Fine-grained Named-Entity Recognition for the East-India Company domain

Sophie Arnoult[1], Brecht Nijman[2], and Leon van Wissen[3]

[1] VU University, Amsterdam, The Netherlands
[2] Huygens Institute, Amsterdam, The Netherlands
[3] University of Amsterdam, Amsterdam, The Netherlands

Abstract

The Digital Humanities can nowadays benefit from easily accessible tools and pretrained models. Questions remain about the adequation between the data used to train these models and the task data. For a task like Named-Entity Recognition, domain specificity expresses itself not only in the linguistic domain but also in the entities of interest. While fine-grained entity tagsets are valuable, they are harder to annotate, leading to smaller, less representative training data, and may also be less interoperable with other NER label sets. In this work, we introduce a new fine-grained NER dataset for early modern Dutch texts related to the Dutch East India Company, covering 15 NER tags and 8000 mentions. We show that training a language model on the task data improves NER performance compared to off-the-shelf multilingual pretrained models. We further introduce a new method, class-agnostic co-training, to augment training data with existing NER datasets from the same domain, but with more restricted tagsets. We demonstrate that this method improves performance for augmented tags while increasing overall precision. Our annotations and code are publicly available.[1]

Keywords: Dutch historical domain, named-entity recognition, pretrained language models, data augmentation

1 Introduction

Named Entity Recognition (NER) is a well-understood task, but one that is difficult to apply, as it is domain sensitive, both in terms of source text and of target labels. Tools such as spaCy [9] and GLINER [29], while they lower the barrier for non-specialists to apply machine learning techniques, are of limited practical applicability to historical and literary sources, as they are trained on modern news sources, and only provide generic labels.

Pretrained language models [4; 21] form a better starting point for domain-specific applications, as transformer encoders [24] can notably be finetuned to predict NER classes at the token and subtoken levels. For the historical domain, one can directly apply multilingual models such as mBERT [4] or XLM-RoBERTa [3], which have been shown to generalise well to other languages for various tasks [28], or models trained on historical data [17; 18]. Resources permitting, one can adapt language models to the source domain by continuing pretraining [8], or directly training a model on the task data [7].

However, finetuning pretrained language models still requires annotated data for training. This in turn requires domain-trained annotators and time and effort to refine guidelines and produce annotations. Refining tagsets for specialised domains increases the task's complexity, reducing

Sophie Arnoult, Brecht Nijman, and Leon van Wissen. "Fine-grained Named-Entity Recognition for the East-India Company domain." In: *Computational Humanities Research 2025*, ed. by Taylor Arnold, Margherita Fantoli, and Ruben Ros. Vol. 3. Anthology of Computers and the Humanities. 2025, 917–931. https://doi.org/10.63744/DRbhWNTzqNzR.

[1] `https://github.com/globalise-huygens/finegrained-hist-ner`. A copy of the repository is deposited on Zenodo: `https://doi.org/10.5281/zenodo.17457075`

the final amount of annotated data for a given time budget. Although data augmentation techniques [10; 27; 30] can be used to increase training data, one would also want to reuse annotations beyond their application within a given project. In this work, we consider how to augment fine-grained training data with existing NER data from the same domain but with a more restricted tagset. We propose to co-train a NER model with external data by adapting the loss computation to make the model agnostic to task-specific tags when facing sequences from the external data. We call this method class-agnostic co-training. We show that this method performs well, improving model performance on augmented NER tags, while preserving the performance on task-specific tags.

The domain we consider in this work is that of texts produced by the Dutch East India Company (VOC) in the seventeenth and eighteenth century. Specifically, we work with the collection of the *Overgekomen Brieven en Papieren* (Letters and Papers Received, OBP) of the VOC [6], which consists of nearly 4.8M scans of hand-written text processed with Loghi [13]. The linguistic characteristics of these documents differ from those of contemporary Dutch in the sense that orthographic conventions and grammar were not yet standardised, sentence structures were often long and convoluted, and the meaning of words could shift depending on historical and colonial context. Nevertheless, the corpus's discourse is relatively consistent. Originating from a single administrative entity (the VOC), the documents revolve around a recurring set of administrative and commercial topics, including trade, logistics, personnel, finances, and enslavement.

The standard NER categories (among others, Person, Location, and Organization) are insufficient to capture the full range of information relevant to the VOC archive and colonial discourse more broadly. To adequately support research on trade, governance, and daily life in the early modern colonial world, a historical NER model must also be able to identify entities such as commodities, units of measurement, and ships, as well as finer-grained personal identifiers such as status or profession. This expansion of entity types contributes towards a better understanding of such corpora, making it easier for researchers to contextualise and reconstruct semantic and institutional structures embedded in the text, besides offering a useful aid on the surface level for terms and references once linked to knowledge bases and vocabularies. In this paper, we present the annotation process leading to a new NER dataset with 15 entity labels and close to 8000 entity mentions for early modern Dutch and the Dutch East-India Company domain.

Multilingual pretrained encoder models form a good basis for historical Dutch [2; 14; 20] and serve as baselines in this work. We compare models of different sizes to trade off performance and computational cost: while larger models are expected to perform better, they would lead to higher inference costs on the full OBP corpus. In addition, as [8] point to the benefit of adapting language models to the target domain and task data, we also experiment with a model trained on the OBP corpus, gloBERTise [26]. We find that finetuning this model for NER outperforms larger models such as multilingual BERT-base [4] and XLM-Roberta-Base [3], being competitive with the 4-times larger XLM-Roberta-Large.

There exist several NER datasets for the VOC data or Dutch historical texts of the same period [2; 14; 20] that could serve as augmentation data. Leveraging other datasets is difficult in practice, as datasets differ in the size of the label set and in the definition of entity types. When datasets are not built from the same source, this also leads to differences in context for entities [5]. We expect that the datasets of [2; 20], which both overlap with the OBP corpus and have annotation guidelines closer to [14], are close enough in terms of the source domain and that, therefore, their tagset can be seen as a subset of ours. We focus on the *voc-gm-ner* dataset of [2], which uses 6 of our 15 tags. We present data-augmentation experiments where we compare the effect of adding task-internal training data to adding data from the *voc-gm-ner* dataset. We show that our class-agnostic co-training method yields improvements even as more task-internal data become available.

2 Annotations

2.1 Data selection

The corpus consists of a subseries within the VOC-archive, the OBP. This series consists of documents in a variety of genres, including but not limited to letters, court cases, and cargo lists. We use the HTR-transcriptions of the corpus published by the GLOBALISE project [6]. We selected 26 documents for annotation, spanning from 1618 to 1782. During selection, HTR quality was taken into account. We selected documents where the layout recognition had performed well, meaning that headers, paragraphs, and marginalia were clearly separated in the resulting machine-readable text. The overall quality of the HTR was deemed very high, suitable for manual reading and automatic processing.

2.2 Annotation task

The label set for this annotation task consists of fifteen labels describing seven larger entity types: *persons*, *locations*, *organizations*, *polities*, *commodities*, *ships*, and *documents*, as well as *dates* (see Table 1 for an overview including descriptions of each entity type). This label set was developed in collaboration with domain experts. The reasons to extend the label set beyond the more common categories of *Person*, *Location*, and *Organisation* are twofold. First, the set of entities considered is expanded to include other entities of historical significance, such as documents, commodities, and ships. The labels `DOC`, `CMTY_NAME`, `CMTY_QUAL`, `CMTY_QUANT`, and `SHIP` were added to cover these additional entities. Second, additional labels are added to cover unnamed instances of various entities, particularly persons. This is done in an effort to compensate for the colonial imbalance in the corpus, since colonial subjects are less likely to be mentioned by name and are instead often referred to by their status, title, or profession [16]. The labels `PRF`, `STATUS`, `PER_ATTR`, and `ETH_REL` can cover such unnamed instances of people in the corpus. In doing so, they provide an additional entrypoint for researchers to locate these individuals.

The labels are mutually exclusive, any span can only be annotated with a single label. Compositional references are split into sequences, see examples (1) and (2).

(1) [Mousabeeck]$_{\text{PER\_NAME}}$, [Ambassadeur]$_{\text{PRF}}$ den [Conincks]$_{\text{PRF}}$ van [Persia]$_{\text{LOC\_NAME}}$[2]

(2) een [deens]$_{\text{LOC\_ADJ}}$ [comp„s]$_{\text{ORG}}$ [scheepje]$_{\text{SHIP\_TYPE}}$[3]

2.3 Annotation process

The INCEpTION platform was used for annotation [12]. All annotations were made by historians with prior familiarity with the corpus or similar corpora. Analysis of an initial pilot annotation round of annotations with four annotators led to a reduction in entity labels from 21 to 15. This was achieved by reducing the number of fine-grained person related labels by combining them, as well as removing the distinction between *locations* and *polities* (similar to GPE). The results of this pilot annotation round are not included in the subsequent analysis of this paper. In all subsequent annotation rounds annotators worked in pairs, allowing annotators to discuss difficult cases and cross-check each others work within the annotator pairs.

These changes, as well as further clarifications in the annotation guidelines[4] led to improvement in label agreement from about 80% to 90%, even with the introduction of new annotators. Disagreement persists in cases where distinction between labels requires specific domain knowledge and cannot be derived from linguistic context alone. For instance, in the following example

[2] Musa Beg, Ambassador to the King of Persia

[3] a small Danish comp[any] ship

[4] Guidelines are included in the data and code repository, https://github.com/globalise-huygens/finegrained-hist-ner

Table 1: Label set with corresponding entities

NER label	Description	Related entities
PER_NAME	Name of person	persons
PRF	Profession, title	persons
STATUS	(Civic) status	persons
PER_ATTR	Person attributes (other than PRF or STATUS)	persons
LOC_NAME	Name of Location	locations, polities
LOC_ADJ	Derived form of Location name	persons, any (through qualification)
ETH_REL	Ethnic, religious or ethno-religious appellation, not derived from location name	persons, any (through qualification)
CMTY_NAME	Name of Commodity	commodities
CMTY_QUAL	Commodity qualifier: colors, processing	commodities
CMTY_QUANT	Quantity	commodities
SHIP	Ship name	ships
SHIP_TYPE	Ship type	ships
ORG	Organisation	organisations, polities
DATE	Date	dates
DOC	Document	documents

(3) *Samarijnsche* refers to *Zamorin*, the title of the ruler of Calicut (label `PRF`). However, if this is not known, it can easily be confused with an adjectival form of a location (label `LOC_ADJ`). Similarly, the `ETH_REL` and `LOC_ADJ` labels can be easily confused. See examples (4) and (5): *Tidoorsche* is a derived form of Tidore, an island and polity in what is now eastern Indonesia; *Alfoerese*, does not refer to a place but to a colonially applied ethnic category.

(3) Het geheele Samarijnsche land[5]

(4) Tidoorsche grooten

(5) Alfoerese grooten

In order to reduce such confusion, annotators have access to glossaries and reference data relevant to the corpus. Additionally, all documents are checked by a designated curator for final quality control.

3 From annotations to training data

3.1 Preprocessing

Data are pretokenised with spaCy (`nl-core-news-lg` model) [9] prior to annotating. For training, as the model does not allow for a reliable sectioning of text into sentences, due to the irregular use of punctuation in these historical texts, long passages are segmented into sequences by their number of tokens. We found that a maximum length of 240 tokens was enough to ensure that the data would remain within a length of 512 subtokens for the XLM-Roberta tokenizer. As all tested pretrained models have a smaller or same-sized vocabulary, we assume that this limit is adequate for them too. For simplicity's sake, the data are preprocessed with this maximum token length for all data and all pretrained-model tokenizers. Finally, span-level annotations are converted to IOB token-level labels [23].

[5] the entire Zamorin land

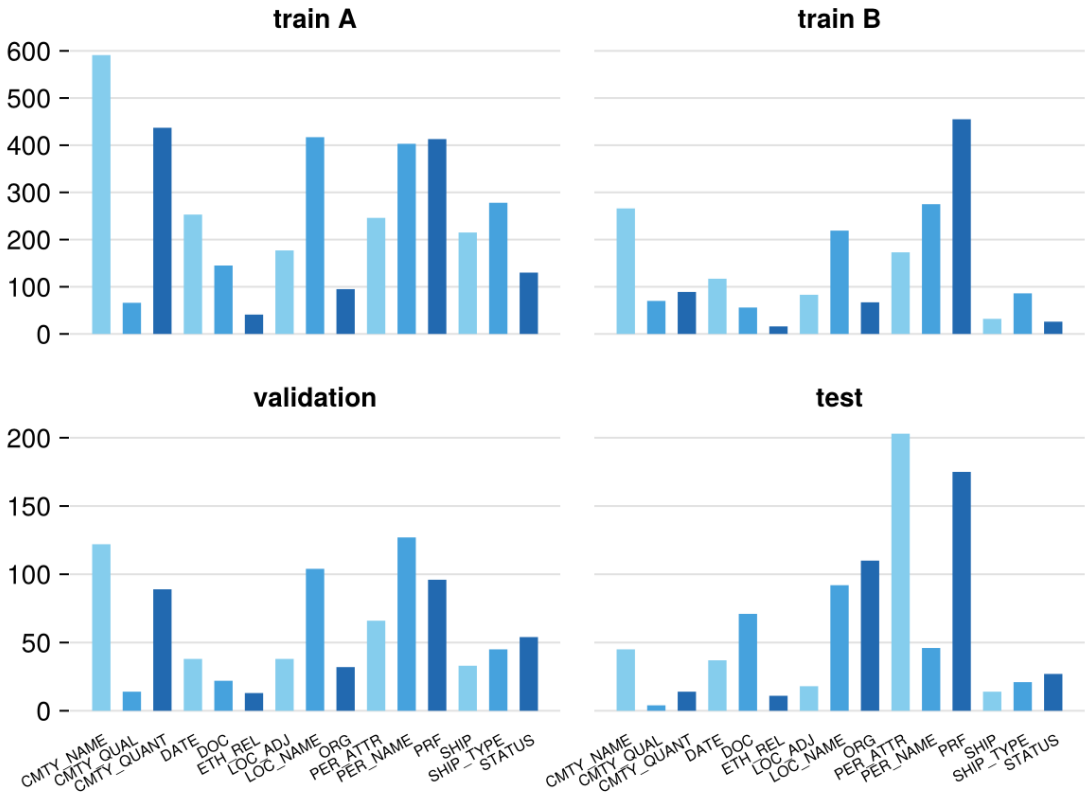

Figure 1: Entity distribution in data splits. The first training set *A* and the validation data have similar distributions as they are sampled by sequence from the same first set of annotations and so as to obtain median performance on the validation set. The *B* training set and the test set, which are taken from a second set of annotations, are more dissimilar as they come from different documents, containing less commodity-related annotations and more person-related annotations; their distributions are also different from each other as they were split by document.

3.2 Data splits

The annotated data were produced and curated in successive rounds, and resulted into two sets of annotations for this work: the first set of annotations was used for early experiments and validating models, while the second set of annotations was used for evaluating the effect of internal data augmentation and for testing.

The models were trained and validated on the first set of annotations. To allow for a representative validation set within these data, we shuffled sequences in the data; folded them into 5 datasets with an 80%-20% training/validation split; finetuned XLM-Roberta-Base on the five resulting datasets; and finally selected the split that led to median performance. The resulting data subsets are referred to as `train-A` and `validation` henceforth.

The second set of annotations was used to provide additional training data and a heldout test set. These are split by document to minimise possible overlap between training and test data. We selected three documents for testing based on their size and entity distribution, aiming at a balanced distribution and a size comparable to that of the validation data. The resulting splits are referred to as `train-B` and `test` in what follows.

Entity type distributions in the resulting training, validation, and test sets are shown in Figure 1.

3.3 Mapping entities to subtokens

Subtoken IOB tags are derived from token-level IOB labels as if by applying the IOB scheme directly to span-level entity annotations: the *first subtoken* of the *first token* of an entity of type X is labelled as B-X, and all other subtokens in the entity's span are labelled as I-X. Subtokens of O-labelled tokens are tagged as O. For instance, the sequence *Sijbrandt Harmansz van Giever* is labelled as follows at the span level (annotations), token level (after tokenization by spaCy) and subtoken levels (after tokenization by gloBERTise, see section 4):

span [Sijbrandt Harmansz]$_{\text{PER\_NAME}}$ van [Giever]$_{\text{LOC\_NAME}}$

token [Sijbrandt]$_{\text{B-PER\_NAME}}$ [Harmansz]$_{\text{I-PER\_NAME}}$ [van]$_{O}$ [Giever]$_{\text{B-LOC\_NAME}}$

subtoken [Sij]$_{\text{B-PER\_NAME}}$ [_brandt]$_{\text{I-PER\_NAME}}$ [Harmansz]$_{\text{I-PER\_NAME}}$ [van]$_{O}$ [Gie]$_{\text{B-LOC\_NAME}}$ [_ver]$_{\text{I-LOC\_NAME}}$

Past training, subtoken predictions are reconstructed into span-level predictions following the reverse operation: the predictions of token-initial subtokens are mapped to token-level predictions, and these IOB tags are mapped to span predictions following a non-strict scheme: entities are identified in principle by B-tags optionally followed by sequences of I-tags of the same type; I-tags that follow on an O-tag are considered as also marking the start of an entity and reinterpreted as B-tags.

3.4 Metrics

Training validation uses *token-level* F1 scores (micro, macro and weighted) as metrics, whereby O-class true positives are ignored so as to prevent them from weighing out scores. F1 scores are computed on token-initial subtokens only, to reflect final span-level computations more closely[6]. Consequently, F1 scores are computed only on token-initial subtoken predictions for which at least the predicted or the true label is not O.

Testing is performed at the entity-span level with *seqeval* [19], using the non-strict IOB2 scheme.

4 Finetuning pretrained language models for NER

4.1 Pretrained models

All models are based on pretrained transformer-encoder language models supplemented with a token classification layer. We experimented with four multilingual models: multilingual BERT-base [4], its distilled variant [22], and the base and large variants of XLM-Roberta [3]; we also experimented with a pretrained language model, gloBERTise [25; 26], trained directly on the OBP data, and based on RoBERTa [15]. We did not experiment with Gysbert [18]: while its training data covers the historical period of the VOC, it includes more out-of-domain data through literary texts. Therefore, we expected that it would perform less well than gloBERTise.

Models vary in the size of the tokenizer vocabulary, the number of attention layers, and hidden dimensions as reported in Table 2.

4.2 Experimental setup

Models are optimised with Adam [11], using default parameters and adapting only the learning rate. For XLM-Roberta-Large, the initial learning rate is $2e^{-5}$ for a batch size is 16; for the other models, the initial learning rate is $4e^{-5}$ for a batch size of 32. The learning rate is halved every

[6] In contrast, loss is computed at the subtoken level.

Table 2: Pretrained-model parameters and sizes

model	vocabulary (k)	layers	dimension	parameters (M)		
				embeddings	attention	total
gloBERTise	50	12	768	40	85	125
dist-mBERT	120	6	768	92	43	135
mBERT	120	12	768	92	85	177
XLMR-base	250	12	768	192	85	277
XLMR-large	250	24	1024	257	302	558

time the token-level micro-F1 fails to increase for more than 3 epochs. All models are trained for up to 40 epochs.

All experiments use the same single seed, except for model validation, where experiments with two more seeds are added. For model validation, evaluation is based on the best average F1 (micro, macro and weighted, respectively) over three seeds within 40 epochs. For testing, the initial seed is used, conducting an evaluation on the checkpoint with the best micro-F1 within 40 epochs.

4.3 Model validation

Table 3 reports model performance based on the underlying pretrained language model, and averaged over three seeds, selecting the epoch with the best average micro F1 for each model. Model performance increases overall with model size, except for gloBERTise, which performs comparably to the four-times larger XLM-Roberta-Large. We see here the effect of the gloBERTise vocabulary being fit to the OBP data; this effect is also reflected by faster model convergence, with the best results being reached after 15 epochs, against 29 to 40 for the other models.

Table 3: Validation scores for models trained on the train-A set, averaged over 3 seeds. The best average metric scores are reported along with the epoch at which they were measured and with standard deviation.

	micro F1			macro F1			weighted F1		
model	epoch	F1	std	epoch	F1	std	epoch	F1	std
gloBERTise	15	**81.8**	0.3	29	70.7	1.0	12	**82.5**	0.44
dist-mBERT	40	72.3	0.31	40	65.5	1.33	26	73.4	0.62
mBERT	29	77.6	0.83	29	69.5	3.56	29	78.3	1.07
XLMR-base	40	77.1	0.44	40	69.8	0.78	34	78.4	1.19
XLMR-large	36	81.1	1.37	22	**72.1**	1.84	39	81.7	1.56

The advantage of gloBERTise for our case carries out to unseen data as shown by the test results in Table 4: even though the best XLM-Roberta-Large model performs better overall than the gloBERTise one on validation data, its performance is lower on the test data.

The good performance obtained with gloBERTise is encouraging, as it allows us to experiment with a lighter model, leading to decreasing development, training, and inference times for our NER model.

Table 4: Pretrained-model comparison by validation (token and seqeval) and test seqeval scores for best micro-F1 checkpoint (single seed).

		validation token F1			validation seqeval F1			test seqeval F1		
model	epoch	mic.	mac.	weigh.	mic.	mac.	weigh.	mic.	mac.	weigh.
gloBERTise	15	81.7	70.0	82.2	82.2	78.3	82.1	**66.6**	**60.8**	**64.2**
dist-mBERT	39	72.3	65.3	73.0	72.0	67.4	72.5	53.6	48.3	52.0
mBERT	29	78.5	71.9	79.4	79.5	75.0	79.4	57.4	49.0	54.5
XLMR-base	28	78.1	71.4	78.5	79.1	74.5	78.9	55.1	48.6	50.5
XLMR-large	37	**82.6**	**72.2**	**83.1**	**84.2**	**80.3**	**84.2**	65.9	59.1	62.8

5 Data augmentation with unknown classes

5.1 The voc-gm-ner corpus

The voc-gm-ner corpus [1; 2] is textually close to the OBP-data: it is taken in part from the *Generale Missiven* corpus, which is a subset of the OBP corpus; for the other part, it consists of 20th century Dutch editorial notes and comments on the historical text. While this second part is not included in the OBP corpus, it follows closely on it, notably mentioning entities that are relevant to the OBP. The data set also differs from the OBP data in that it was derived from digitised OCRed texts, whereas we work with HTR data.

When it comes to entity types, the version of the corpus provided for training (the *datasplit_all_standard* view of the corpus [1], which we refer to henceforth as *vocgm* for VOC Generale Missiven) uses a subset of our label set, corresponding to the following types: `ETH_REL`, `LOC_ADJ`, `LOC_NAME`, `ORG`, `PER_NAME` and `SHIP`. We assume at least that the entity types defined in the corpus are close enough to our own definitions to be mapped directly to these labels. This leaves us with nine entity types that are defined in our data, probably well represented in the *vocgm* data, but labelled as non-entities `O`.

As the *vocgm* dataset is larger than ours (with about 18.7k training entity mentions against 3.9k for train-A and 5.9k for the joint train-A and B sets), simply joining the training data would prevent correct learning of these types, as their predictions would be penalised against `O` reference labels in the *vocgm* data. To limit this effect, we experimented with both data selection and class-agnostic co-training.

5.2 Selecting data by entity density

We experimented with selecting sequences with a minimum entity density, i.e. proportion of entity tokens in a sequence. Enforcing a threshold on entity density decreases the size difference between our data and the augmentation data, and might decrease the representation of unseen entities. Note that this last hypothesis rests on an independence assumption of entity type occurrences, which is likely to hold better for some types than others: it seems likely enough for commodity listings, but not for letters where attributive types like `PER_ATTR` or `SHIP_TYPE` can be expected to occur closely to `PER_NAME` and `SHIP`, respectively.

We experimented with two thresholds, 0.05 and 0.1, selecting *vocgm* sequences with at least 5% and 10% of entity tokens.

5.3 Class-agnostic co-training

To counteract the effect of unseen types in the *vocgm* dataset on the learning of these types, we can consider that `O` annotations in the *vocgm* data are not truly representative of `O` classes, but of any

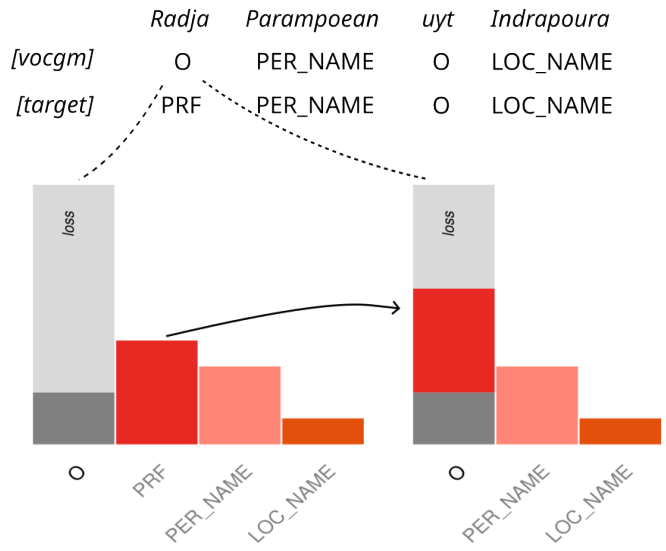

Figure 2: Class-agnostic co-training. Generic entity labels are represented for brevity, but they stand for corresponding B or I tags. The example *vocgm* training sequence contains a token, *Radja*, that is tagged as *O* whereas the target tag should be *PRF*. As cross-entropy loss for that token is computed against the model's output for the *O* class, we want to report the softmax weight of the target class *PRF* onto that of the *O* class to avoid penalizing true-class predictions. As the true class cannot be known for a given *vocgm O* class, one reports the softmax weights of *all vocgm*-unknown target classes to that of the *O* class for *vocgm* training sequences.

type in the union set of O and the *vocgm*-unknown 9 classes. To align the model with these reference data, one can then make it equally agnostic, by reporting the probability mass of *vocgm*-unknown types to the O class before computing the loss, as shown in Figure 2.

In practice, a mask of unknown classes is attached to *vocgm* sequences when adding them to the training data (sequences from our annotations and *vocgm* data are shuffled through each other in the training data, whereas the validation data consist only of OBP data). The output of the model for these sequences is passed through a softmax to obtain class probability estimates, and the mask is used to compute the sum of the unknown-class probability mass, which is then added to the softmax value of the O class. Subsequently, instead of directly computing cross-entropy loss over output logits for these sequences, one computes the negative log-likelihood loss over (the log of) the recomputed softmax values.

5.4 Model validation

Table 5 shows that class-agnostic co-training generally improves model performance. Entity-density selection has a small positive effect on its own, but micro F1 are lower than for the gloBERTise baseline trained on the A training dataset only. In contrast, models with class-agnostic co-training benefit from the addition of external data, and perform slightly better as more external data is included. We adopt class-agnostic co-training with the full *vocgm* as setting for the following external-data augmented experiments.

5.5 Data augmentation with task data and external data

As more annotations become available in a project, one can wonder if there is still benefit in augmenting data from external datasets. Validation experiments on task-internal data with the *train-B* set and on *vocgm*-augmented data show that both kinds of data are beneficial to model performance,

Table 5: Data augmentation of the *train-A* set with the *vocgm* training data. Effect of entity-density threshold (*edt*) and class-agnostic co-training (*cact*) with gloBERTise on validation scores. Training size refers to the number of entities.

		training	micro F1			macro F1			weighted F1		
edt	cact	size (k)	epoch	F1	std	epoch	F1	std	epoch	F1	std
train A		3.9	15	81.8	0.3	29	70.7	1.0	12	82.5	0.44
0	no	22.6	33	80.8	0.31	33	72.6	1.19	33	83.2	0.26
0.05	no	19.3	20	81.4	0.33	25	73.4	3.15	10	83.5	0.98
0.1	no	12.5	18	81.3	0.49	34	73.4	0.46	23	83.3	0.78
0	yes	22.6	6	**82.7**	0.27	11	**74.7**	1.55	7	**84.3**	0.79
0.05	yes	19.3	6	82.4	0.98	26	74.1	1.04	15	83.2	0.61
0.1	yes	12.5	10	82.5	0.62	24	74.4	1.85	34	83.5	0.69

both separately and when combined, as shown in Table 6. One can also observe that models trained on the B set only score poorly on the validation data, reflecting their difference in entity distribution with the `train-A` and validation sets.

Table 6: Data augmentation with task-data (*train-B*) and external data (*vocgm*). Finetuning scores on the train-A, train-B, joint A and B sets, and *vocgm*-augmented sets.

	micro F1			macro F1			weighted F1		
training data	epoch	F1	std	epoch	F1	std	epoch	F1	std
A	15	81.8	0.3	29	70.7	1.0	12	82.5	0.44
B	36	66.2	0.84	37	53.2	2.25	36	67.6	0.75
A+B	28	83.2	0.62	14	71.6	0.75	20	83.5	1.04
A+vocgm	6	82.7	0.27	11	74.7	1.55	7	84.3	0.79
A+B+vocgm	35	**84.3**	0.46	13	**76.1**	1.23	39	**84.8**	0.36

In contrast, test results, reported in Table 7, show that models co-trained on the `train-B` set perform relatively better on the test data, reflecting the somewhat closer distributions within the `train-B/test` set of annotations. Class-agnostic co-training with the *vocgm* data however still provides gains for micro and macro F1 test seqeval scores.

Table 7: Data augmentation: validation (token and seqeval) and test seqeval scores for best micro-F1 checkpoint (single seed). All checkpoints use the same initial seed, and are selected by (token) micro F1 scores.

		validation token F1			validation seqeval F1			test seqeval F1		
training data	ep.	mic.	mac.	weigh.	mic.	mac.	weigh.	mic.	mac.	weigh.
A	15	81.7	70.0	82.2	82.2	78.3	82.1	66.6	60.8	64.2
B	40	66.2	54.6	67.9	65.9	58.3	64.4	64.9	57.9	63.9
A+B	8	83.0	69.5	83.2	82.6	78.0	82.5	69.0	63.6	**67.2**
A+vocgm	15	82.8	**75.5**	**84.1**	85.3	81.6	85.3	67.9	63.4	63.3
A+B+vocgm	40	**83.1**	75.0	83.4	**85.8**	**82.4**	**85.8**	**69.7**	**64.7**	66.4

A comparison of precision and recall, shown in Table 8, shows that *vocgm* augmented models have higher overall precision but lower recall on the test data. The higher precision of *vocgm*-

Table 8: Data augmentation with task-data (*train-B*) and external data (*vocgm*): seqeval precision and recall. Best test scores for augmented data models are boldfaced.

	validation						test					
training	micro		macro		weight.		micro		macro		weight.	
data	P	R	P	R	P	R	P	R	P	R	P	R
A	80.8	83.5	79.9	78.6	81.2	83.5	72.2	61.9	65.2	60.9	74.8	61.9
B	65.3	66.5	61.1	58.8	65.4	66.5	63.7	66.3	57.0	60.5	63.4	66.3
A+B	83.1	84.8	80.9	81.4	83.6	84.8	74.5	**65.3**	69.2	**65.9**	74.8	**65.3**
A+vocgm	85.6	85.0	83.6	80.8	86.0	85.0	77.6	60.4	**74.4**	60.2	**80.5**	60.4
ABvocgm	85.1	86.6	83.7	82.4	85.4	86.6	**77.7**	63.1	71.6	62.8	79.3	63.1

augmented models can be attributed to the larger training data: as models are exposed to more data during training, even `O` instances contribute to penalizing false positives.

Table 9: Type level seqeval scores for the in-task (*A+B*) and external (*A+B+vocgm*) augmented training data. We separate types by their presence in the *vocgm* training data. Best F1 scores per type are boldfaced; underlined values highlight types for which the *A+B+vocgm* trained model has higher precision on unknown types and types for which the non-*vocgm*-augmented model has higher precision or recall on augmented types.

	train	A+B			train	A+B+vocgm			test
	support	prec.	recall	F1	support	prec.	recall	F1	supp.
CMTY_NAME	857	87.5	77.8	82.4	857	97.2	77.8	**86.4**	45
CMTY_QUAL	136	0.0	0.0	0.0	136	0.0	0.0	0.0	4
CMTY_QUANT	526	80.0	57.1	**66.7**	526	63.6	50.0	56.0	14
DATE	370	72.5	80.6	**76.3**	370	61.9	72.2	66.7	37
DOC	201	67.9	51.4	**58.5**	201	73.8	44.3	55.4	71
PER_ATTR	417	79.3	34.0	**47.6**	417	85.5	23.2	36.4	203
PRF	866	70.7	80.0	75.1	866	70.6	81.1	**75.5**	175
SHIP_TYPE	364	83.3	95.2	**88.9**	364	86.4	90.5	88.4	21
STATUS	156	64.5	74.1	69.0	156	69.0	74.1	**71.4**	27
ETH_REL	57	33.3	9.1	14.3	262	75.0	27.3	**40.0**	11
LOC_ADJ	260	46.2	66.7	54.5	3105	58.3	77.8	**66.7**	18
LOC_NAME	636	88.6	84.8	86.7	9173	91.4	92.4	**91.9**	92
ORG	162	70.2	72.7	71.4	1760	85.7	76.4	**80.8**	110
PER_NAME	678	68.8	71.7	70.2	4700	71.4	76.1	**73.7**	46
SHIP	247	100.0	85.7	**92.3**	1730	84.6	78.6	81.5	14

In some cases, namely `CMTY_NAME`, `DOC`, `PER_ATTR`, `SHIP_TYPE`, and `STATUS`, this leads to a higher precision of the *A+B+vocgm* model on *vocgm*-unknown types, as shown in Table 9. The lower test recall with the *A+B+vocgm* model results mostly from the combination of poor recall for the `PER_ATTR` class and the weight of that class in the text data, which also explains the lower overall weighted F1 score in Table 7. The low recall for this class is the combined result of under-representation in the training data and over-representation in the test data (see Figure 1). Whereas

performance for this class should improve from retraining a NER model on all the data, this result shows how sensitive the less represented types are to differences between training and unseen data.

The type-level scores in Table 9 also show that the decrease in recall with the external *vocgm* data is mostly related to *vocgm*-unknown classes, reflecting the fact that these classes become less well represented in the augmented training data.

In contrast, the *A+B+vocgm* model exhibits higher F1 scores for shared classes, with the notable exception of the `SHIP` class. Although the results for this class are subject to variance given its poor representation in the test data, we note that there is no direct overlap of the test instances with the rest of the data. As about 75% of the *vocgm* `SHIP` instances come from editorial notes rather than historical text [2], we suspect that the lower score for this class may come from textual differences between the OBP and the editorial notes of the *voc-gm-ner* corpus, which may be further aggravated by tokenization, as gloBERTise is trained on the `OBP` and not on the *vocgm* data. It would be interesting to retrain a model using only the historical part of the *voc-gm-ner* corpus in this respect.

6 Conclusion

We have presented a fine-grained NER dataset for VOC-related Dutch historical texts, covering 8000 mentions of 15 entity types for the VOC domain. NER modelling experiments on this dataset show the benefit of adapting the language model to the task's data, leading to a lightweight, performing model. We propose a simple technique, class-agnostic co-training, to leverage datasets from similar sources but with more restricted tagsets for data augmentation. We have shown that this method increases overall performance for the NER tags that are shared with the in-task data, while generally increasing precision for unknown tags. We hope that this method will encourage the reuse of existing datasets in specialised domains.

Acknowledgements

This work was undertaken as part of the GLOBALISE project[7]. The overarching goal of GLOBALISE is to enrich VOC materials with structured, semantic annotations that aid users in reading, interpreting, and contextualizing the historical records in the vast corpus of texts created by the Company as kept in the National Archive of the Netherlands. By layering this information on top of the original texts and making it (re)searchable, the project aims to present the archival material in a digital infrastructure that bridges the temporal, linguistic, and cultural distance between then and now.

We thank SURF (www.surf.nl) for the support in using the Dutch National Supercomputer Snellius, and the anonymous reviewers for their constructive comments.

References

[1] Arnoult, Sophie. "VOC GM NER corpus". en. Datapackage. Publisher: Vrije Universiteit Amsterdam. Dec. 2022. DOI: 10.48338/VU01-HI67KL. URL: https://publication.yoda.vu.nl/full/VU01/HI67KL.html.

[7] Funded by the Dutch Research Council, https://globalise.huygens.knaw.nl/

[2] Arnoult, Sophie I., Petram, Lodewijk, and Vossen, Piek. “Batavia asked for advice. Pretrained language models for Named Entity Recognition in historical texts.” In: *Proceedings of the 5th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, ed. by Stefania Degaetano-Ortlieb, Anna Kazantseva, Nils Reiter, and Stan Szpakowicz. Punta Cana, Dominican Republic (online): Association for Computational Linguistics, Nov. 2021, pp. 21–30. DOI: 10.18653/v1/2021.latechclfl-1.3. URL: https://aclanthology.org/2021.latechclfl-1.3/.

[3] Conneau, Alexis, Khandelwal, Kartikay, Goyal, Naman, Chaudhary, Vishrav, Wenzek, Guillaume, Guzmán, Francisco, Grave, Edouard, Ott, Myle, Zettlemoyer, Luke, and Stoyanov, Veselin. “Unsupervised Cross-lingual Representation Learning at Scale”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault. Online: Association for Computational Linguistics, July 2020, pp. 8440–8451. DOI: 10.18653/v1/2020.acl-main.747. URL: https://aclanthology.org/2020.acl-main.747/.

[4] Devlin, Jacob, Chang, Ming-Wei, Lee, Kenton, and Toutanova, Kristina. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. arXiv:1810.04805 [cs]. May 2019. DOI: 10.48550/arXiv.1810.04805. URL: http://arxiv.org/abs/1810.04805.

[5] Ghosh, Sreyan, Tyagi, Utkarsh, Suri, Manan, Kumar, Sonal, S, Ramaneswaran, and Manocha, Dinesh. “ACLM: A Selective-Denoising based Generative Data Augmentation Approach for Low-Resource Complex NER”. en. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Toronto, Canada: Association for Computational Linguistics, 2023. DOI: 10.18653/v1/2023.acl-long.8. URL: https://aclanthology.org/2023.acl-long.8.

[6] GLOBALISE Project. “VOC transcriptions v2 - GLOBALISE”. 2024. URL: https://hdl.handle.net/10622/LVXSBW.

[7] Gu, Yu, Tinn, Robert, Cheng, Hao, Lucas, Michael, Usuyama, Naoto, Liu, Xiaodong, Naumann, Tristan, Gao, Jianfeng, and Poon, Hoifung. “Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing”. In: *ACM Transactions on Computing for Healthcare 3*, no. 1 (Oct. 2020), pp. 1–23. ISSN: 2637-8051. DOI: 10.1145/3458754. URL: http://dx.doi.org/10.1145/3458754.

[8] Gururangan, Suchin, Marasović, Ana, Swayamdipta, Swabha, Lo, Kyle, Beltagy, Iz, Downey, Doug, and Smith, Noah A. “Don’t Stop Pretraining: Adapt Language Models to Domains and Tasks”. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault. Online: Association for Computational Linguistics, July 2020, pp. 8342–8360. DOI: 10.18653/v1/2020.acl-main.740. URL: https://aclanthology.org/2020.acl-main.740/.

[9] Honnibal, Matthew, Montani, Ines, Van Landeghem, Sofie, and Boyd, Adriane. “spaCy: Industrial-strength Natural Language Processing in Python”. In: (2020). DOI: 10.5281/zenodo.1212303.

[10] Hu, Xuming, Jiang, Yong, Liu, Aiwei, Huang, Zhongqiang, Xie, Pengjun, Huang, Fei, Wen, Lijie, and Yu, Philip S. “Entity-to-Text based Data Augmentation for various Named Entity Recognition Tasks”. In: *Findings of the Association for Computational Linguistics: ACL 2023*, ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada: Association for Computational Linguistics, July 2023, pp. 9072–9087. DOI: 10.18653/

v1/2023.findings-acl.578. URL: https://aclanthology.org/2023.findings-acl.578/.

[11] Kingma, Diederik P. and Ba, Jimmy. “Adam: A Method for Stochastic Optimization”. 2014. arXiv: 1412.6980 [cs.LG]. URL: https://arxiv.org/abs/1412.6980.

[12] Klie, Jan-Christoph, Bugert, Michael, Boullosa, Beto, Eckart de Castilho, Richard, and Gurevych, Iryna. “The INCEpTION Platform: Machine-Assisted and Knowledge-Oriented Interactive Annotation”. In: *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, ed. by Dongyan Zhao. Santa Fe, New Mexico: Association for Computational Linguistics, Aug. 2018, pp. 5–9. URL: https://aclanthology.org/C18-2002/ (visited on 07/19/2025).

[13] Koert, Rutger van, Klut, Stefan, Koornstra, Tim, Maas, Martijn, and Peters, Luke. “Loghi: An End-to-End Framework for Making Historical Documents Machine-Readable”. en. In: *Document Analysis and Recognition – ICDAR 2024 Workshops*, ed. by Harold Mouchère and Anna Zhu. Cham: Springer Nature Switzerland, 2024, pp. 73–88. ISBN: 978-3-031-70645-5. DOI: 10.1007/978-3-031-70645-5_6.

[14] Koolen, Marijn, Renkema, Esger, Groskamp, Nienke, Smit, Frank, Reinders, Jirsi, Sluijter, R.G.H., Hoekstra, Rik, and Oddens, Joris. “Accessing the Republic: Digital Humanities in the Benelux 2024 Conference”. In: 2024. DOI: 10.5281/zenodo.11485227.

[15] Liu, Yinhan, Ott, Myle, Goyal, Naman, Du, Jingfei, Joshi, Mandar, Chen, Danqi, Levy, Omer, Lewis, Mike, Zettlemoyer, Luke, and Stoyanov, Veselin. “RoBERTa: A Robustly Optimized BERT Pretraining Approach”. arXiv:1907.11692 [cs]. July 2019. DOI: 10.48550/arXiv.1907.11692. URL: http://arxiv.org/abs/1907.11692.

[16] Luthra, Mrinalini, Todorov, Konstantin, Jeurgens, Charles, and Colavizza, Giovanni. “Unsilencing colonial archives via automated entity recognition”. In: *Journal of Documentation* (2023). ISSN: 0022-0418. DOI: 10.1108/JD-02-2022-0038. URL: https://doi.org/10.1108/JD-02-2022-0038 (visited on 07/13/2023).

[17] Manjavacas, Enrique and Fonteyn, Lauren. “Adapting vs. Pre-training Language Models for Historical Languages”. English. In: *Journal of Data Mining & Digital Humanities NLP4DH*, 3 (June 2022): *Digital humanities in languages*. ISSN: 2416-5999. DOI: 10.46298/jdmdh.9152. URL: https://jdmdh.episciences.org/9152.

[18] Manjavacas Arevalo, Enrique and Fonteyn, Lauren. “Non-Parametric Word Sense Disambiguation for Historical Languages”. In: *Proceedings of the 2nd International Workshop on Natural Language Processing for Digital Humanities*, ed. by Mika Hämäläinen, Khalid Alnajjar, Niko Partanen, and Jack Rueter. Taipei, Taiwan: Association for Computational Linguistics, Nov. 2022, pp. 123–134. DOI: 10.18653/v1/2022.nlp4dh-1.16. URL: https://aclanthology.org/2022.nlp4dh-1.16/.

[19] Nakayama, Hiroki. “seqeval: A Python framework for sequence labeling evaluation”. Software available from https://github.com/chakki-works/seqeval. 2018. URL: https://github.com/chakki-works/seqeval.

[20] Provatorova, Vera, Erp, Marieke van, and Kanoulas, Evangelos. “Too Young to NER: Improving Entity Recognition on Dutch Historical Documents”. In: *Proceedings of the Third Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA) @ LREC-COLING-2024*, ed. by Rachele Sprugnoli and Marco Passarotti. Torino, Italia: ELRA and ICCL, May 2024, pp. 30–35. URL: https://aclanthology.org/2024.lt4hala-1.4/.

[21] Radford, Alec, Wu, Jeffrey, Child, Rewon, Luan, David, Amodei, Dario, and Sutskever, Ilya. “Language Models are Unsupervised Multitask Learners”. en. Tech. rep. OpenAI, 2019. URL: https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

[22] Sanh, Victor, Debut, Lysandre, Chaumond, Julien, and Wolf, Thomas. “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter”. arXiv:1910.01108 [cs]. Mar. 2020. DOI: 10.48550/arXiv.1910.01108. URL: http://arxiv.org/abs/1910.01108.

[23] “Text Chunking Using Transformation-Based Learning”. en. In: *Text, Speech and Language Technology*. ISSN: 1386-291X. Dordrecht: Springer Netherlands, 1999, pp. 157–176. ISBN: 978-90-481-5349-7 978-94-017-2390-9. DOI: 10.1007/978-94-017-2390-9_10. URL: http://link.springer.com/10.1007/978-94-017-2390-9_10.

[24] Vaswani, Ashish, Shazeer, Noam, Parmar, Niki, Uszkoreit, Jakob, Jones, Llion, Gomez, Aidan N., Kaiser, Lukasz, and Polosukhin, Illia. “Attention Is All You Need”. 2017. arXiv: 1706.03762 [cs.CL]. URL: https://arxiv.org/abs/1706.03762.

[25] Verkijk, Stella. “gloBERTise”. 2025. URL: https://huggingface.co/globalise/GloBERTise.

[26] Verkijk, Stella, Vossen, Piek, and Sommerauer, Pia. “Language Models Lack Temporal Generalization and Bigger is Not Better”. In: *Findings of the Association for Computational Linguistics: ACL 2025*, ed. by Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar. Vienna, Austria: Association for Computational Linguistics, July 2025, pp. 20629–20637. ISBN: 979-8-89176-256-5. DOI: 10.18653/v1/2025.findings-acl.1060. URL: https://aclanthology.org/2025.findings-acl.1060/.

[27] Wang, Huiming, Cheng, Liying, Zhang, Wenxuan, Soh, De Wen, and Bing, Lidong. “Order-Agnostic Data Augmentation for Few-Shot Named Entity Recognition”. In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Bangkok, Thailand: Association for Computational Linguistics, Aug. 2024, pp. 7792–7807. DOI: 10.18653/v1/2024.acl-long.421. URL: https://aclanthology.org/2024.acl-long.421/.

[28] Wu, Shijie and Dredze, Mark. “Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT”. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 833–844. DOI: 10.18653/v1/D19-1077. URL: https://www.aclweb.org/anthology/D19-1077.

[29] Zaratiana, Urchade, Tomeh, Nadi, Holat, Pierre, and Charnois, Thierry. “GLiNER: Generalist Model for Named Entity Recognition using Bidirectional Transformer”. arXiv:2311.08526 [cs]. Nov. 2023. DOI: 10.48550/arXiv.2311.08526. URL: http://arxiv.org/abs/2311.08526.

[30] Zhou, Ran, Li, Xin, He, Ruidan, Bing, Lidong, Cambria, Erik, Si, Luo, and Miao, Chunyan. “MELM: Data Augmentation with Masked Entity Language Modeling for Low-Resource NER”. en. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, 2022. DOI: 10.18653/v1/2022.acl-long.160. URL: https://aclanthology.org/2022.acl-long.160.