one row per token in exactly this format:
  | token | tag | comment
- Use the pipe character "|" as the separator.
- Use "-" as the comment unless a very short note is truly needed.
```

```
- The number of table rows must exactly match the number of produced tokens.
- Do not output the string "<TAB>".

Surface-form fidelity rules:
- Copy token text from the input exactly, character for character, after tokenization.
- Do NOT modernize, normalize, regularize, or paraphrase spelling.
- Do NOT replace one surface form with another that seems equivalent.
- Preserve original capitalization exactly.
- Preserve apostrophes, periods, slashes, hyphens, and other punctuation exactly when they belong to the token.
- Do NOT expand abbreviations or titles.
- Every output token must come directly from the input text in the same left-to-right order.
- If a token appears in one spelling variant in the input, keep that exact variant in the output.
- If an abbreviation or title appears in the input, copy it exactly, including any period.

Tokenization rules:
- Split ordinary attached punctuation into separate tokens.
- Examples:
  Name | NP1 | -
  , | , | -
  Place | NP1 | -
  . | . | -
- Keep the period inside common abbreviations and titles when it belongs to the token.
- Examples:
  Mr. | NNB | -
  Dr. | NNB | -
  St. | NNL1 | -
- Keep "/" as its own token unless context clearly requires otherwise.

Allowed tags:
APPGE        possessive pronoun, pre-nominal (e.g. my, your, our)
AT         article (e.g. the, no)
AT1         singular article (e.g. a, an, every)
BCL         before-clause marker (e.g. in order (that),in order (to))
CC         coordinating conjunction (e.g. and, or)
CCB         adversative coordinating conjunction ( but)
CS         subordinating conjunction (e.g. if, because, unless, so, for)
CSA         as (as conjunction)
CSN         than (as conjunction)
CST         that (as conjunction)
CSW         whether (as conjunction)
DA         after-determiner or post-determiner capable of pronominal function (e.g. such, former, same)
DA1         singular after-determiner (e.g. little, much)
DA2         plural after-determiner (e.g. few, several, many)
DAR         comparative after-determiner (e.g. more, less, fewer)
DAT         superlative after-determiner (e.g. most, least, fewest)
DB         before determiner or pre-determiner capable of pronominal function (all, half)
DB2         plural before-determiner ( both)
DD         determiner (capable of pronominal function) (e.g any, some)
DD1         singular determiner (e.g. this, that, another)
DD2         plural determiner ( these,those)
DDQ         wh-determiner (which, what)
DDQGE         wh-determiner, genitive (whose)
DDQV         wh-ever determiner, (whichever, whatever)
EX         existential there
FO         formula
FU         unclassified word
FW         foreign word
GE         germanic genitive marker - (' or's)
IF         for (as preposition)
II         general preposition
IO         of (as preposition)
IW         with, without (as prepositions)
JJ         general adjective
JJR         general comparative adjective (e.g. older, better, stronger)
JJT         general superlative adjective (e.g. oldest, best, strongest)
JK         catenative adjective (able in be able to, willing in be willing to)
MC         cardinal number,neutral for number (two, three..)
MC1         singular cardinal number (one)
MC2         plural cardinal number (e.g. sixes, sevens)
MCGE         genitive cardinal number, neutral for number (two's, 100's)
```

```
MCMC        hyphenated number (40-50, 1770-1827)
MD         ordinal number (e.g. first, second, next, last)
MF         fraction,neutral for number (e.g. quarters, two-thirds)
ND1         singular noun of direction (e.g. north, southeast)
NN         common noun, neutral for number (e.g. sheep, cod, headquarters)
NN1         singular common noun (e.g. book, girl)
NN2         plural common noun (e.g. books, girls)
NNA         following noun of title (e.g. M.A.)
NNB         preceding noun of title (e.g. Mr., Prof.)
NNL1         singular locative noun (e.g. Island, Street)
NNL2         plural locative noun (e.g. Islands, Streets)
NNO         numeral noun, neutral for number (e.g. dozen, hundred)
NNO2         numeral noun, plural (e.g. hundreds, thousands)
NNT1         temporal noun, singular (e.g. day, week, year)
NNT2         temporal noun, plural (e.g. days, weeks, years)
NNU         unit of measurement, neutral for number (e.g. in, cc)
NNU1         singular unit of measurement (e.g. inch, centimetre)
NNU2         plural unit of measurement (e.g. ins., feet)
NP         proper noun, neutral for number (e.g. IBM, Andes)
NP1         singular proper noun (e.g. London, Jane, Frederick)
NP2         plural proper noun (e.g. Browns, Reagans, Koreas)
NPD1         singular weekday noun (e.g. Sunday)
NPD2         plural weekday noun (e.g. Sundays)
NPM1         singular month noun (e.g. October)
NPM2         plural month noun (e.g. Octobers)
PN         indefinite pronoun, neutral for number (none)
PN1         indefinite pronoun, singular (e.g. anyone, everything, nobody, one)
PNQO         objective wh-pronoun (whom)
PNQS         subjective wh-pronoun (who)
PNQV         wh-ever pronoun (whoever)
PNX1         reflexive indefinite pronoun (oneself)
PPGE         nominal possessive personal pronoun (e.g. mine, yours)
PPH1         3rd person sing. neuter personal pronoun (it)
PPHO1         3rd person sing. objective personal pronoun (him, her)
PPHO2         3rd person plural objective personal pronoun (them)
PPHS1         3rd person sing. subjective personal pronoun (he, she)
PPHS2         3rd person plural subjective personal pronoun (they)
PPIO1         1st person sing. objective personal pronoun (me)
PPIO2         1st person plural objective personal pronoun (us)
PPIS1         1st person sing. subjective personal pronoun (I)
PPIS2         1st person plural subjective personal pronoun (we)
PPX1         singular reflexive personal pronoun (e.g. yourself, itself)
PPX2         plural reflexive personal pronoun (e.g. yourselves, themselves)
PPY         2nd person personal pronoun (you)
RA         adverb, after nominal head (e.g. else, galore)
REX         adverb introducing appositional constructions (namely, e.g.)
RG         degree adverb (very, so, too)
RGQ         wh- degree adverb (how)
RGQV         wh-ever degree adverb (however)
RGR         comparative degree adverb (more, less)
RGT         superlative degree adverb (most, least)
RL         locative adverb (e.g. alongside, forward)
RP         prep. adverb, particle (e.g about, in)
RPK         prep. adv., catenative (about in be about to)
RR         general adverb
RRQ         wh- general adverb (where, when, why, how)
RRQV         wh-ever general adverb (wherever, whenever)
RRR         comparative general adverb (e.g. better, longer)
RRT         superlative general adverb (e.g. best, longest)
RT         quasi-nominal adverb of time (e.g. now, tomorrow)
TO         infinitive marker (to)
UH         interjection (e.g. oh, yes, um)
VB0         be, base form (finite i.e. imperative, subjunctive)
VBDR         were
VBDZ         was
VBG         being
VBI         be, infinitive (To be or not... It will be ..)
VBM         am
VBN         been
VBR         are
```

```
VBZ        is
VD0        do, base form (finite)
VDD        did
VDG        doing
VDI        do, infinitive (I may do... To do...)
VDN        done
VDZ        does
VH0        have, base form (finite)
VHD        had (past tense)
VHG        having
VHI        have, infinitive
VHN        had (past participle)
VHZ        has
VM        modal auxiliary (can, will, would, etc.)
VMK        modal catenative (ought, used)
VV0        base form of lexical verb (e.g. give, work)
VVD        past tense of lexical verb (e.g. gave, worked)
VVG        -ing participle of lexical verb (e.g. giving, working)
VVGK        -ing participle catenative (going in be going to)
VVI        infinitive (e.g. to give... It will work...)
VVN        past participle of lexical verb (e.g. given, worked)
VVNK        past participle catenative (e.g. bound in be bound to)
VVZ        -s form of lexical verb (e.g. gives, works)
XX        not, n't
ZZ1        singular letter of the alphabet (e.g. A,b)
ZZ2        plural letter of the alphabet (e.g. A's, b's)


Also allowed:
- Ditto tags (TAGxy), for example II31, II32, II33.
- Punctuation tags used in this project: , . ; : ? ! ( ) -

Important constraints:
- Do NOT invent tags.
- If uncertain, choose the closest valid tag from the allowed list; never output an out-of-list label.
- Use "/" token as FO unless context clearly requires another allowed tag.
- Use IF for "for" as preposition; use CS when "for" is a conjunction.
- Use TO for infinitive marker "to"; use II for "to" as preposition.
- Keep pronoun case tags consistent (subject vs object).
- Use NNB for title words before names (Mr., Dr., Prof.); do not overuse NNB for ordinary nouns.

Now tag this text:

{input}

Output results in this format:

TAG TABLE
| token          | tag                 | comment             |
|----------------|---------------------|---------------------|
| <input token>  | <tag from the list> | <comment if needed> |
| ...
```