Submitted 05/2017; Published 12/2017

Evaluation of Named Entity Recognition in Dutch online criminal complaints

Marijn Schraagen M.P.Schraagen@uu.nl
Matthieu Brinkhuis M.J.S.Brinkhuis@uu.nl
Floris Bex F.J.Bex@uu.nl

Utrecht University, The Netherlands

Abstract

The possibility for citizens to submit crime reports and criminal complaints online is becoming ever more common, especially for cyber- and internet-related crimes such as phishing and online trade fraud. Such user-submitted crime reports contain references to entities of interest, such as the complainant, counterparty, items being traded, and locations. Using named entity recognition (NER) algorithms these entities can be identified and used in further information extraction and legal reasoning. This paper describes an evaluation of the de facto standard NER algorithm for Dutch on crime reports provided by the Dutch police. An analysis of confusion in entity type assignment and recall errors is presented, as well as suggestions for performance improvement. Besides traditional evaluation based on a manually created gold standard, an alternative assessment method is performed to allow for more efficient evaluation and error analysis. The paper concludes with a general discussion on the use of NER in information extraction.

1. Introduction

Named-entity recognition (NER) is the task of automatically recognising and classifying names that refer to some entity in a text. NER started out as a subtask in the MUC-6 Message Understanding Conference (Grishman and Sundheim 1996), and has since become a standard task in the areas of natural language processing and information retrieval. NER looks for 'unique identifiers of referents in reality', such as persons (*Dwight Eisenhower*), locations (*Amsterdam*), companies (*Google*) or products (*iPhone*).

Very often, NER is partly domain dependent; for example, in the biomedical domain it is desired that the names of genes are correctly classified, and in the context of cyber crime we want to identify email addresses and usernames. In our project 'Intelligence Application for Cybercrime' (Bex et al. 2016), we are developing an intake system for the Dutch police that automatically processes criminal complaints regarding cases of online fraud, such as fake webshops and malicious second-hand traders. Every year there are about 40,000 such complaints filed online, and the high volume and relatively low damages of such cases makes them ideal for further automated processing. The system consists of a dialogue interface that asks the complainant questions about the fraud case (e.g. 'What happened' or 'Which product did you try to buy?'). Because the complainant can answer using free text input, we need to be able to extract the entities (e.g. fraudsters, email addresses, products) so that the correct questions can be asked.

Early approaches to NER were very much rule-based, often combined with gazetteers in which specific entities are listed (Nadeau and Sekine 2007). The problem of this approach is that it involves a lot of manual work, and that rules and lists of entities do not transfer to other domains. Newer approaches typically use supervised machine learning, and for English news texts the task is as good as solved, with F-scores for algorithms close to human scores (around 94%, (Zhou and Su 2002)). However, for these approaches it is also the case that they do not transfer well to other domains or texts which are stylistically and grammatically of lesser quality than news texts, such as email or

Purchase of **Iphone 5s**$_{\text{PRODUCT}}$ on **marktplaats**$_{\text{ORG.LOCATION}}$. 250 euro transferred to the account of **John Doe**$_{\text{PERSON}}$ trusting that he would send the **iphone**$_{\text{PRODUCT}}$ by registered mail. The next day I received a message from **marktplaats**$_{\text{ORGANISATION}}$ that the account of **John Doe**$_{\text{PERSON}}$ is fraudulent. I have therefore transferred money to an account of a swindler named **John Doe**$_{\text{PERSON}}$.

Figure 1: Translated, anonymised example of a crime report, describing a typical fraud case on the online sales platform Marktplaats. Named entities are shown in bold, with the associated entity type in subscript, optionally followed by metonymic type.

other online communications (Poibeau and Kosseim 2001). This poses a problem for our system, where the criminal complaints are filed via online free text forms.

Related approaches for user-generated content (UGC) include normalisation of the data as well as adding features to indicate, e.g., whether a token or a document is correctly capitalised or not. This type of preprocessing generally involves an external database (e.g., Freebase), a semi-structured corpus (e.g., Wikipedia) or a basic gazetteer as a knowledge base, which is used as a reference for capitalisation and punctuation patterns for specific tokens, or as a source for distributional semantic methods (Ritter et al. 2011, Şeker and Eryiğit 2017, Cherry and Guo 2015). However, although these approaches have shown to address specific issues of UGC, the methods still depend on the assumption that the (possibly normalised) entity occurs in some form in an external data source. For UGC published online (oriented towards news, entertainment, general knowledge or other public discussion topics) such as Twitter messages or forum posts, this assumption is likely to be justified in many cases. However, for private data (such as the crime report documents in the current research) many entities cannot be assumed to occur in existing databases or web corpora.

Alternative solutions for the issues encountered with user-generated data focus on human participation through active learning (Tran et al. 2017) or crowdsourcing (Bontcheva et al. 2017). For private data with strict legal constraints, such as crime reports (as in the current research) or medical records, the applicability of such methods is however highly limited.

One other challenge for our project is that much of the research in NER has been performed on English texts, whereas the online criminal complaints we are dealing with are in Dutch. NER for Dutch was for a long time an area with relatively little research, the exception being the 2002 CoNLL shared task on language-independent NER (Tjong Kim Sang and De Meulder 2003). However, relatively recently Desmet and Hoste (Desmet and Hoste 2014) have trained various classifiers on a 1-million token set derived from the Dutch SoNaR corpus (Oostdijk et al. 2013b). This derived set (called SoNaR-1) contains various text types, with a high proportion of autocues, brochures, magazines, newspapers and Wikipedia pages (78%, see (Oostdijk et al. 2013a, Table 2.2)). These classifiers, which reach F-scores of about 80% on news texts similar to the training set, have subsequently been used for the NER module in Frog (van den Bosch et al. 2007), a freely available natural language processing suite.

The objective of this paper is to evaluate how good an "out-of-the-box" NER system for Dutch performs on the online fraud criminal complaints received by the Dutch police. To our knowledge, the NER module for Frog is the only freely available system for Dutch.[1] Testing how it performs on our corpus will give us valuable insights into the state-of-the-art on Dutch NER, and an analysis of the results will allow for the further development of an accurate NER-tagger for our intake system.

[1] https://languagemachines.github.io/frog/

2. Approach

The performance of the Frog named entity recognition module is evaluated using a traditional classification evaluation paradigm. A gold standard is established by manual annotation of test data. The algorithm is applied on the same dataset, and the recognised entities are compared with the gold standard using precision, recall and F-score. However, the classification of a named entity can also be partially correct, therefore multiple performance measures are computed, reflecting various levels of correctness of recognition (see Section 3 for details). Furthermore, an alternative evaluation paradigm is presented to reduce the amount of effort and expertise needed for annotation (see Section 4).

2.1 Annotation of Named Entities

A number of annotation guidelines for named entities have been developed (Chinchor et al. 1999, Linguistic Data Consortium 2008, Brunstein 2002). Typically, named entities are associated with *enamex* entities: persons, organizations and locations. Later annotation guidelines expanded the typology by considering, for example, geo-political entities, products and events, and by considering metonymic usage of entities (e.g. in the sentence 'Spain has won the world cup', *Spain*, which is a geo-political entity, is used metonymically as an organisation, namely the Spanish football team).

For our evaluation, we annotated 250 criminal complaints with the entity types the Frog NER-module recognises: location, person, organisation, event, product and miscellaneous. Two expert annotators manually marked all entities and assigned a type to each entity. Under strict conditions (exact string and entity type equality) the agreement between annotators as measured by Cohen's κ was 0.75. Therefore, during a post-hoc discussion a decision was made on annotation differences and the interpretation of the guidelines. In general, we followed the same annotation guidelines as Desmet and Hoste (2014), but due to these guidelines not being exhaustive we had to make a few additions of our own.

The most important annotation issue to decide was how to annotate web platforms (e.g. *eBay* or its Dutch equivalent *Marktplaats*, but also social networking sites such as *Facebook* and *Instagram*). Sometimes the complainants mean the organisation ('I received a message from Marktplaats'), but often they mean the (virtual) environment ('I met him on Facebook'). It is for this reason that in the latter case, we annotated the entity as *organization, metonymically location*, similar to Schuurman et al. (2010). URLs (also e.g. `Facebook.com`), bank account numbers, email addresses, telephone numbers and usernames were marked as *miscellaneous*. The resulting annotations are used as a manual reference set for algorithm results. Note that the current guidelines result in a broader definition of named entities than used in the Frog training set, especially in the miscellaneous category. The impact of this difference on algorithm results is discussed further in Section 5.

2.2 Data

The data is extracted from a set of crime reports submitted to the official website of the Dutch police, in the domain of internet fraud. These reports contain a free text description of the situation, which often contains a number of named entities describing the counterparty, the product being traded, details on addresses and locations, etc. A report typically contains 1–5 sentences with around 85 tokens per report on average. In Figure 1 an example is provided. Note that in this example the first mention of the organisation *marktplaats* is used metonymically as a location, whereas the second mention is the organisation as such.

The evaluation set of 250 crime reports contains a total of 23,294 tokens, containing 1,059 named entities, the manual reference set. In comparison, Frog's NER-module detected a total of 839 named entities. The distribution of entity types of the manual reference set used for evaluation (non-metonymically) and the Frog NER-module are presented in Figure 2. In this figure we also present the distribution of entity types of the SoNaR-1 corpus as a reference. SoNaR-1 is the corpus used

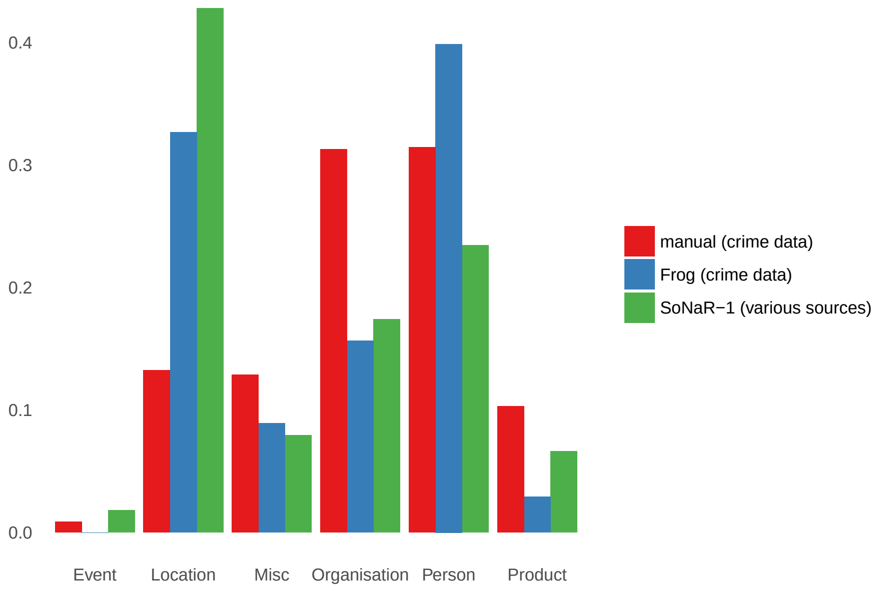


Figure 2: Proportion of entity types in the manual reference set, Frog's NER output, and the algorithm training set SoNaR-1.

by the developers of Frog to train the various components of the tool, among which the named entity module. Using Frog "out-of-the-box", as in the current research, therefore means to use Frog as trained on SoNaR-1. The corpus itself is not used in any way in the current research, only the model resulting from training on this corpus is used as bundled with the default Frog release. As the bundled model determines the coverage, accuracy and possible bias of the classification, the corpus is listed in Figure 2. Most noticeable is the relatively low proportion of locations and the high proportion of organisations in the manual reference annotations. The distributions are however relatively similar, so a category bias is not expected to contribute significantly to the error rate. Section 3 (illustrated in Figure 3) contains a further discussion on type confusion between manual annotation and Frog results.

3. Results

In Table 1 the performance of the algorithm compared to the manual annotation is shown. Recognition errors are characterised in terms of *type*, referring to the predefined entity categories *person, location, organisation, product, miscellaneous*, and *scope*, referring to the detected token sequence boundaries of the entity. Performance is defined for five conditions, namely (1) recognition without considering entity type or scope, (2) recognition with the correct scope, (3) recognition where the recognised type is either the actual type or the type that is used metonymically, (4) recognition where the recognised type is equal to the actual type, and (5) fully correct recognition. Results by entity type are presented in Table 2. For presentation purposes, the performance by entity type is represented as the proportion of items within an evaluation category (summing to 1.0 for each entity type). From these values the various precision and recall scores in Table 1 can be computed.[2] The F-score in the most strict condition equals 0.38. This is considerably lower than the F-score of 0.80 for the NER-module that was reported in Desmet and Hoste (2014), where the algorithm

[2] The evaluation categories in Table 2 contribute in different ways to the values in Table 1, e.g., true positives of the *scope correct* category in Table 1 are composed of both the *correct* and *scope correct, type error* categories in Table 2, and false negatives are composed of the *scope error* and *not recognised* categories.

	category	precision	recall	F-score
1.	entity detected	0.83	0.61	0.71
2.	scope correct	0.63	0.54	0.58
3.	type or metonymic type correct	0.47	0.47	0.47
4.	type correct	0.43	0.45	0.44
5.	scope and type correct	0.35	0.40	0.38

Table 1: Performance of the NER algorithm.

	event	location	misc	organisation	person	product
correct	0.0	0.73	0.09	0.16	0.57	0.00
scope error	0.3	0.14	0.16	0.06	0.23	0.19
scope correct, type error	0.4	0.07	0.14	0.35	0.08	0.25
not recognised	0.3	0.06	0.60	0.43	0.12	0.56

Table 2: Performance by entity type. Type errors are counted only if scope is correct.

was evaluated on a test set similar to the training set (i.e. both the test and training set were part SoNaR-1). Therefore, the current evaluation indicates that the Frog NER-module does not provide adequate performance for unedited non-professional text, in this case user-provided crime reports submitted in an online interface. However, if the type assignment is not taken into account, then both recall and precision increase considerably. In contrast, allowing errors caused by metonymic use does not cause a large increase in score. Furthermore, a substantial number of errors are caused by a difference in the objective of the Frog NER module and the law enforcement application, i.e., several types of entities are considered interesting for the current application, while these entity types are not included in the development of the NER module (see Sections 3.1 and 5 for further discussion). The virtual F-score, i.e., the performance on items for which the algorithm was originally intended, is therefore closer to the previously reported values.

Extending the performance by entity type in Table 2, in Figure 3 a graph is presented to visualise type confusion. The arrows indicate the number of times an entity type recognised by Frog is categorised differently by the experts. A special class here is the *noNE* type, for which outgoing arrows indicate named entities recognised by the experts but not by Frog, and incoming errors indicate named entities falsely recognised by Frog according to the experts. For clarity, the graph excludes arrows with values under 15. Self-directed arrows indicate classifications deemed correct. A few observations stand out in this graph. First, one can see how for the *Product*, *Misc* and *Organisation* categories, the incoming arrow sizes from *noNE* are much larger than the self-directed arrows. Hence, many of these are missed by the Frog NER module compared to the manual reference set. In addition, one can see how *Persons* are classified correctly a lot, but if these are misclassified, they are mostly *Organisations*. A similar observation hold for *Locations*, though these are less often classified correctly. Interestingly, the *Misc* category also has the largest amount of named entities which are not considered to be named entities by the experts.

3.1 Error analysis

A number of main issues can be identified as the source of algorithm errors. Table 3 lists a selection of properties for incorrectly recognised entities, which may explain the results of the algorithm. For example, several company or brand names are not recognised correctly, which could be addressed with a gazetteer. In many domains, a limited set of names may improve recognition significantly. For example, in the current dataset of crime reports, the set *[Marktplaats, Whatsapp, Facebook, Paypal, Google]* accounts for 151 out of 156 misclassified brand name entities. Furthermore, several categories of entities (e.g., international bank account numbers, e-mail addresses, urls, alphanumerical codes)

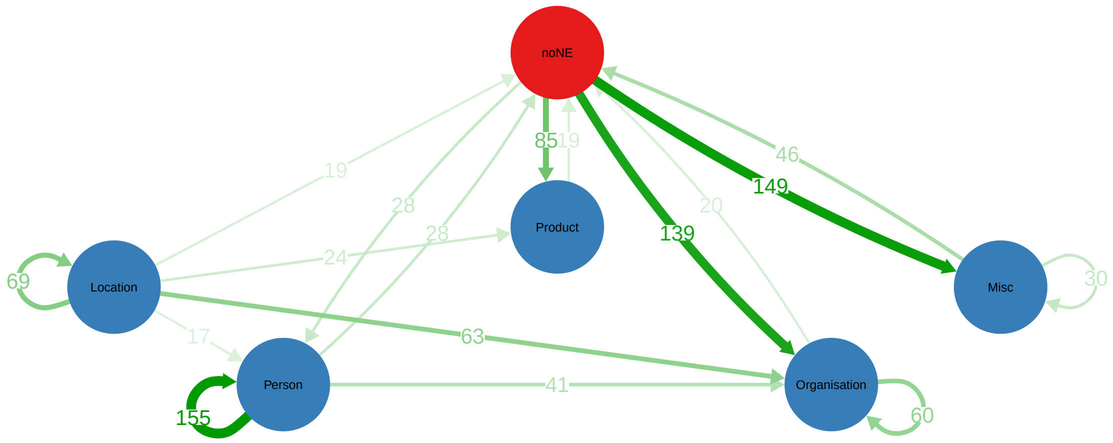


Figure 3: Graph showing type confusion between types recognised by the algorithm and manually assigned types. The full confusion table is included in the Appendix.

property	amount	proportion
brand or company name	156	0.21
capitalisation incorrect	94	0.13
(alpha)numerical code	39	0.05
punctuation incorrect	35	0.05
partial or full url	32	0.04
bank account number	27	0.04
start of sentence capital	26	0.04
e-mail address	24	0.03
bank country code as location	15	0.02
abbreviation	14	0.02

Table 3: Selection of properties of incorrectly recognised items.

that are not annotated in SoNaR-1 are, unsurprisingly, not recognised correctly by the algorithm. This may be remedied by adding such examples to the training set, or by incorporating pattern matching algorithms to the NER approach. However, other causes of error may be more difficult to address. For example, the errors related to incorrect punctuation and capitalisation (found in 18% of all errors) will remain challenging for NER methods in general (cf. Section 1).

Further error analysis is performed on (morpho-)syntactic aspects of entities. Table 4 lists properties regarding capitalization, number of words in the entity, and entities following a selection of function words (mostly prepositions), with associated type errors, i.e., scope errors are disregarded. The table shows that entities written in lower case are generally not recognised correctly, mostly due to failed detection. Entities with the first letter capitalised, which is the default for named entities in professional text, are recognised correctly more often. Full caps and mixed case entities are predominantly incorrect as well, however the frequency of this type of casing is relatively low.

Considering the number of words, NER shows higher performance on multi-word entities. To exclude capitalisation effects, a more detailed analysis is performed on number of words combined with capitalisation. The results are generally consistent with the analysis on number of words only. Interestingly, the difference in performance is highest for entities with the first letter capitalised (for all words), in which case for single-word entities the majority of items is recognised incorrectly,

	no error	*error*		*no error*	*error*	*difference*
capitalisation			*function words*			
lower	7	340	via	3	11	-8
1st upper	239	288	als	0	1	-1
fullcaps	39	96	van	20	18	2
other	0	30	bij	5	6	-1
number of words			in	18	0	18
single	125	550	naar	1	8	-7
multi	160	203	aan	2	2	0
single, lower	0	286	op	2	5	-3
single, 1st upper	107	194	met	5	4	1
single, fullcaps	18	46	per	0	1	-1
single, other	0	24	uit	5	0	5
multi, lower	7	54	te	2	2	0
multi, 1st upper	132	93	voor	0	3	-3
multi, fullcaps	21	50	door	1	1	0
multi, other	0	6	over	1	0	1

Table 4: Error analysis on (morpho-)syntactic properties

whereas for multi-word entities the majority of items is recognised correctly (note that scope errors have not been considered in this analysis).

For entities directly following a function word, some effects can be observed, although it should be noted that the frequency of entities following a word in this set is generally low. For entities recognised by Frog, entities following *in* (English: idem) are always recognised correctly in this dataset. Conversely, prepositions *via* (idem) and *naar* (to) are problematic for the NER algorithm. Other function words do not show large differences.

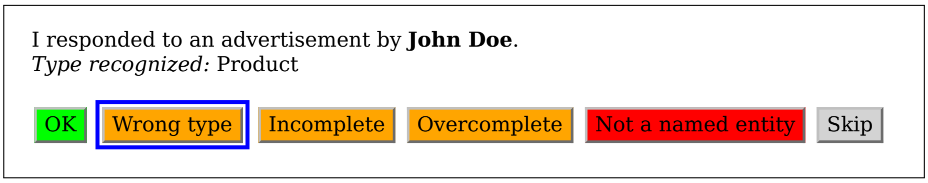
4. Methodology

Manual annotation of named entities in a corpus is a time-consuming activity, which requires linguistic expertise, consistency regarding annotation guidelines, and consultation between annotators. As a methodological consideration, an alternative evaluation method has been performed. In this evaluation the annotator is asked to review named entities as recognised by the algorithm directly, instead of marking entities in the original text. This annotation task is similar to, e.g., treebank development (Marcus et al. 1993) or machine translation-related corpora (Dolan and Brockett 2005), where automatically generated items are presented to human annotators for postcorrection or relevance judgements. The current method is intended to provide a faster annotation process, for which the level of expertise and the difficulty in obtaining consistency within and between annotators is lower than required for manual annotation in the source text. In this section the method is explained in more detail, and the results are compared to the traditional annotation approach.

4.1 Annotation procedure

A pool of four annotators was asked to assess the entities as recognised by the NER algorithm on the first 224 documents, containing 498 entities in total. The annotation interface is shown in Figure 4. This type of evaluation is by design limited to precision, given that false negative items are not present in the evaluation set.

The assessment is based on four categories: fully correct, partially correct, incorrect, or unclear. For partially correct cases, the scope of an entity can be too narrow or too wide, or the type of the



I responded to an advertisement by **John Doe**.
Type recognized: Product

Figure 4: Annotation interface, translated in English for illustration purposes.

error type	*example sentence*
no error	and **John Doe**$_{\text{PERSON}}$ didn't respond to my messages
incomplete	I transferred money to **NL01** ABCD 1234 5678 90
overcomplete	He lives in **Amsterdam The** next day I called
wrong type	On **Marktplaats**$_{\text{PERSON}}$ I bought shoes
not a named entity	Very **bad reviews**.
unclear	talked on WhatsApp: [01/01 10:00] **See You**: thanks

Figure 5: Examples of classification errors

entity can be incorrect. Metonymic types are not considered in this evaluation. Examples of each annotation category are listed in Figure 5. The annotators are instructed that if an item is both assigned a wrong type and the wrong scope, then the scope error should be annotated (except when the recognised scope also constitutes a valid entity[3]). After the instructions, which also include a general introduction on named entities and the enamex types, the annotators are presented with 10 example items and 20 trial items, followed by the 498 items of the evaluation set. In this set duplicate entries are removed (i.e., items consisting of the same string and the same assigned type, occurring in possibly different contexts).

4.2 Classification confidence scores

The classification performed by Frog includes a confidence score for the assigned class, which allows to check whether this confidence is distributed differently between categories, as illustrated in Figure 6. A strong relation between categories and confidence could be used to improve reliability of the application without improving the NER itself. However, though an ANOVA shows that there is a significant difference between the classes, $F(1,5) = 4.3$, $p < .001$, the effect size is small ($\eta^2 = 0.01$), indicating that either the confidence scores are somewhat inaccurate, or that the performance of the NER algorithm itself is insufficient to attach a meaningful interpretation to the confidence scores.

4.3 Inter-annotator agreement

On all annotations the Fleiss' κ multi-rater reliability coefficient is calculated, which ranges from -1 (perfect disagreement) through 0 (no agreement) to 1 (perfect agreement). The value in this set is 0.75, with 4 raters, 497 joint subjects and which is significantly different from 0 ($z = 79.73$, $p = 0$). This is also reflected in the *majority vote*, i.e., the number of annotators that agree for each of the entities (Table 5). An absolute majority is observed in 67.2% of the cases (4 raters agree), a majority in 91.15% of the cases (3 or 4 raters agree), and a tie or minority in the remaining 8.85% of the cases. Agreement values differ between annotation categories, as shown in Table 6. The *Skip* category has the least agreement, which is unsurprising given that the annotators have been specifically instructed to skip an item in case of doubt. It is interesting to note that there is a high

[3] E.g., the sentence "I bought a **Samsung** S6" is incomplete as product name, yet correct as organisation.

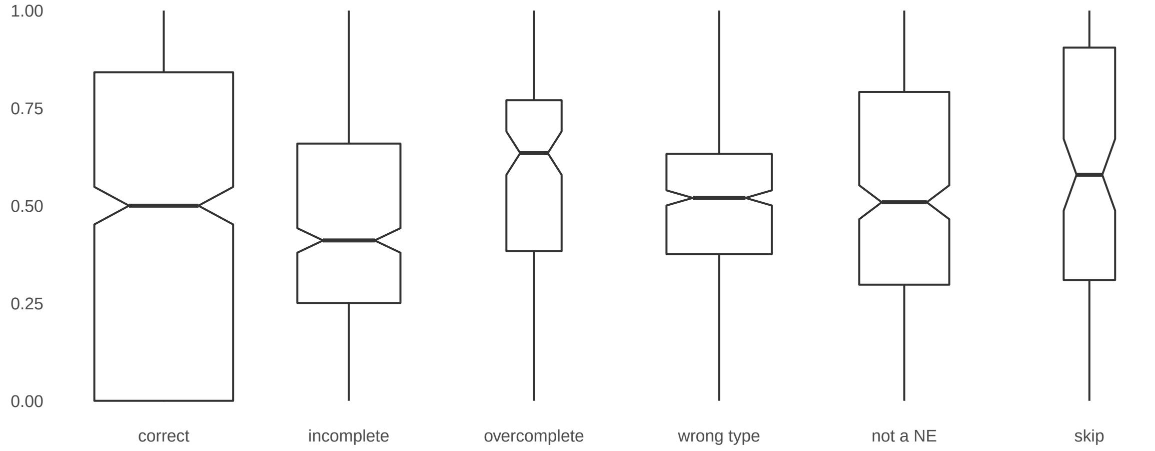


Figure 6: Confidence value vs. evaluation category.

majority	frequency	proportion
1	4	0.01
2	40	0.08
3	119	0.24
4	334	0.67

Table 5: Majority vote distribution

	Fleis κ
Correct	0.85
Incomplete	0.69
Overcomplete	0.70
Wrong type	0.76
Not a named entity	0.79
Skip	0.38

Table 6: Agreement by annotation category

agreement on the *Correct* category, which indicates that agreement on what is right is quite high. However, agreement is lower for cases where annotators consider the classification of the algorithm incorrect, meaning that, while agreeing that an error is made by the algorithm, annotators may not agree on the type of error being made.

4.4 Method comparison

The methodology described in this section is intended to provide an NER evaluation which is less time-consuming and requires less expert knowledge as compared to creating a gold standard reference, while maintaining the validity of the evaluation.

To assess the claim regarding time and knowledge requirements, a separate timing experiment has been performed. A single non-expert annotator has received a ten-minute instruction on the named entity classification scheme and the details of the annotation interface, including a number of introductory items. Next, this annotator was asked to assess 227 entities in the interface, as recognised by Frog in 100 crime reports (note that this includes all entities detected in this set of reports, i.e., the entities are not sampled). This task was completed in 20 minutes (note that the time measurement was not communicated to the annotator). For the same 100 reports an expert annotated all named entities from the source text (containing around 8,500 words in total), by marking entities with a special symbol and an entity type number in a text editor. This task was finished in 60 minutes.[4,5] This experiment suggests that the proposed method is indeed more efficient and

[4] Note that the annotation speed is higher than the 2,000-3,500 words per hour reported by Desmet and Hoste (2010). The annotation scheme is however considerably less complex.

[5] For comparison purposes this gold standard annotation did not include metonymic use.

annotation	all types	person	location	organisation	misc	product
Correct	0.36	**0.49**	0.27	0.43	0.21	*0.10*
Incomplete	0.19	0.19	0.23	*0.12*	0.15	**0.26**
Overcomplete	0.06	0.05	*0.01*	**0.12**	0.08	0.08
Wrong type	0.20	0.15	**0.32**	*0.11*	0.16	0.17
Not a named entity	0.15	*0.08*	0.12	0.17	**0.32**	0.27
Unclear, skip	0.05	*0.02*	0.05	0.04	0.08	**0.12**

Table 7: Results of direct assessment. Highest values per annotation in bold, lowest values in italics.

less knowledge-intensive than creating a traditional evaluation resource. To substantiate the claim, a larger sample would be preferable. For the pool of annotators and the gold standard mentioned above, precise timing measurements are unfortunately not available, however the approximate time used for annotation is consistent with the single-annotator results.

Considering the validity of the evaluation, the results of the direct evaluation experiment are listed in Table 7. The *event* entity type has been excluded because of the low number of occurrences. Compared to Table 1, the precision score of the direct evaluation (0.36) is consistent with the traditional evaluation (0.38).

The two tasks in this comparison, i.e., full annotation and classification assessment, are considerably different in terms of prerequisites and scope. Therefore, it is expected that classification assessment is found to be faster and easier to perform. However, both methods are used to evaluate the precision of the NER process, using the same set of entities, with highly similar results. Because the two methods are used for the same goal, a comparison is useful to quantify the differences in applying the methods. In the current setting, the comparison shows that the performance of the Frog NER results can be adequately evaluated using classification assessment, which uses significantly fewer resources than full annotation.

5. Discussion

The results presented in Section 3 indicate that the performance of the current state-of-the-art algorithm for Dutch NER is not adequate on unedited free-entry data. However, the interpretation of this result depends in part on the intended application of the entity recognition, as there is a clear distinction between the law enforcement goal of information retrieval and the linguistic goal of named entity recognition (Oard et al. 2010). From an IR perspective, entity recognition is a starting point to identify items of interest in a text. For making a case, the police and public prosecutor require references to the criminal, the fraud victim, and the traded product to be identified, as well as details on bank transactions, contact information, advertisement ids and so on. On the other hand, the name of the trading platform, the name of the credit card company or, for example, the location of an event in case of a ticket sale, is of much less interest from a legal perspective. From a computational linguistic point of view, however, the main objective is entity recognition as such, and therefore every item that conforms to the linguistic definition of a named entity is equally important.

This difference occurs on the level of type assignment as well. Entity recognition is a first step of applying NLP, further processing is necessary to identify domain-specific roles in the text (see, e.g., (Carreras and Màrquez 2005)). In these processing steps, it is not always necessary or beneficial to know the correct entity type in advance. For example, the counterparty in a crime report could be either a person or an organisation, possibly metonymically referred to using a web address or username. In this specific case the limited use of type attribution is apparent, however also in the general case the role labelling process is not affected by entity type assignment. Therefore, type errors have to be interpreted differently for information retrieval as a whole than for NER as such, where a type error is considered to be a lack in precision.

6. Future work

The error analysis suggests two parallel approaches for further research. First, pre- and postprocessing, combined with straightforward pattern matching and a comprehensive gazetteer, could solve a significant amount of the observed errors. Second, the algorithm could be re-trained on the data under consideration, which is a common and often necessary practise from a machine learning perspective. The manually annotated reference set could provide a starting point for this training effort. However, a significantly larger amount of training data is required to obtain sufficiently high recognition performance on the current (and similar) datasets.

Considering the law enforcement application as a whole, further research is required to identify relevant types of information in a crime report for police investigation and legal follow-up. For example, complainants provide the name and the bank account of the counterparty in a form accompanying the report. However, as evidenced by the Dutch police and prosecutors, the product or the means of communication with the counterparty (e.g., email, Whatsapp) is often omitted, while this is essential information from a legal perspective. We also aim to integrate a NER-module in the systems of the police, to be applied directly on the 200-300 complaints received daily, intended to allow the system to automatically ask a complainant for further information (e.g. 'which product did you try to buy?') as well as to collect user feedback to further evaluate and train the system.

7. Acknowledgements

This research is funded by the Dutch National Police in the project *Intelligence Amplification for Cybercrime* (IAC).

Appendix

The following table shows the amount of type confusions for the gold standard evaluation method. A selection of these values ($n \geq 15$) is presented as a graph in Figure 3.

	noNE	Organisation	Misc	Location	Person	Product	Event	Sum
noNE	0	139	149	6	28	85	3	410
Organisation	20	60	15	0	3	13	0	111
Misc	46	13	30	2	4	2	1	98
Location	19	63	14	69	17	24	3	209
Person	28	41	15	5	155	15	1	260
Product	19	5	1	1	4	6	0	36
Event	0	0	0	0	0	0	0	0
Sum	132	321	224	83	211	145	8	1124

References

Bex, Floris, Joeri Peters, and Bas Testerink (2016), AI for online criminal complaints: From natural dialogues to structured scenarios, *Artificial Intelligence for Justice Workshop (ECAI 2016)*, p. 22.

Bontcheva, Kalina, Leon Derczynski, and Ian Roberts (2017), Crowdsourcing named entity recognition and entity linking corpora, *Handbook of Linguistic Annotation*, Springer, pp. 875–892.

van den Bosch, Antal, Bertjan Busser, Sander Canisius, and Walter Daelemans (2007), An efficient memory-based morphosyntactic tagger and parser for Dutch, *Selected Papers of the 17th Computational Linguistics in the Netherlands Meeting*, Netherlands Graduate School of Linguistics, pp. 99–114.

Brunstein, Ada (2002), Annotation guidelines for answer types.

Carreras, Xavier and Lluís Màrquez (2005), Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling, *Proceedings of the Ninth Conference on Computational Natural Language Learning*, ACL, pp. 152–164.

Cherry, Colin and Hongyu Guo (2015), The unreasonable effectiveness of word representations for Twitter named entity recognition., *Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL*, pp. 735–745.

Chinchor, Nancy, Erica Brown, Lisa Ferro, and Patty Robinson (1999), *1999 Named Entity Recognition Task Definition*, MITRE and SAIC.

Desmet, Bart and Véronique Hoste (2010), Towards a balanced named entity corpus for Dutch, *7th Conference on International Language Resources and Evaluation (LREC 2010)*, European Language Resources Association (ELRA), pp. 535–541.

Desmet, Bart and Véronique Hoste (2014), Fine-grained Dutch named entity recognition, *Language resources and evaluation* **48** (2), pp. 307–343, Springer.

Dolan, William and Chris Brockett (2005), Automatically constructing a corpus of sentential paraphrases, *Proceedings of the 3rd International Workshop on Paraphrasing*, pp. 9–16.

Grishman, Ralph and Beth Sundheim (1996), Design of the MUC-6 evaluation, *Proceedings of the 6th Message Understanding Conference*, ACL, pp. 413–422.

Linguistic Data Consortium (2008), ACE (Automatic Content Extraction) English annotation guidelines for entities version 6.6.

Marcus, Mitchell, Mary Ann Marcinkiewicz, and Beatrice Santorini (1993), Building a large annotated corpus of English: The Penn Treebank, *Computational Linguistics* **19** (2), pp. 313–330, MIT Press.

Nadeau, David and Satoshi Sekine (2007), A survey of named entity recognition and classification, *LingvisticæInvestigationes* **30** (1), pp. 3–26, John Benjamins publishing company.

Oard, Douglas, Jason Baron, Bruce Hedin, David Lewis, and Stephen Tomlinson (2010), Evaluation of information retrieval for E-discovery, *Artificial Intelligence and Law* **18** (4), pp. 347–386, Springer.

Oostdijk, Nelleke, Martin Reynaert, Véronique Hoste, and Henk van den Heuvel (2013a), *SoNaR User Documentation.* version 1.0.4.

Oostdijk, Nelleke, Martin Reynaert, Véronique Hoste, and Ineke Schuurman (2013b), The construction of a 500-million-word reference corpus of contemporary written Dutch, *Essential speech and language technology for Dutch*, Springer, pp. 219–247.

Poibeau, Thierry and Leila Kosseim (2001), Proper name extraction from non-journalistic texts, *Language and computers* **37** (1), pp. 144–157, Rodopi.

Ritter, Alan, Sam Clark, Mausam, and Oren Etzioni (2011), Named entity recognition in tweets: An experimental study, *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, ACL, pp. 1524–1534.

Schuurman, Ineke, Véronique Hoste, and Paola Monachesi (2010), Interacting semantic layers of annotation in SoNaR, a reference corpus of contemporary written Dutch., *7th Conference on International Language Resources and Evaluation (LREC 2010)*, European Language Resources Association (ELRA), pp. 2471–2477.

Şeker, Gökhan Akın and Gülşen Eryiğit (2017), Extending a CRF-based named entity recognition model for Turkish well formed text and user generated content, *Semantic Web* **8** (5), pp. 625–642, IOS Press.

Tjong Kim Sang, Erik and Fien De Meulder (2003), Introduction to the CoNLL-2003 Shared Task: Language-independent named entity recognition, *Proceedings of the seventh Conference on Natural Language Learning at HLT-NAACL 2003-Volume 4*, ACL, pp. 142–147.

Tran, Van Cuong, Ngoc Thanh Nguyen, Hamido Fujita, Dinh Tuyen Hoang, and Dosam Hwang (2017), A combination of active learning and self-learning for named entity recognition on Twitter using conditional random fields, *Knowledge-Based Systems*, Elsevier.

Zhou, GuoDong and Jian Su (2002), Named entity recognition using an HMM-based chunk tagger, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, ACL, pp. 473–480.