Studying Language Variation Considering the Re-Usability of Modern Theories, Tools and Resources for Annotating Explicit and Implicit Events in Centuries Old Text

Stella Verkijk
Vrije Universiteit Amsterdam
Huygens Institute
s.verkijk@vu.nl

Pia Sommerauer and **Piek T. J. M. Vossen**
Vrije Universiteit Amsterdam
p.sommerauer@vu.nl
p.t.j.m.vossen@vu.nl

Abstract

This paper discusses the re-usability of existing approaches, tools and automatic techniques for the manual annotation and automatic extraction of events in a challenging variant of centuries old Dutch in documents of the Dutch East India Company. We describe our annotation process and provide a thorough analysis of different versions of manually annotated data and the first automatic results from two fine-tuned Language Models. The paper studies to what extent we can use NLP theories and tasks formulated for modern English to design an annotation task for early modern Dutch and to what extent we can use NLP models and tools built for modern Dutch (and other languages) on early modern Dutch. We believe these analyses give us insight into how to deal with the large variation that language shows in describing events, and how this variation may differ across domains. We release the annotation guidelines, annotated data, and code (https://github.com/StellaVerkijk/VarDial2024).

1 Introduction

Event extraction is a well-researched but very challenging task in Natural Language Processing (NLP). Though there are many datasets, systems and ontologies created for event extraction, there is little consensus on how to create a robust system for heterogeneous material. This problem is amplified when the texts are centuries old and the context is to a large extent unknown.

In this paper, we study a use case of annotating early modern Dutch texts for event trigger detection and classification. These texts originate from the Dutch East India Company (VOC) archives. This corpus of handwritten communications within the VOC holds a vast amount of information on trade, culture, business, slavery and early globalisation, which took place across much of the Indian Ocean World in the 17th and 18th centuries. The complete corpus consists of twenty-five million pages. It has been hard to conduct historical research with this corpus, because of its size, and because not many people can read the handwritten text (see Figure 1[1]).

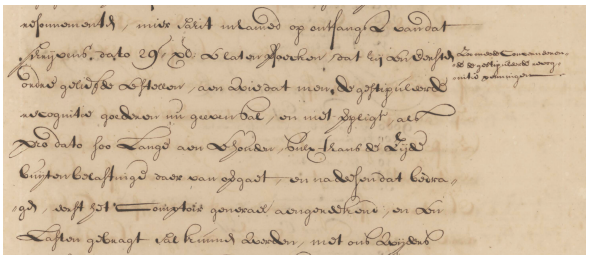

Figure 1: Snippet of the VOC archives

This paper describes the creation of a small annotated dataset that serves as a starting point for an automatic system that labels the archival material and enables a human-computer interaction solution: we are developing an event reconstruction pipeline to support a (re)search interface for historians.

The challenging nature of event extraction is showcased in recent results as reported in Hong et al. (2018) where deep learning systems achieve f-scores in the seventies for English, but drastically drop in performance when tested on data from a slightly different domain. Hong et al. (2018) also show how reaching high recall is a persistent problem in event extraction. We hypothesize that the main reason for this is because there is so much variation in how language is used to refer to events. We note that high recall is essential when building software that should support a search engine. In our case, we start our task unknown to the type and degree of variation in the language used, since much of the corpus' content and form remains unstudied. Although this poses considerable challenges, we can utilize event extraction as a looking glass

[1]National Archive, The Hague, The Netherlands, 1.04.02 (Archive of the VOC), inventory no. 1812, p. 33. https://www.nationaalarchief.nl/onderzoeken/archief/1.04.02/invnr/1812/file/NL-HaNA_1.04.02_1812_0803

Proceedings of the Eleventh Workshop on NLP for Similar Languages, Varieties, and Dialects (VarDial 2024), pages 174–187

through which we study the variation in the language.

In order to create any automatic system for event extraction, we have to ask ourselves the question: What kind of variation of Dutch are we dealing with? The subjects discussed and the way they are discussed might be vastly different from other early modern sources. We are looking at two centuries of history of a huge organization that in the early 17th century had more operations in Asia than all other European nations combined (Lucassen, 2004). There were no spelling conventions, countless different clerks writing, summarizing or translating texts, and intricate political and cultural conventions to adhere to in the language.

We find that i) the challenging nature of event extraction as a task, ii) the fact that the performance of automatic solutions highly depends on how similar the domain they were trained on was and iii) the complexity of the language we work with itself specifically call for a tailored solution. In our case this begins with defining a new annotation task and subsequently fine-tuning language models pre-trained on different varieties of Dutch.

Our contributions are the following. Firstly, we discuss and illustrate the complexity of interpreting the language in this specific corpus, providing deep analyses of examples of our data. Secondly, we evaluate the re-usability of existing tools and resources by employing them on these examples, showing how models trained on modern language struggle with the variation present in our data. Thirdly, we present a new annotation approach where annotators work in teams and annotations are guided by an ontology specifically built for our data. We provide agreement analyses at different stages of the annotation process and show how our approach leads to an inter-annotator agreement (IAA) of 84% for trigger detection, 86% for classification (of 80+ event types) and 72% for the combined task of detection and classification. We also provide first insights of automatic solutions fine-tuned on the annotated data. Lastly, we publish our annotated datasets, containing a thoroughly analysed test set with annotations adjudicated by four historians and a linguist.

2 Related Work

Various English datasets have been annotated with events. While these approaches yielded valuable insights, none of the existing annotation schemes satisfies the needs of our use case. The main limitation lies in the selection of events annotated. Some of the proposed schemes only cover event types that refer to an event's aspectuality (distinguishing between state, process, action etc.) such as in Saurí et al. (2006) (as used in for example TempEval-3 (UzZaman et al., 2013)), ISO-TimeML (Pustejovsky et al., 2010) and THYME-TimeML (Styler IV et al., 2014) (as used in SemEval-2016: Clinical TempEval (Bethard et al., 2016)). Other datasets contain semantically more informative event types like TRANSPORT, but still only represent one corner of a modern Western world, such as ACE (Walker et al., 2006) and a light-weight version of ACE, ERE (Chen et al., 2023), both created to represent a limited number of event types of interest to the military, the latter created to make annotation easier and more consistent (Aguilar et al., 2014). FrameNet (Baker et al., 1998) is too specific for our purposes, requiring specialised linguistic knowledge about frame semantics not relevant for historical analysis. PropBank (Kingsbury and Palmer, 2003) and VerbNet (Schuler, 2005) are overly driven by syntax and lexica. Existing lexical and syntax-driven approaches do not fit our purposes because we are dealing with text that has no clear sentence boundaries (see Section 4) and for which we have very limited lexical semantic resources.

There has also been extensive research in the field of event-centric ontologies. However, the event classes they contain are mostly not representative for an early modern Dutch world (e.g., SUMO (Pease et al., 2002), DOLCE (Borgo et al., 2022)). For example, SUMO has a class for *PoliticalRevolution*, but none for *Mutiny* or a revolt that does *not* result in overthrowing of government. Also, while it has an entity class for *HumanSlave*, it does not feature an event like *Enslaving*. Still, we can draw on the way general ontologies include certain axioms, such as the Brandeis Semantic Ontology (BSO) (Pustejovsky et al., 2006) and the Rich Event Ontology (REO) (Bonial et al., 2021) that explicitly incorporate qualia relations. Even more relevant in this respect is the Circumstantial Event Ontology (CEO) (Segers et al., 2017), which includes pre-, during- and post- states of events, to incorporate *weak* causality. One event possibly causes a second when the post-state of the first equals a pre-state of the second. Pustejovsky (2021) urges to embed the state-change model from AI within the compositional model of semantics adopted in linguis-

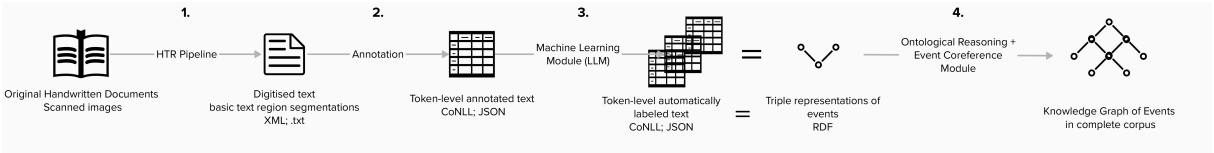

Figure 2: Event Reconstruction pipeline

tics. Verkijk and Vossen (2023) take Pustejovsky's and Segers' frameworks as a starting point and model Static Events (states) as logically inferred post-conditions of Dynamic Events (changes), e.g. an event like killing leads to a post-state of someone being dead. For this ontology, a team of historians identified and formulated event classes that are relevant for historical research in the VOC archive.

Recent state-of-the-art event extraction models using distributional embedding representations do not acquire an F-score above 0.77 for trigger detection and 0.74 for trigger detection + event classification on the ACE dataset (Hong et al., 2018). Hong et al. (2018) also show that systems trained on the broadcast news and newswire parts of the dataset and evaluated on the webblogs drop in F1 performance with 19.5-22 percentage points. This shows that even when adhering to the same annotation scheme, a difference in domain heavily influences performance. Finally, Hong et al. (2018) show that many systems demonstrate large gaps between precision and recall, where precision is almost always higher. We speculate that the variation in describing events is much larger than expected and requires other approaches than those offered by traditional NLP. We therefore expect that an end-to-end neural system for our use case can only partially reconstruct events and needs to be augmented with richer and more explicit semantics to connect the dots.

3 Approach

In order to support event-centric search in the archives, we aim to build an event-centric Knowledge Graph (KG). There are several steps that have to be undertaken to reach this end product: Figure 2 shows the most important steps in this pipeline. The handwritten documents first have to go through Handwritten Text Recognition (HTR) in order to become digitised (see Section 4). We then perform manual annotation on the digitised text. We plan to perform and experiment with some data augmentation at a later stage of the project, i.e. create synthetic training data. Finally, we fine-tune a Language Model (LM) to automatically annotate the rest of the corpus. This will provide us triple representations of events, which we gather in a KG. Through ontological reasoning we filter and complement our KG.

For the last step, we utilise the event ontology described by Verkijk and Vossen (2023). This event ontology is made for VOC archival material and models Static Events (SEs) as logical implications of Dynamic Events (DEs). For example, the election of a new person as king or raja in a certain region implies their status of *being a leader* from the moment of the election onward. Similarly, the Agent of a *Leaving* event *is no longer at the place* it left from the time of the event onward. These post-states of events are automatically inferable through the ontology. The ontology also features a taxonomic structure of DEs, allowing for generalizations like a *Leaving* event being a type of *Translocation* event. Such generalizations capture the variation in the data and language.

The ontology features 65 dynamic and 18 static events. Of the DEs, 50 of them imply a SE as a post-condition. There are only two SEs that cannot be inferred from the occurrence of a DE. The class *Dynamic Event* branches out in two classes that do not have any subclasses and five broad classes that branch out into more fine-grained subclasses. Those five classes are *SocialStatusChange*, *ChangeOfPossession*, *SocialInteraction* (with subclasses like *Mutiny, StartingAConflict*), *Translocation* and *InternalChange* (with subclasses like *Dying, Increasing, FallingIll*). The taxonomic structure of DEs is four steps at the deepest level.

The choice to create new manually annotated data following the event classes of Verkijk and Vossen's (2023) ontology was motivated by results of preliminary experiments we performed where we tried to use existing resources for automatic event detection (see Section 5). Furthermore, the ontology forms a closed world that guides annotations, where the richer semantics steer annotators to look for specific information in the text. It also enables us to alleviate some of the annotation labour:

The possibility of automatic inference allows us to infer unexpressed information. We expect that the automatic extraction of SEs will help solve a recall gap in future automatic labelling systems.

4 Data

4.1 The Corpus and its Contents

The corpus is the collection of *Overgekomen Brieven en Papieren* (Received Letters and Papers, OBP) within the VOC archive. The OBP contains the *Generale Missiven* (General Missives) and a large and varied collection of documents on which these missives are based. The General Missives are reports from the VOC's central administration (Council of India) in Batavia to the board (Gentlemen Seventeen) in the Dutch Republic. They contain accounts of all things current for the VOC world over almost two centuries, including, for example, detailed overviews of historical events, as well as social, political, economic and ecological developments. These narrative accounts begin with a brief introduction and then report in long sentences events that are more broadly described in the other documents that make up the OBP. In the margins are small summaries, which we call marginalia (see Figure 1).

The OBP spans the period 1610-1796 and contains around seven million handwritten pages. While the OBP has an average document length of 28 pages, a General Missive is on average 207 pages. For annotation, we selected pages from different types of documents from a range of different years. The annotated data we are releasing upon publication comprise 62 pages of 6 different documents. They include parts of General Missives, original missives, letters, journals and notes and they span a period of 151 years (1626-1777).

Throughout this paper we will refer to Example (1), the transcribed text of Figure 1, to illustrate the complexity of interpreting our data through an event annotation task. Example (1) is a snippet of a paragraph that spans four pages[1]. Within the paragraph, there is no indication of the end or beginning of a sentence. The text is written as one long description of happenings in a specific place at a specific time, which is a very common way of writing in our corpus. In order to illustrate the type of language used, we offer a word-by-word translation to English in (1b). For a paraphrased and more readable version and its translation to English, see Appendix A. Event triggers are printed in boldface. Corresponding event classes from our annotation scheme are *Getting; Request; SocialInteraction; Giving; HavingInPossession; ForcingToAct; FinancialTransaction; FinancialTransaction*.

(1a) **Original source**

'*(...) op* ***ontfangst*** *van dat „schrijvens, dato 29„e xb: te laten* ***versoecken****, dat hij ten Eersten* ***ordre*** *geliefde te stellen, aen wie dat men, de gestipuleerde recognitie goederen nu* ***geeven*** *bal, en niet* ***verpligt****, als pro dato soo Lange* ***aen tehouden****, sulx thans de zijde buijten* ***belastinge*** *daer van overgaet, en na deesen dat bedragen, eerst het comptoir generael* ***aengereekend; en ten Lasten gebragt*** *sal kunnen werden, (...)*[2]

(1b) **Literal translation**

(...) on ***reception*** *of that writing, date 29 xb to be* ***requested*** *that he firstly an* ***order*** *would like to establish, to who that one, the before identified taxable goods now shall* ***give****, and not* ***obliged****, if per the date so long* ***to hold****, so that the silk free of* ***tax*** *there from go off, and after these the amounts, first the local office general* ***charged; and debited*** *will be, (...)*

4.2 Data Processing: HTR

For HTR we use Loghi[3]. As mentioned, the archive contains handwritings of a vast amount of different people living in time periods that can differ more than a hundred years and there were no spelling conventions in early modern Dutch. On top of that, defining reading order and separating main text from marginalia is very challenging. Because of this, we are often dealing with very noisy output. For example, (1) showcases a character misclassification that transforms a verb into a noun. The transcribed 'bal' (*ball*) in 'aan wie men de gestipuleerde recognitie goederen nu geeven **bal**" (*to whom one the identified taxable goods now give* ***ball***) represents 'sal' (*shall*) in the original text. The untouched transcription of (1) is given in Appendix A, also showcasing how the HTR pipeline mis-identifies text regions, complicating the annotation process. Different transcription conventions in the different ground truth sets that Loghi was trained on, especially for punctuation, affect the transcriptions and make the Character Error Rate (CER) currently quite high (>10 percent). However,

[2] HTR errors related to region detection have been taken out of this example for clarity reasons

[3] https://github.com/knaw-huc/loghi

Loghi's HTR quality on our data using a classifying tool that was created for our data[4] shows that the HTR quality of a large majority of our corpus is what domain experts on the project classify as 'good'. Loghi is still under development and we expect to have digitized data of sufficient quality for our Event Extraction pipeline in the future.

5 Early Modern Dutch and Existing Tools

Since we have seen how unique our corpus is, we can expect existing NLP tools and models to perform poorly on our data. Also, we expect a high degree of variation, so even in cases where modern Dutch is similar, it is extremely hard to generalise. For example, early modern Dutch contains lexical items that no combination of subtokens in a Dutch Language Model (LM) trained on modern data can approximate to represent. An example of this is the word 'natgierig' (having alcoholic tendencies being described as an illness), which would be split in 'nat' (wet) and 'gierig' (greedy). We conducted several preliminary experiments to assess to what degree existing resources could be used to process our data, which we will discuss in this section.

5.1 Predicate Mapping

With a large amount of data to be annotated with a large amount of event classes, it is good practice to adopt heuristic methods to narrow down trigger and type candidates and automatically pre-label to help annotators (Wang et al., 2020). In the NewsReader pipeline (Vossen et al., 2016), events were extracted by linking them to FrameNet frames with the Predicate Matrix (PM) (Lopez de Lacalle et al., 2016). This matrix links entries in WordNet, VerbNet, PropBank, and FrameNet in different languages. As an experiment we tried to apply this approach on our data. We first extracted possible predicates with dependency parsing with spaCy, after which we automatically annotated the possible predicates with the corresponding lemmas and POS-tags by mapping them to a historic Dutch lexicon created by the Institute for Dutch Language (INT) made for OCR and OCR-postcorrection (for the period from 1550 to around 1970)[5]. We proceeded to select the set of lemmas with a verb POS-tag annotation of a mid-frequency range (occurring between 5 and 15 times in the corpus we had available at that time). We then provided those with translations to modern Dutch lemmas manually, using a dictionary that covers Dutch word meanings over several ages (Woordenboek der Nederlandsche Taal, WNT)[6]. Those translations were mapped to the PM and the corresponding FrameNet frames were extracted. We performed a small error analysis of this experiment which showed that the PM produced more false positives (126) than true positives (95)[7]. These results indicate that using existing resources for pre-annotation poses too many issues; we expect that developing our own lexicon for pre-annotation will be more fruitful.

5.2 Zero-shot POS-tagging

In order to see to what extent several LMs are familiar with lexical and syntactic aspects of early modern Dutch sentences, we tested their zero-shot POS-tagging accuracy on sample (1). Measuring zero-shot performance can give us insight into which models are best suited to fine-tune on our event extraction task. We do this by masking each token in the sample one by one and asking the models to fill the masked token each time. We then manually label the predicted tokens with POS-tags and compare these to the gold labels. Gold labels as well as the labelling of the predictions was done by an expert linguist. We also test two Dutch spaCy models.

The LMs we compare are RobBERT (Delobelle et al., 2020), trained on modern Dutch, XLM-R (Conneau et al., 2019), a multilingual RoBERTa model, which outperformed Dutch LMs in an entity labeling task on early modern Dutch in a study by Arnoult et al. (2021), and two versions of GysBERT (Manjavacas and Fonteyn, 2022), a LM pre-trained on historical Dutch. The first version of GysBERT was trained on 7.1B tokens spanning almost 500 years of Dutch data (up to 20th-century Dutch). Early modern Dutch was underrepresented in the training data. The second version of GysBERT was pre-trained in exactly the same way but with the inclusion of 1.3B extra tokens from early modern Dutch datasets, of which 940M tokens from our HTR'ed VOC archival material.

As we can see in Table 1, all scores are low. The second version of GysBERT outperforms all other models but not the best performing spaCy model. It is noteworthy that GysBERT outperforms XLM-R

[4] https://github.com/LAHTeR/htr-quality-classifier

[5] https://taalmaterialen.ivdnt.org/download/tstc-int-historische-woordenlijst/

[6] https://ivdnt.org/woordenboeken/woordenboek-der-nederlandsche-taal/

[7] The full report of this experiment can be found here

and RobBERT but has severely lower performance than GysBERT-v2.

model	accuracy
spacy_sm	.61
spacy_lg	**.66**
RobBERT	.38
XLM-R	.38
GysBERT	.46
GysBERT-v2	.65

Table 1: Zero-shot performance on POS-tagging of sample text in early modern Dutch

Looking at the individual predictions[8], we see a trend where XLM-R predicts very general tokens (adverbs, adpositions, determiners, pronouns, auxiliary verbs, conjunctions). This makes sense since XLM-R is trained to carry general information about several languages and is expected to be stronger when fine-tuned. This is worth further investigation. Parsing example (1) with the best performing model at this task, the largest spaCy model, still shows many issues, for example with 'versoucken' (requesting) being labeled as a noun, 'buijten' (free/outside) and 'bedragen' (amounts) as verbs and 'belastinge' (tax) as an adjective.

The results indicate that existing models seem to have encountered a diverging lexicon and syntactic structure in their training data. Though a base understanding of syntactic structure is necessary for any meaningful NLP task, we want to investigate whether existing models can be useful for other aspects of linguistic modelling.

5.3 Fill-mask for Events

In order to see whether LMs perform better at a semantically relevant task, we check how they fill masked event triggers (to control for cases where a model for example predicts the verb 'receive' in the place of the noun 'reception').

We used all LMs to fill masked event triggers in example (1) and provide results in Tables 11 and 12 in Appendix B. RobBERT, GysBERT and XLM-R all show very poor results. XLM-R and RobBERT do not predict the right token in any of their top 5 predictions for any masked event trigger, nor a token that has a similar meaning, and GysBERT only once. Noteworthy is that XLM-R predicts Dutch words in almost all cases both in this task and zero-shot POS-tagging. It therefore recognises this version of the language as Dutch. Also telling is the fact that RobBERT never predicts any token with a confidence score of above 0.39; for GysBERT this is even lower, namely 0.27. GysBERT-v2 outperforms all models by far.

Existing resources and tools show unpromising results when confronted with our data. Even a model trained on historical Dutch but not on the VOC letters (GysBERT) is enormously outperformed by the exact same model but in which the VOC letters were included in the pretraining (GysBERT-v2). Additionally, pre-annotation methods using existing resources and heuristics also fail. We therefore argue for a new annotation scheme that captures the information we want to extract by clearly establishing i) our model of the world and ii) the way we deal with the variation in sense and reference, since the language in our corpus is often vague and woolly.

6 Annotation

6.1 Task

Annotators are presented with a document and are instructed to label any token or span of tokens that refers to an event that corresponds to one of the 83 event classes described in the ontology of Dynamic and Static events (Verkijk and Vossen, 2023). Apart from event trigger detection and classification, our annotators also labeled participants of each event. Which participants could be annotated for each specific event was specified in our event wiki.[9]

One of the most challenging parts of this task is deciding what it means for a string of tokens to refer to an event class (trigger detection). In order to facilitate the labelling of explicitly described events (directly referring to an event class) as well as implicitly described events (indirectly referring to an event class), we adopt two types of reference. A (span of) tokens either isOfType <eventclass> or evokes <eventclass>. We adopt this distinction from Postma et al. (2020) and Remijnse and Minnema (2020), who propose a very similar distinction for FrameNet annotation. The distinction is important for our annotation task because of the vague language in our data. For example, in (1), 'requested' directly refers to our event class *Request*, while 'order' is a noun that directly refers to an intangible entity, while it evokes a type of *SocialInteraction*. Also, 'to hold' directly refers to keeping something, but evokes *HavingInPosses-*

[8] https://github.com/StellaVerkijk/VarDial2024

[9] https://github.com/globalise-huygens/nlp-event-detection/wiki

sion and also *BeingAtAPlace*. Indirect referrals are essential to extract and model as much important information as possible (i.e., that fits in our model of the world, i.e. the predefined event classes).

The combination of the difficulties of the HTR'ed handwritten language we work with, as illustrated in Section 4.2, the linguistic and historical knowledge needed to annotate, and the inclusion of implicit reference annotation makes our task very challenging. We tried out different annotation settings in order to see what best practices are, which we will describe in the following section.

6.2 Annotation Settings

All annotations were performed by expert historians. They annotated individually in the first setting. Agreement was analysed on annotations of a General Missive of 1628[10], where we noticed that agreement in trigger detection was very low. We asked individual annotators to check each other's annotations, which we will refer to as the *check-task*. This check-task consisted of the following: each annotator was presented individually with all spans annotated by the other annotators as triggering an event, but not by them. They were asked to indicate whether they would now, reviewing it for a second time, also label it with an event class, and if so, with which one. We saw that they often agreed with each other's mention detection – hence, annotators were initially missing event triggers, most probably due to the demanding nature of the task. We further adjudicated the document we performed this experiment on into a test set, which meant we discussed each possible annotation among all annotators and an expert linguist after the check-task. We calculated precision and recall scores for trigger detection (no classification) before and after the check-task compared to the final test set. We see a steep increase in recall scores after the check-task (see Tables 7 and 8 in Appendix B). We therefore performed all further annotations in teams of two, so that annotators can discuss annotations and correct each other. We performed two more annotation rounds in this team setting. After each round, we sharpened the annotation guidelines, taking into account continuous feedback and questions.

[10]National Archive, The Hague, The Netherlands, 1.04.02 (Archive of the VOC), inventory no. 1092, folio 1, r. https://www.nationaalarchief.nl/onderzoeken/archief/1.04.02/invnr/1092/file/NL-HaNA_1.04.02_1092_0017

6.3 Ontological Resolutions

In order to compensate for the difficulty of the annotation task and provide a valuable IAA analysis, we also analyse results after performing two types of automatic resolutions.

Taxonomic resolutions We resolve disagreements on direct subclasses of the same class. E.g., when one annotator labels a token as a trigger for *Leaving* and another labels it for *Voyage*, it is resolved to a *Translocation* annotation (the superclass of *Leaving, Voyage, Arriving and Transportation*). If one annotator uses the superclass (*Translocation*) and another a direct subclass (*Transportation*), it is also resolved to the superclass.

Implicative resolutions The second type of resolution has to do with the implications built in the ontology, modeling how some dynamic events automatically imply a change in state, hence the occurrence of a static event (Section 3). Any event trigger label disagreements where one annotator chose a dynamic event and the other a related static event (e.g., one annotator chose *Attacking* and the other *BeingAtConflict*), the annotation was counted as an agreement and resolved to the static event (*BeingAtConflict*). This was done because the static event is the most conservative meaning (there are often multiple dynamic events that share the same static event as implication).

6.4 IAA Evaluation

Agreement on event mention detection + classification among annotators or annotator teams, presented in Table 2, was calculated with

$$\bar{A} = \frac{1}{2}\left(\frac{A^{xy}}{S^x} + \frac{A^{xy}}{S^y}\right)$$

where A^{xy} is the number of spans both teams labeled with the same event class (using span overlap, not exact span matching), S^x is the total number of spans annotated with an event class by one annotator team and S^y is the total number of spans annotated with an event class by the other annotator team. We then calculate the ratio of agreed upon annotations out of all annotations made by one of the teams. We calculate this ratio for both teams and then take the average of the ratios as our agreement score. We decide to use a simpler calculation than Cohen's Kappa (Cohen, 1960), which includes a chance of accidental agreement in the calculation. Since we have many class types, chance of accidental agreement is quite low. Given that the Kappa score is not transparent and sensitive to skewed

distributions, it is more informative to consider a simple ratio. We also provide results on only class agreement in Table 2. These results were calculated by comparing how often two annotators agreed on the class label, only considering those spans that received a class label from both annotators. In the first annotation round, there were four individual annotators. In the second annotation round, there were three teams of two, of which one new to the task. In the third round there were four, of which also one new to the task. The scores in Table 2 are average scores: for individual comparisons, see Appendix B, Tables 4 to 6.

	Det. + Class.			**Class.**		
	R1*	R2	R3	R1*	R2	R3
Before Resolution	.32	.49	**.57**	.59	.70	.68
After Resolution	.48	.55	**.72**	.91	.78	.86

Table 2: Average agreement scores on event detection + classification (Det. + Class.) and class agreement scores (Class.) between individuals* or teams in different annotation rounds (R = Round) (partial span overlap)

The results show that agreement increased with each round, indicating the task became more clearly defined through several rounds of discussion and reflection. The high score in classification in Round 1 can be explained through the low score in detection: the most obvious event triggers are also easiest to classify. In Round 2, trained teams annotated a total of 79 and 62 triggers respectively, whereas the untrained team annotated a total of only 21 triggers. In Round 3, trained teams annotated a total of 139, 147 and 151 triggers and the untrained team 141. Agreement score on only event trigger detection for the last round was 84% (see Table 9 in Appendix B), while in Round 2 this score was 63% comparing only trained teams and 45% including the untrained team. Note that class agreement is high in spite of a large selection of event classes (more than 80). It is hard to compare our results to IAA scores of other annotated datasets (like ACE) because they either do not evaluate trigger detection, cover much fewer event types, or report on different metrics. Wang et al. (2020) report a Cohen's Kappa score for trigger and type annotation of 38.2% and 42.7% respectively for crowd-source annotation with 168 event types in their contemporary English dataset MAVEN using pre-annotation with heuristics. See Table 10 in Appendix B for an overview of the annotated data we are releasing.

7 Automatic Baselines

The annotations should serve as training data for software that supports event-centric search in the VOC archives. In order to establish a baseline for this, we fine-tuned XLM-R and GysBERT-v2 on our event trigger detection task. Although XLM-R showed disappointing results in our preliminary experiments, it has shown to outperform general LMs at NLP tasks on historical Dutch (Arnoult et al., 2021) and there might be ways to leverage its general knowledge of language in the fine-tuning phase. For this experiment we split the development data we currently have available ('Dev' in Table 10) into a train set of 171KB and a test set of 22KB (json format). We fine-tuned both LMs on a token classification task for event mention detection (binary BIO classification). Since results with fine-tuned versions with early stopping showed low scores, we decided to try the grokking principle (Power et al., 2022; Murty et al., 2023) and evaluate several versions of fine-tuned models trained for increasing amounts of epochs, thereby training far beyond overfitting.

	XLM-R	**GysBERT-v2**
epochs	**P/R**	**P/R**
6	0 / 0	.31 / .06
9	.35 / .24	.20 / .08
12	.40 / .36	.26 / .14
20	.40 / **.43**	.35 / .16
50	.54 / .32	.42 / .20
150	.47 / .32	**.55** / .22

Table 3: Precision and recall scores of fine-tuned models on event trigger detection

Table 3 shows precision and recall scores on token level, which were obtained by mapping the model's prediction of the first sub-token to the complete token. The results show that GysBERT-v2 learns earlier from our data than XLM-R, which is in line with the results of our zero-shot experiments (Section 5). Surprisingly, XLM-R surpasses GysBERT-v2 in recall, and, for several epoch settings, also in precision. GysBERT-v2 eventually reaches slightly higher precision. The results indicate potential to leverage different LMs for different aspects of our task. Users could leverage different LMs at different levels of the system, allowing them to chose a model that suits their needs.

8 Discussion & Conclusion

This paper motivated a newly defined event annotation task by on the one hand discussing ex-

isting literature and theories on event extraction and on the other hand experimenting with existing tools. We show that the early modern Dutch used in the archives of the VOC is different from modern Dutch to such an extent that it calls for a tailored solution. We presented our bespoke annotation scheme and showed that a reasonable IAA could be reached by taking into account annotator needs and following an ontology that allows for the grouping of event classes through inference where necessary. Experiments with baseline automatic solutions for a VOC event-centric search engine show that we need to do more research into what kind of training strategies are needed for this task, and whether grokking can be a solution. Results seem to indicate that both more general LMs and more domain-specific LMs can be useful for different purposes. Future research should include a thorough comparison of different LMs, such as GysBERT and GysBERT-v2. We also aim to create more manually annotated data, develop a domain-specific lexicon for pre-annotation and experiment with automatic data augmentation techniques.

Acknowledgements

This research falls under the GLOBALISE project, funded by the Dutch Research Council (NWO) under project number 175.2019.003. We want to thank our annotators: Kay Pepping, Brecht Nijman, dr. Lodewijk Petram, dr. Manjusha Kuruppath, Femke Brink, Renate Smit, Pascal Konings and Philipp Huber. The code used to fine-tune XLM-R and GysBERT-v2 was an adapted version of code written by dr. Sophie Arnoult. Finally, we thank dr. Lodewijk Petram for proofreading.

References

Jacqueline Aguilar, Charley Beller, Paul McNamee, Benjamin Van Durme, Stephanie Strassel, Zhiyi Song, and Joe Ellis. 2014. A comparison of the events and relations across ACE, ERE, TAC-KBP, and FrameNet annotation standards. In *Proceedings of the second workshop on EVENTS: Definition, detection, coreference, and representation*, pages 45–53.

Sophie I. Arnoult, Lodewijk Petram, and Piek Vossen. 2021. 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*, pages 21–30.

Collin F. Baker, Charles J Fillmore, and John B Lowe. 1998. The Berkeley FrameNet project. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.

Steven Bethard, Guergana Savova, Wei-Te Chen, Leon Derczynski, James Pustejovsky, and Marc Verhagen. 2016. Semeval-2016 task 12: Clinical tempeval. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pages 1052–1062.

Claire Bonial, Susan W Brown, Martha Palmer, and Ghazaleh Kazeminejad. 2021. The rich event ontology. *Computational Analysis of Storylines: Making Sense of Events*, page 47.

Stefano Borgo, Roberta Ferrario, Aldo Gangemi, Nicola Guarino, Claudio Masolo, Daniele Porello, Emilio M Sanfilippo, and Laure Vieu. 2022. Dolce: A descriptive ontology for linguistic and cognitive engineering. *Applied ontology*, 17(1):45–69.

Song Chen et al. 2023. DEFT English Light and Rich ERE annotation LDC2023T04. *Philadelphia: Linguistic Data Consortium.*

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Pieter Delobelle, Thomas Winters, and Bettina Berendt. 2020. Robbert: a dutch roberta-based language model. *arXiv preprint arXiv:2001.06286*.

Yu Hong, Wenxuan Zhou, Jingli Zhang, Guodong Zhou, and Qiaoming Zhu. 2018. Self-regulation: Employing a generative adversarial network to improve event detection. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 515–526.

Paul Kingsbury and Martha Palmer. 2003. PropBank: the next level of treebank. In *Proceedings of Treebanks and lexical Theories*, volume 3. Citeseer.

Maddalen Lopez de Lacalle, Egoitz Laparra, Itziar Aldabe, and German Rigau. 2016. Predicate matrix: automatically extending the semantic interoperability between predicate resources. *Language Resources and Evaluation*, 50:263–289.

Jan Lucassen. 2004. A multinational and its labor force: The dutch east india company, 1595–1795. *International Labor and Working-Class History*, 66:12–39.

Enrique Manjavacas and Lauren Fonteyn. 2022. Non-parametric word sense disambiguation for historical languages. In *Proceedings of the 2nd International*

Workshop on Natural Language Processing for Digital Humanities, pages 123–134.

Shikhar Murty, Pratyusha Sharma, Jacob Andreas, and Christopher D Manning. 2023. Grokking of hierarchical structure in vanilla transformers. *arXiv preprint arXiv:2305.18741*.

Adam Pease, Ian Niles, and John Li. 2002. The suggested upper merged ontology: A large ontology for the semantic web and its applications. In *Working notes of the AAAI-2002 workshop on ontologies and the semantic web*, volume 28, pages 7–10.

Marten Postma, Levi Remijnse, Filip Ilievski, Antske Fokkens, Sam Titarsolej, and Piek Vossen. 2020. Combining conceptual and referential annotation to study variation in framing. In *Proceedings of the International FrameNet Workshop 2020: Towards a Global, Multilingual FrameNet*, pages 31–40, Marseille, France. European Language Resources Association.

Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. 2022. Grokking: Generalization beyond overfitting on small algorithmic datasets. *arXiv preprint arXiv:2201.02177*.

James Pustejovsky. 2021. The role of event-based representations and reasoning in language. *Computational Analysis of Storylines: Making Sense of Events*, page 23.

James Pustejovsky, Catherine Havasi, Jessica Littman, Anna Rumshisky, and Marc Verhagen. 2006. Towards a generative lexical resource: The brandeis semantic ontology. In *LREC*, pages 1702–1705.

James Pustejovsky, Kiyong Lee, Harry Bunt, and Laurent Romary. 2010. Iso-timeml: An international standard for semantic annotation. In *LREC*, volume 10, pages 394–397.

Levi Remijnse and Gosse Minnema. 2020. Towards reference-aware FrameNet annotation. In *Proceedings of the International FrameNet Workshop 2020: Towards a Global, Multilingual FrameNet*, pages 13–22.

Roser Saurí, Jessica Littman, Bob Knippen, Robert Gaizauskas, Andrea Setzer, and James Pustejovsky. 2006. TimeML annotation guidelines version 1.2. 1.

Karin Kipper Schuler. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. University of Pennsylvania.

Roxane Segers, Tommaso Caselli, and Piek Vossen. 2017. The circumstantial event ontology (ceo). In *Proceedings of the Events and Stories in the News Workshop*, pages 37–41.

William F. Styler IV, Steven Bethard, Sean Finan, Martha Palmer, Sameer Pradhan, Piet C. De Groen, Brad Erickson, Timothy Miller, Chen Lin, Guergana Savova, et al. 2014. Temporal annotation in the clinical domain. *Transactions of the association for computational linguistics*, 2:143–154.

Naushad UzZaman, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, and James Pustejovsky. 2013. Semeval-2013 task 1: Tempeval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9.

Stella Verkijk and Piek Vossen. 2023. Sunken ships shan't sail: Ontology design for event reconstruction in the dutch east india company archives. In *Proceedings of the Fourth Conference on Computational Humanities Research*.

Piek Vossen, Rodrigo Agerri, Itziar Aldabe, Agata Cybulska, Marieke van Erp, Antske Fokkens, Egoitz Laparra, Anne-Lyse Minard, Alessio Palmero Aprosio, and German Riga. 2016. NewsReader: using knowledge resources in a cross-lingual reading machine to generate more knowledge from massive streams of news. *Knowledge-Based Systems*.

Christopher Walker et al. 2006. ACE 2005 multilingual training corpus LDC2006T06. *Philadelphia: Linguistic Data Consortium*.

Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020. Maven: A massive general domain event detection dataset. *arXiv preprint arXiv:2004.13590*.

A Example texts

- **Original source, parts of marginalia in boldface and in brackets**[11]

  '(...) op ontfangst van dat „schrijvens, dato 29„e xb: te laten versoecken, dat hij ten Eersten **[zomee conconeren„]** ordre geliefde te stellen, aen wie dat men, de gestipuleerde recognitie goederen nu geeven bal, en niet verpligt, als pro dato soo Lange aen tehouden, sulx thans de zijde buijten belastinge daer van overgaet, en na deesen dat bedra„ „gen, eerst het comptoir generael aengereekend; en ten Lasten gebragt sal kunnen werden(...)'[1]

- **Literal translation**

  (...) on reception of that writing, date 29 xb to be requested that he firstly would like to be put an order, to who that one, the beforely identified taxable goods shall give, and not be obliged, if per the date so long to hold, so that the silk free of tax there be shipped off, and after these the amounts, first the local office general charged; and debited will be, (...)

[11] 'zomee conconeren„' is one line in a marginalium that originally reads 'zo meede concerneerende(...)', meaning *also concerning....*

- **Paraphrased Dutch**
  (...) op ontvangst van die brief, heeft hij op 29 december verzocht om instructies te krijgen aan wie dat men de besproken belastbare goederen zal geven, zodat hij niet gedwongen is de goederen zo lang ter plaatse te laten blijven dat hij daardoor belasting zal moeten betalen over de zijde, wat het lokale comptoir zal worden aangerekend, (...)

- **Paraphrased English**
  (...) on receiving the letter, he requested instructions on the 29th of December as to whom the goods should be given to, so that he will not be forced to keep the goods for such a long time that he would be forced to pay taxes for the silk, for which the regional office would be charged, (...)

B IAA results, data characteristics, fill-mask results

	T1/T2	T2/T1	T1/T3	T3/T1	T2/T3	T3/T2	avg-tr	avg
Before Resolution	.46	.58	.19	.71	.26	.76	.52	.49
After Resolution	.52	.66	.22	.81	.27	.81	.59	.55

Table 4: Agreement between seperate teams in Round 2 on event trigger detection + classification. avg-tr: only trained teams, avg: including the untrained team

	T4/T5	T5/T4	T4/T6	T6/T4	T4/T7	T7/T4	T5/T6	T6/T5	T5/T7	T7/T5	T6/T7	T7/T6	avg
BR	.52	.51	.60	.56	.57	.60	.53	.57	.57	.60	.60	.58	.57
AR	.68	.67	.72	.68	.71	.76	.74	.77	.71	.74	.74	.74	.72

Table 5: Agreement between separate teams in Round 3 on event trigger detection + classification. avg: including the untrained team

	Ann1	Ann2	Ann3	Ann4
Before Resolution				
Ann1	x	.36	.40	.31
Ann2	.26	x	.29	.22
Ann3	.43	.43	x	.34
Ann4	.25	.25	.25	x
After Resolution				
Ann1	x	.59	.49	.54
Ann2	.43	x	.38	.43
Ann3	.54	.59	x	.50
Ann4	.44	.47	.37	x

Table 6: Agreement between individual annotators on event trigger detection + classification (Round 1).

	P	R	n
Ann1	.85	.55	131
Ann2	.95	.35	75
Ann3	.81	.43	108
Ann4	.85	.44	105

Table 7: Precision and recall scores per annotator on event trigger detection before check-task. Gold = test set. n = true + false positives

	P	R	n
Ann1	.83	.81	199
Ann2	.85	.80	192
Ann3	.86	.70	166
Ann4	.82	.71	176

Table 8: Precision and recall scores per annotator on event trigger detection after check-task. Gold = test set. n = true + false positives

T4/T5	T4/T6	T4/T7	T5/T6	T5/T7	T6/T7
.82	.84	.83	.85	.83	.86

Table 9: Event trigger detection agreement in Round 3

	Pages	Docs	Agreement	years
Dev	57	5	59%	1626-1777
Test	5	1	100%	1628

Table 10: Characteristics of annotated data currently processed and of acceptable quality, which includes data annotated by the two trained teams in Round 2 and the adjudicated test set. The third annotation round is currently still in process

	RobBERT				XLM-R			
Masked token	Prediction	Probability	L2C?	FW?	Prediction	Probability	L2C?	FW?
ontfangst	grond	0.26	no	no	grond	0.55	no	no
(reception)	basis	0.13	no	no	basis	0.01	no	no
Getting	een	0.09	no	yes	Grund	0.01	no	no
	straffe	0.05	no	no	aanleiding	0.01	no	no
	elk	0.02	no	yes	grond	0.01	no	no
versoucken	weten	0.15	no	no	weten	0.77	no	no
(requesting)	zien	0.05	no	no	zien	0.04	no	no
Request	staan	0.04	no	no	wissen	0.02	no	no
	toe	0.04	no	yes	merken	0.02	no	no
	zeggen	0.03	no	no	horen	0.02	no	no
ordre	is	0.05	no	no	de	0.21	no	yes
(order/instruction)	om	0.03	no	yes	,	0.07	no	yes
SocialInteraction	,	0.02	no	yes	een	0.04	no	yes
	heeft	0.02	no	no	is	0.03	no	no
	bekent	0.01	no	no	in	0.02	no	yes
geeven	te	0.10	no	yes	te	0.04	no	yes
(giving)	,	0.05	no	yes	reeds	0.02	no	no
Giving	niet	0.04	no	yes	al	0.02	no	no
	kan	0.02	no	no	betalen	0.02	no	no
	sal	0.02	no	no	,	0.02	no	yes
verpligt	meer	0.07	no	no	meer	0.12	no	no
(obliged/ to oblige)	anders	0.02	no	no	langer	0.07	no	no
ForceToAct	is	0.02	no	no	,	0.05	no	yes
	ook	0.01	no	yes	zoo	0.02	no	yes
	zijnde	0.01	no	no	zo	0.02	no	yes
belastinge	,	0.39	no	yes	,	0.30	no	yes
(tax)	en	0.09	no	yes	d	0.04	no	yes
FinancialTransaction	de	0.08	no	yes	s	0.04	no	yes
	ende	0.06	no	yes	en	0.04	no	yes
	daer	0.02	no	no	der	0.03	no	no
aengereekend	is	0.12	no	no	,	0.08	no	yes
(charged)	wordt	0.03	no	no	aan	0.03	no	yes
FinancialTransaction	eert	0.02	no	no	zal	0.03	no	no
	e	0.02	no	yes	naar	0.02	no	no
	int	0.01	no	yes	dient	0.01	no	no

Table 11: Top 5 predicted tokens per model with probability scores. L2C = whether the predicted token is linkable to the corresponding event class. FW = whether the predicted token is a function word.

	GysBERT				GysBERT-v2			
Masked token	Prediction	Probability	L2C?	FW?	Prediction	Probability	L2C?	FW?
ontfangst	ordre	0.17	no	no	ontfang	0.14	yes	no
(reception)	copie	0.10	no	no	antwoorde	0.10	no	no
Getting	grond	0.04	no	no	antwoord	0.08	no	no
	last	0.04	no	no	grond	0.06	no	no
	ende	0.03	no	yes	dato	0.04	no	no
versoucken	weten	0.27	no	no	versoeken	0.70	yes	no
(requesting)	weeten	0.24	no	no	dienen	0.06	no	no
Request	volgen	0.15	no	no	weten	0.04	no	no
	blijken	0.04	no	no	weeten	0.03	no	no
	verstaan	0.03	no	no	versoecken	0.03	yes	no
ordre	,	0.07	no	yes	ordre	0.82	yes	no
(order/instruction)	soo	0.05	no	yes	ordres	0.07	yes	no
SocialInteraction	dat	0.03	no	yes	vast	0.06	no	no
	vast	0.03	no	no	order	0.01	yes	no
	daer	0.02	no	no	uijt	0.00	no	yes
geeven	te	0.13	no	yes	soude	0.04	no	no
(giving)	doen	0.02	no	no	toe	0.03	no	yes
Giving	sal	0.02	no	no	kan	0.02	no	no
	geeven	0.02	yes	no	moet	0.02	no	no
	,	0.01	no	yes	sal	0.02	no	no
verpligt	anders	0.17	no	no	anders	0.41	no	no
(obliged/ to oblige)	meer	0.14	no	no	meer	0.09	no	no
ForceToAct	deselve	0.05	no	yes	langer	0.09	no	no
	die	0.03	no	yes	verder	0.05	no	no
	om	0.03	no	yes	deselve	0.05	no	no
belastinge	,	0.09	no	yes	,	0.05	no	yes
(tax)	cours	0.05	no	no	verwagting	0.05	no	no
FinancialTransaction	die	0.04	no	yes	verantwoording	0.04	no	no
	ende	0.02	no	yes	factuur	0.04	yes	no
	##waerts	0.02	no	yes	gebruijk	0.03	no	no
aengereekend	is	0.07	no	no	belast	0.13	yes	no
(charged)	,	0.02	no	yes	gebragt	0.08	no	no
FinancialTransaction	gebracht	0.02	no	no	overgebragt	0.08	no	no
	overgegeven	0.02	no	no	verantwoord	0.07	yes	no
	gehouden	0.02	no	no	toegesonden	0.04	no	no

Table 12: Top 5 predicted tokens per model with probability scores. L2C = whether the predicted token is linkable to the corresponding event class. FW = whether the predicted token is a function word.