Recognizing and Linking Entities in Old Dutch Text: A Case Study on VOC Notary Records

Hendriks, B.; Groth, P.; van Erp, M.

Publication date
2021

Document Version
Final published version

Published in
COLCO 2020 : Collect and Connect: Archives and Collections in a Digital Age 2020

License
CC BY

Link to publication

Citation for published version (APA):
Hendriks, B., Groth, P., & van Erp, M. (2021). Recognizing and Linking Entities in Old Dutch Text: A Case Study on VOC Notary Records. In A. Weber, M. Heerlien, E. Gassó Miracle, & K. Wolstencroft (Eds.), *COLCO 2020 : Collect and Connect: Archives and Collections in a Digital Age 2020: Proceedings of the International Conference Collect and Connect: Archives and Collections in a Digital Age : Leiden, the Netherlands, November 23-24, 2020* (pp. 25-36). (CEUR Workshop Proceedings; Vol. 2810). CEUR-WS. http://ceur-ws.org/Vol-2810/paper3.pdf

Recognising and Linking Entities in Old Dutch Text: A Case Study on VOC Notary Records

Barry Hendriks[1], Paul Groth[1], and
Marieke van Erp[2]

[1] University of Amsterdam
barryhendriks98@gmail.com, p.t.groth@uva.nl
[2] KNAW Humanities Cluster
marieke.van.erp@dh.huc.knaw.nl
Amsterdam, the Netherlands

Abstract. The increased availability of digitised historical archives allows researchers to discover detailed information about people and companies from the past. However, the unconnected nature of these datasets presents a non-trivial challenge. In this paper, we present an approach and experiments to recognise person names in digitised notary records and link them to their job registration in the Dutch East India company's records. Our approach shows that standard state-of-the-art language models have difficulties dealing with 18th century texts. However a small amount of domain adaption can improve the connection of information on sailors from different archives.

Keywords: named entity recognition · maritime history · domain adaptation.

1 Introduction

The Dutch East India Company (the VOC) is known as one of the first multinational corporations, employing thousands of people from a variety of countries during its existence (1601-1800)[17]. The company held extensive records about their employees, recording information about their place of origin, the ships they sailed on, and the reason for their termination of employment [21]. Of these records, 774,200 have been preserved and digitised, facilitating research into the corporation (cf. [20]). Identifying which records within this collection refer to the same person can provide more insight into the lives of VOC employees as shown by [16]. Being able to connect to other sources (e.g. notary records) would provide another dimension to the analysis. Enabling, for instance, research into the lives of sailors as such records can provide information on who a sailor's beneficiaries were or whether they had any debts.

The Amsterdam City Archive has undertaken large-scale digitisation projects. An example is the Alle Amsterdamse Akten project,[3] which includes many documents from notaries who are known to have dealt with VOC employees.

[3] 'All Amsterdam Deeds' `https://www.amsterdam.nl/stadsarchief/organisatie/projecten/alle-amsterdamse/`

In this paper, we present an approach and experiments for identifying and linking sailors in both the VOC and Amsterdam notary records. We show the importance of domain adaptation of state of the art Dutch language models (e.g. BERTje[22]) in order to achieve acceptable performance on the named entity recognition (NER) task for this domain. Our contributions are threefold: 1) named entity recognition and linking software adapted to the 17th century maritime domain; 2) a gold standard dataset for evaluation in this domain; and 3) experimental insights into language technology for early-modern documents.

This paper is structured as follows. In Section 2, we describe related work and in Section 3, the datasets. Section 4 presents our approach and experimental setup, followed by an evaluation in Section 5. This is followed by conclusions and recommendations for future work (Section 6). The code and data of all experiments performed can be found at *https://github.com/barry98/VOC-project*.

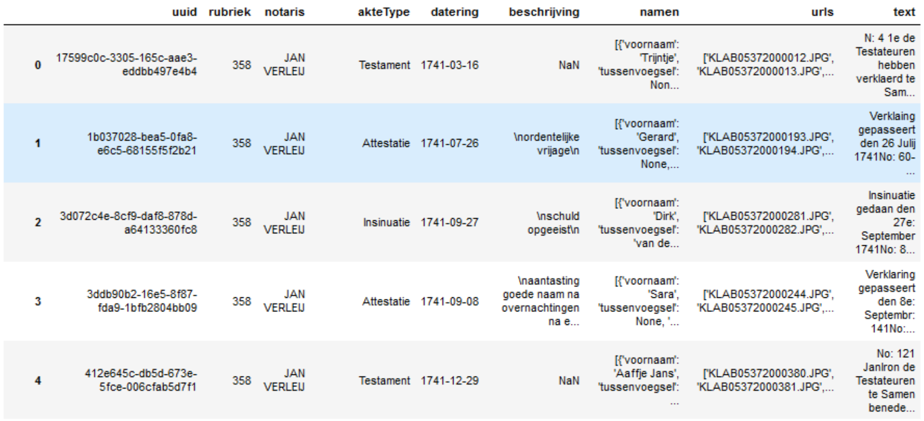
2 Related Work

We employ a combination of named entity recognition (NER), record linkage (RL), and named entity linking (NEL). In this section, we give a brief overview of the most important techniques from these three areas.

Named Entity Recognition: Extensive research has been done into Named Entity Recognition [15]. Currently, neural networks [7, 3, 13] achieve top performance in NER with F_1 scores of around .81-.82 as compared to 0.77 in previous approaches. In particular, NER systems based on large scale language models such as BERT [3] perform well. Dutch versions of BERT have been created by training on Dutch texts, resulting in the BERTje and RobBERT models. [2, 22]. BERTje achieves an F_1 score of 0.88 on standard benchmark datasets.

Record Linkage Record Linkage, the finding of records that refer to the same entity, has been a topic of interest for statisticians, historians, and computer scientists alike [1]. There are two main types of record linkage models, deterministic and probabilistic. The older deterministic models are only able to find exact matches whilst newer probabilistic models can use a threshold to determine whether non-exact matches should be linked [19]. Previous research has investigated the use of record linkage on text data from the middle ages and the early modern period using artificially created database [6]. Other work has shown the advantages of probabilistic record linkage for humanities related data (e.g. to link entities of three different databases on Finnish soldiers in World War II) [11].

Named Entity Linking Named Entity Linking, the linking of entities to a knowledge base, has been extensively researched [18]. Approaches using Wikipedia as a knowledge base can achieve impressive performance with accuracy scores ranging from 91.0 to 98.2 for linking to persons [9]. Recent work has looked the



	uuid	rubriek	notaris	akteType	datering	beschrijving	namen	urls	text
0	17599c0c-3305-165c-aae3-eddbb497e4b4	358	JAN VERLEIJ	Testament	1741-03-16	NaN	[{'voornaam': 'Trijntje', 'tussenvoegsel': Non...	['KLAB05372000012.JPG', 'KLAB05372000013.JPG',...	N: 4 1e de Testateuren hebben verklaerd te Sam...
1	1b037028-bea5-0fa8-e6c5-68155f5f2b21	358	JAN VERLEIJ	Attestatie	1741-07-26	\nordentelijke vrijage\n	[{'voornaam': 'Gerard', 'tussenvoegsel': None,...	['KLAB05372000193.JPG', 'KLAB05372000194.JPG',...	Verklaing gepasseert den 26 Julij 1741No: 60-...
2	3d072c4e-8cf9-daf8-878d-a64133360fc8	358	JAN VERLEIJ	Insinuatie	1741-09-27	\nschuld opgeeist\n	[{'voornaam': 'Dirk', 'tussenvoegsel': 'van de...	['KLAB05372000281.JPG', 'KLAB05372000282.JPG',...	Insinuatie gedaan den 27e: September 1741No: 8...
3	3ddb90b2-16e5-8f87-fda9-1bfb2804bb09	358	JAN VERLEIJ	Attestatie	1741-09-08	\naantasting goede naam na overnachtingen na e...	[{'voornaam': 'Sara', 'tussenvoegsel': None, '...	['KLAB05372000244.JPG', 'KLAB05372000245.JPG',...	Verklaring gepasseert den 8e: Septembr: 141No:...
4	412e645c-db5d-673e-5fce-006cfab5d7f1	358	JAN VERLEIJ	Testament	1741-12-29	NaN	[{'voornaam': 'Aaffje Jans', 'tussenvoegsel': ...	['KLAB05372000380.JPG', 'KLAB05372000381.JPG',...	No: 121 Janlron de Testateuren te Samen benede...

Fig. 1: Five entries of the notary dataset, indicating the record identifier (uuid), the catalogue number (rubriek), the notary name (notaris), the type of document (akteType), its date (datering), a short description (beschrijving), the annotated names (namen), URLs to the scans (urls) and the document text (text)

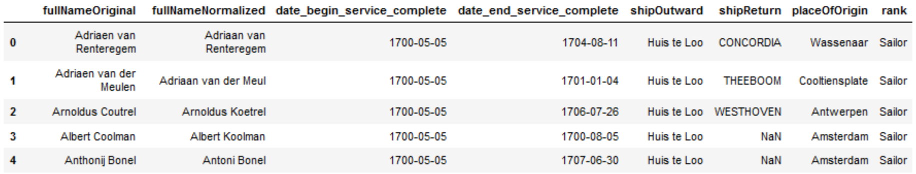
use of graph-based methods that use the neighborhood of an entity to improve linking performance on knowledge basis other than Wikipedia [10]. Other work has tried to remove hand crafted features by learning a entity linking end-to-end using neural networks[12]. A critical challenge to using entity linking methods in this domain is the lack of free text associated with the entities under consideration (i.e. sailors).

3 Data

Amsterdam Notary Archives: Jan Verleij The Amsterdam City Archives contain scans from various notaries. For this project, we focused on Jan Verleij, as his office was situated near the harbour of Amsterdam and he is known to have had many dealings with VOC personnel. All records were digitised with help of Handwritten Text Recognition (HTR) software. The dataset[4] consists of annotated data detailing: date, names of those involved, entry type, description of record, and the corresponding scans making up the record. The annotations were performed by 938 volunteers and involved manual tagging of the names of all clients and their associates found within each record. The names of the notary and the professional witnesses that worked for the notary were not tagged. An example of the data is shown in Figure 1.

VOC data: The VOC data is a list of 774,200 entries describing personnel sailing for the Dutch East India Company (VOC). As it was possible to reenlist after completing a tour, not all entries describe distinct individuals. The

[4] The dataset can be found at: `https://assets.amsterdam.nl/publish/pages/885135/saa_index_op_notarieel_archief_20191105.zip`



	fullNameOriginal	fullNameNormalized	date_begin_service_complete	date_end_service_complete	shipOutward	shipReturn	placeOfOrigin	rank
0	Adriaen van Renteregem	Adriaan van Renteregem	1700-05-05	1704-08-11	Huis te Loo	CONCORDIA	Wassenaar	Sailor
1	Adriaen van der Meulen	Adriaan van der Meul	1700-05-05	1701-01-04	Huis te Loo	THEEBOOM	Cooltiensplate	Sailor
2	Arnoldus Coutrel	Arnoldus Koetrel	1700-05-05	1706-07-26	Huis te Loo	WESTHOVEN	Antwerpen	Sailor
3	Albert Coolman	Albert Koolman	1700-05-05	1700-08-05	Huis te Loo	NaN	Amsterdam	Sailor
4	Anthonij Bonel	Antoni Bonel	1700-05-05	1707-06-30	Huis te Loo	NaN	Amsterdam	Sailor

Fig. 2: Five entries of the VOC dataset describing an employee's name as found on the record (fullNameOriginal), his normalised name (fullNameNormalised), his service start date (date begin service complete), his service end date (data end service complete), the name of the ship he sailed on during the outward journey to Indonesia (shipOutward), the ship he came back on (shipReturn), the place of birth (placeOfOrigin) and the rank he was hired in (rank).

information used for this project are name, birthplace, date of employment, date of resignation, ships that were sailed on, and rank. An example of a single entry can be found in Figure 2.[5]

Data Annotation To train and evaluate the proposed record linkage model, links between individuals in the notary data and the VOC data were created. The process consisted of selecting possible matches and subsequently confirming or denying of these matches. To reduce the amount of annotation work, two conditions were specified that needed to be satisfied in order for two individuals to be considered a possible match.

- The first condition was a fuzzy match ratio of at least 80% between the name of the notary entry and the name of the VOC employee;
- The second condition consisted of a notary entry date that was 90 days or fewer before the leave date of the ship that the VOC employee left on, or a notary entry date that was 90 days or fewer after the return date of the ship that the VOC employee returned on.

If these conditions were satisfied, then the individual of the notary entry and the VOC employee were reviewed by annotators. Annotators manually reviewed possible matches by looking in the HTR text of the notary entry for keywords obtained from the VOC employee data. Examples of keywords include: the name of the ships that were sailed on, rank, and place of origin. Aside from these keywords some general keywords suggesting involvement with the VOC were also looked for (e.g. 'Oostindische Compagnie', 'Kamer Amsterdam', and 'Oost Indie'). If, based on the found keywords, the annotator believed the possible match to be a true match, then this match was recorded as such.

Four annotators were used, one of which was one of the authors of this study. To check the effort and speed of the annotation process, the annotators were asked to perform the task for an hour. The slowest annotator annotated 554 possible matches within the given hour, the fastest 991. Fleiss' Kappa was used

[5] The dataset can be found at: `https://www.nationaalarchief.nl/onderzoeken/index/nt00444?searchTerm=`

Annotator	match	non match
A	6	548
B	6	548
C	8	546
D	47	507

(a) The number of confirmed matches and non matches from the test set.

Annotators	Fleiss' Kappa
All	0.359
Without D	0.899

(b) Fleiss' Kappa with and without annotator D

Table 1: Inter-annotator agreement

to measure inter-annotator agreement over the 554 cases that all four annotators completed [5]. The number of matches confirmed by each annotator and matches disconfirmed by each annotator are found in Table 1a.

Calculating Fleiss' Kappa results in value of 0.359 which according to [14] equals a fair agreement. Many disagreements can be explained by the many confirmed matches made by annotator D. After discussing the results with annotator D, it became clear that they misunderstood the annotation guidelines, assuming the data was far less imbalanced than it actually is. As a result, the annotation guidelines were updated to clarify that partial matches of location, rank, and ships alone are not enough to warrant a match. When annotator D is excluded from the Fleiss' Kappa calculation, we find a value of 0.899, equalling an almost perfect agreement (see Table 1b).

To further minimise mistakes made during the annotation process, we reviewed all confirmed matches resulting in re-annotating three matches as non-matches and confirming three ambiguous matches. In total 1,624 possible matches were annotated, resulting in 101 confirmed matches.

4 Methodology

Our approach starts with identifying names in the notary records, for which we then try to find candidate matches and then the best match in the VOC records. In the remainder of this section, we detail each step.

4.1 Named Entity Recognition

To identity individuals in the notary data we first created a basic NER model, we then adapted it for persons in 18th century data, and finally other named entities were recognised.

Basic Model: Two different existing NER models were considered: 1) the spaCy dutch model [4]; and 2) the dutch BERT model BERTje [22]. BERTje is chosen for its accuracy compared to other multilingual BERT models, spaCy for its simple API and fast processing time. We used recall and precision to evaluate these models on the available notary data. As the HTR text contains many misspellings, fuzzy matching is used. If a recognised person name matches at least

90% with an annotated name, the recognised entity is considered to be a true positive. Since all names recorded in the notary texts are the full name of a person, person entities consisting of a single token are discarded.

Domain adaptation: As 18th century Dutch differs in form from the contemporary texts the models were trained on, our results were much lower than reported F_1 scores of 0.8 or higher. We therefore adapted the model to the 18th century domain by first adding the named persons the model previously recognised correctly to further train the model. This method ensures that all annotated entities are correct, however as we do not introduce new entities, this method will probably not increase the recall of the model, only increase its precision.

The second approach is to use fuzzy matching to find all instances of the annotated names of the notary data in each HTR text. To accomplish this for every HTR text available in the notary data, the corresponding annotated names are gathered. Fuzzy matching is then used to find each annotated name within the text, requiring 80% of the names to match. This method not only allows previously unrecognised names to be used for training, but it also removes many falsely recognised persons from the training data.

4.2 Record Linkage

We tested both record linkage (RL) and entity linking approaches for the linking individuals in the notary and VOC datasets. The RL method proved to be more effective. This is likely due to the fact that the entries do not contain much free text thus hurting the performance of entity linking as mentioned in Section 2.

Blocking As typical in record linkage, an initial selection of potential matching records is performed (i.e. blocking). First, possible VOC candidates for each individual are found within the notary data using fuzzy string matching. To speed up the selection process, we only try to match individuals from the VOC data that have a leave or return data within one year from the notary record date. To consider an individual a possible candidate, there has to be at least an overlap of 80% in the spelling of their names in both datasets. We further narrow down the initial selection based on the date of the notary record and the leave or return date of the VOC individual: the date of the notary entry has to be either 90 days or less before the leave date or 90 days or less after the return date of the VOC individual.

Linking Records Dedupe was used to link the records [8]. Dedupe uses machine learning to perform fuzzy matching, deduplication, and entity resolution with the help of active learning. Models train themselves by presenting the user with its least certain match to judge. Using the user's judgment, the model recalculates the weights for each feature and repeats the process until the user stops it. Each match is provided with a certainty score to establish a threshold for discarding or keeping matches. One drawback of active learning is that it can be very hard to confirm and disconfirm the exact same matches for each model leading to variations in the optimal threshold for each model.

We trained several different models, each containing a different number of confirmed and disconfirmed matches. All models are trained and tested on a subset of the notary data that was linked with the VOC data through annotation as described in section 3. For a match to be considered a true positive the RL model has to match a notary entity with its corresponding VOC entity.

5 Evaluation

In this section, we first present the results of the named entity recognition step, followed by the record linkage step.

5.1 Named Entity Recognition

The NER models are evaluated on either the entire notary dataset in the case of the basic models, or a test subset of the notary data in the case of domain adaptation. For each model, the entities tagged as a person are compared to the annotated names of those involved in the notary entry (Table 2a).

Both basic models do not perform very well, most likely due to a combination of both HTR text and old Dutch being too different from the modern Dutch the models were trained on. Furthermore, there is a significant difference in processing time between these two models. Both models were tested on a single computer possessing an Intel i5-6600 CPU, a NVIDIA GeForce GTX 1060 GPU, and 16 GB of RAM. BERTje processes all 13,063 HTR texts in about two hours. The spaCy model processes all texts in about 20 minutes. A possible explanation for this difference could be that current BERT models, including BERTje, only allow for a maximum of 512 tokens to be processed at once, requiring many of the HTR texts to be split into smaller texts.

We evaluated two domain adaptation methods for the spaCy NER model, the *previously recognised* approach and the *fuzzy matching* approach as explained in section 4.1. The data was split randomly into a training set containing with 70% and the remaining 30% was held out for testing (Table 2b).

Both adapted approaches far outperform the basic spaCy model. However, the *fuzzy matching* approach also achieves a higher recall than the *previously recognised* approach. To validate the performance of the model created by the fuzzy matching approach, we perform a k-fold cross validation with 10 folds. As Table 2c shows, the fuzzy matching approach delivers a reliable model independent of the way the data has been split.

Error Analysis: To gain more insight into the mistakes that the NER model makes, we analysed 500 false negatives and 536 false positives. From this analysis, we found that some errors were caused by flaws in the model, others can be attributed to flaws in the HTR or the annotation of the data.

The false negatives from the model come in three different types:

Model	Precision	Recall	F1 score
spaCy	0.101	0.416	0.163
BERTje	0.072	0.538	0.127

(a) Precision, Recall, and F1 score for the basic NER models

Model	Precision	Recall	F1 score
Previously Recognised	0.732	0.491	0.588
Fuzzy Matching	0.733	0.737	0.735

(b) Precision, Recall, and F1 score for the different approaches for further training

Model	Precision	Recall	F1 score
Worst Fold	0.694	0.745	0.719
Average	0.732	0.736	0.732
Best Fold	0.731	0.756	0.743

(c) Precision, recall, and F1 score for the worst fold, best fold, and the average over all folds from the trained NER model

Table 2: Results of NER experiment

- The name was not tagged (313 counts)
- The name was tagged but too dissimilar from the annotated name due to HTR (127 counts)
- The name was only partially tagged (60 counts)

The majority of the mistakes are when the NER model will simply not tag certain names. There does not seem to be any pattern in the names themselves suggesting that the model was simply unable to tag them due to their position within the text. Explanations for this could be the absence of capitalisation in some names or a lack of punctuation between names. For example, for the name ‘pieter Jansen Hendrik havens’, the lack of capitalisation in the first and last word and the lack punctuation between ‘pieter Jansen’ and ‘Hendriks havens’ causes the model to recognise ‘Jansen Hendrik’ as the name. The second most common mistake is not directly a flaw in the NER model as much as it is a flaw in the HTR software. A prominent mistake is the use of the letter ‘y’ or the digraph ‘ij’ as the use of the letter ‘y’ was more common in the past, where as it has been replaced with the digraph ‘ij’ in many cases in modern Dutch. The last and least common mistake is the partial tagging of a name. Since tagged entities consisting of a single word are discarded before evaluation, all names involving this mistake either include a family name affix or a middle name. The most common mistake seems to be that the NER model mistakes the middle name for the last name. This is the case for 41 of the 60 mistakes. An example of this would be the name ‘Pieter Hendrik Kornelisse’. Here the NER model would recognise the middle name ‘Hendrik’ as the last name, resulting in tagging ‘Pieter Hendrik’ as a person whilst ‘Kornelisse’ is discarded.

The false positives can be divided in four different types:

- The name was annotated but too dissimilar from the tagged name due to HTR (211 counts)
- The name was only partially tagged (136 counts)

Model	Threshold	Precision	Recall	F1 score
10 disconfirmed	0.9	0.882	0.577	0.698
20 disconfirmed	0.6	0.900	0.692	0.783
30 disconfirmed	0.7	0.947	0.692	0.799
40 disconfirmed	0.3	0.792	0.731	0.760
50 disconfirmed	**0.3**	**0.846**	**0.846**	**0.846**
60 disconfirmed	0.2	0.762	0.615	0.681
70 disconfirmed	0.2	0.864	0.773	0.816
80 disconfirmed	0.7	0.944	0.654	0.773
90 disconfirmed	0.3	0.786	0.846	0.815
100 disconfirmed	0.3	0.875	0.808	0.840

Table 3: Optimal threshold, precision, recall, and F_1 score for multiple RL models. The boldfaced row indicates the best performing model.

- The name was not annotated (97 counts)
- The tagged entity is not an actual person (92 counts)

Similarly to the false negatives, the false positives contain many mistakes due to the HTR software distorting the name of a person. However, unlike the false negatives the use of the letter 'y' and digraph 'ij' seems to have far less of an impact. Instead it would seem that since many names can be found multiple times within a single text, the HTR software has a higher chance to make just enough mistakes for one of the occurrences of a name so that it is no longer similar enough to be recognised as the same name. The second most common mistake was that the name was only partially tagged. Similarly to the partially tagged false negatives, the largest problem here seems to be that the middle name is mistaken for the last name. The third most common mistake was the absence of annotation for an entity that was confirmed to be a person. These mistakes can be explained by the fact that professional witnesses were not annotated. The least common mistake is that the tagged entity was simply not a person. Common mistakes are the combination of a persons last name along with his or her trade, place of origin, or first name of a different person. The latter case can be explained by the lack of punctuation in the texts. Examples of this would be 'Evert Hendriks Kruidenier', where kruidenier is the trade of the person Evert Hendriks or 'Wilhelmina Hugenoot van Volendam' where van Volendam is the place of origin. It is important to note that it is common for Dutch last names to be either a trade or the place of origin, making this a hard problem to solve.

5.2 Record Linkage

We evaluated the RL models on a subset of the test data containing a representative number of matches and non-matches for the entire dataset. We evaluated ten different models, each with a different number of matches confirmed and disconfirmed during active training. We decided to test a model for each increment of ten disconfirmed matches, as the minimum recommended amount of confirmed and disconfirmed matches is ten, according to the developers of Dedupe. The results of these models can be found in Table 3.

As expected, active learning causes some fluctuation in the optimal threshold for each model. The models that have a very low optimal threshold such as the 50 disconfirmed and 100 disconfirmed models seem to have the best performance. Despite the low threshold, these models still obtain a satisfactory precision score. This suggests that while models are hesitant to make matches, the matches that it does make are accurate. Conversely, the models with high thresholds seem to make more matches, which has to be compensated for with a high threshold resulting in a reduced recall. An important thing to keep in mind when looking at these results is that the test data had only 26 actual matches. This means that just a single true positive more or less can make a significant difference in the recall. For example, the difference between the recall of the 50 disconfirmed model and the 100 disconfirmed model can be explained by a single true positive.

Error Analysis: As the test set was relatively small, we analysed all mistakes made by the best performing RL model. In total, the model was responsible for four false positives and four false negatives, with just a single type of mistake for both categories. In the case of the false positives, the entity recognition of ship names falsely recognised unrelated words as ship names. These where then also deemed similar enough by the model to warrant a match. An example of this is the word ‘beiden’, the Dutch word for ‘both’, that was falsely recognised as the ship name ‘Leiden’. Two of the four false positives had no matches in rank or location, implying the model based the match solely on the presence of the ship name.

For the false negatives, the problem would be the inverse of that of the false positives. In these cases, a ship name was not found in the texts, causing the model to not make any matches. Again, the model seems to value the presence of a ship name far above the presence of a rank or location, as two of the four false negatives did have a matching rank or location. However, if the model finds a ship name, a matching rank or location does increase the certainty score of the model. Matches based solely on a found ship name posses certainty scores lower than 0.5. Meanwhile, matches with a matching rank or location all posses scores of 0.65 or higher, with most obtaining a score between 0.85 and 0.99.

6 Conclusions and future work

We trained a NER model and an RL model to recognise and link entities between notary records from 18th century notary Jan Verleij and the VOC employee records. Our experiments show that readily available NER models, such as the Dutch spaCy model and BERTje, perform poorly on HTR data of old Dutch texts. However, if some annotated named entities are available performance can be improved from F_1=0.163 to F_1=0.743. This is still somewhat below state-of-the-art performance of these models on modern text (e.g. CoNLL-2002 benchmark dataset ($F_1 = 0.883$) but the notary records constitute far less training data and are less conformant to spelling and punctuation standards.

Although the precision of linking entities is quite high, the recall is still somewhat lacking. In practice, this means that although the predicted matches will almost always be a true match, only about 60-70% of the actual matches are found. The usefulness of the current model depends on the use case and the amount of data that is available. If enough data is available then the current model can produce a sufficient number of actual matches without providing too many false matches. However, if data is scarce then the matches not found by the model might be necessary, reducing the usability of the model.

Annotated data for the locations, ranks, and ships in the notary records would be a valuable addition to the the NER model. For the RL model the lack of annotated data greatly reduced the amount of data that could be trained and tested on. It is clear that for the advancement of NLP on old text more training data is needed. Given the experience in this project, we believe that there is further scope for finding latent training data in newly digitised historical data. The dataset created for this project can provide a template for such initiatives.

There are certainly areas of improvement possible for the language models. For the NER model it would be interesting to fully train multilingual versions of BERT on this type of text. Since these models perform extremely well on modern texts, further training of these models for old texts might result in far better models than those obtained in this project.[6]

Aside from improving the RL model, the linking of entities might also be improved by instead opting to make use of named entity linking techniques. Further research could be conducted into named entity linking with smaller local knowledge bases instead of the large knowledge bases such as Wikipedia. Combining named entity linking and record linkage is also an interesting avenue of research given the semi-structured nature of much of this data.

Acknowledgements

The authors would like to thank their KNAW HuC colleagues from the maritime careers project, in particular Lodewijk Petram and Jelle van Lottum for their advice on all things VOC, Jirsi Reinders from Stadsarchief Amsterdam for access to and explanations of the notary data, our annotators for helping create the gold standard and the anonymous reviewers for their feedback and suggestions.

References

1. Christen, P.: Data matching: concepts and techniques for record linkage, entity resolution, and duplicate detection. (2012)
2. Delobelle, P., Winters, T., Berendt, B.: Robbert: a dutch roberta-based language model (2020)
3. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. CoRR **abs/1810.04805** (2018), `http://arxiv.org/abs/1810.04805`
4. Explosion: spacy (2020), `https://spacy.io/models/nl`

[6] The first steps of this have already begun with the MacBERTh project that aims to create natural language models trained on both English and Dutch historical text data `https://pdi-ssh.nl/en/2020/06/funded-projects-2020-call/`

5. Fleiss, J.L.: Measuring nominal scale agreement among many raters. Psychological bulletin **76**(5), 378 (1971)
6. Georgala, K., van der Burgh, B., Meeng, M., Knobbe, A.: Record linkage in medieval and early modern text. In: Population Reconstruction, pp. 173–195. Springer (2015)
7. Gillick, D., Brunk, C., Vinyals, O., Subramanya, A.: Multilingual language processing from bytes (2015), https://arxiv.org/pdf/1512.00103.pdf
8. Gregg, F., Eder, D.: Dedupe (2019), https://github.com/dedupeio/dedupe
9. Hachey, B., Radford, W., Nothman, J., Honnibal, M., Curran, J.R.: Evaluating entity linking with wikipedia. Artificial Intelligence **194**, 130 – 150 (2013), http://www.sciencedirect.com/science/article/pii/S0004370212000446, artificial Intelligence, Wikipedia and Semi-Structured Resources
10. Han, X., Sun, L., Zhao, J.: Collective entity linking in web text: A graph-based method. In: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. pp. 765–774. SIGIR '11, Association for Computing Machinery, New York, NY, USA (2011), https://doi.org/10.1145/2009916.2010019
11. Koho, M., Leskinen, P., Hyvönen, E.: Integrating historical person registers as linked open data in the warsampo knowledge graph. In: SEMANTiCs 2020, In the Era of Knowledge Graphs, Proceedings. Springer-Verlag (09 2020), accepted
12. Kolitsas, N., Ganea, O.E., Hofmann, T.: End-to-end neural entity linking. arXiv preprint arXiv:1808.07699 (2018)
13. Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., Dyer, C.: Neural architectures for named entity recognition. CoRR **abs/1603.01360** (2016), http://arxiv.org/abs/1603.01360
14. Landis, J.R., Koch, G.G.: The measurement of observer agreement for categorical data. biometrics pp. 159–174 (1977)
15. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvisticæ Investigationes **30**(1), 3–26 (2007), https://www.jbe-platform.com/content/journals/10.1075/li.30.1.03nad
16. Petram, L., van Lottum, J., van Koert, R., Derks, S.: Small lives, big meanings. expanding the scope of biographical data through entity linkage and disambiguation. In: BD. pp. 22–26 (2017)
17. Petram, L., van Lottum, J.: Maritime careers: The life and work of european seafarers, 1600-present (2019)
18. Rao, D., McNamee, P., Dredze, M.: Entity Linking: Finding Extracted Entities in a Knowledge Base, pp. 93–115. Springer Berlin Heidelberg, Berlin, Heidelberg (2013), https://doi.org/10.1007/978-3-642-28569-1_5
19. Sayers, A., Ben-Shlomo, Y., Blom, A.W., Steele, F.: Probabilistic record linkage. International journal of epidemiology **45**(3), 954–964 (2016)
20. Van Bochove, C., Van Velzen, T.: Loans to salaried employees: the case of the Dutch East India Company, 1602–1794. European Review of Economic History **18**(1), 19–38 (02 2014). https://doi.org/10.1093/ereh/het021, https://doi.org/10.1093/ereh/het021
21. Velzen, D.A., Gaastra, P.F.: Thematische collectie: Voc opvarenden. https://doi.org/10.17026/dans-xpp-abdp (2000). https://doi.org/https://doi.org/10.17026/dans-xpp-abdp
22. de Vries, W., van Cranenburgh, A., Bisazza, A., Caselli, T., van Noord, G., Nissim, M.: Bertje: A dutch bert model. arXiv preprint arXiv:1912.09582 (2019)