Evaluation of Named Entity Recognition in Dutch online criminal complaints

Marijn Schraagen
Department of Information and Computing Sciences & Department of Humanities
Utrecht University

Matthieu Brinkhuis
Department of Information and Computing Sciences
Utrecht University

Floris Bex
Department of Information and Computing Sciences
Utrecht University

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 eDiscovery 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. The paper concludes with a general discussion on the use of NER in eDiscovery.

KEYWORDS

named entity recognition, evaluation, crime reports

1 INTRODUCTION

Named-entity recognition (NER) is the task of automatically recognizing and classifying names that refer to some entity in a text. NER started out as a subtask in the MUC-6 Message Understanding Conference [8], 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*), referred to as ‘entity name expressions’ or *enamex*. MUC-6 also defines time, currency, and numerical expressions, however these are generally not considered to be named entities[1].

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’ [1], 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 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 [10]. 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%, [15]). 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 other online communications [13]. This poses a problem for our system, where the criminal complaints are filed via online free text forms.

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 [14]. However, relatively recently Desmet and Hoste [6] have trained various classifiers on a 1-million token set derived from the Dutch SoNaR corpus [12]. 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 [2], 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 [7]. 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. Some suggestions for improving performance are presented, however, the focus of the current research is on evaluation of the existing system, and not on improvements as such.

2 APPROACH

The performance of the Frog named entity recognition module is evaluated using a traditional classification evaluation paradigm. A gold standard reference set is established by manual annotation of test data. The algorithm is applied on the same dataset, and the recognized entities are compared with the gold standard using precision, recall and F-score. However, the classification of a named entity can

[1] MUC-6, and later MUCs, refer to these entities as *timex* and *numex*.

DESI'17, London, UK

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.

also be partially correct, therefore multiple performance measures are computed, reflecting various levels of correctness of recognition (see Section 3 for details).

2.1 Annotation of Named Entities

A number of annotation guidelines for named entities have been developed [3, 5, 9]. 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 [6], but due to these guidelines not being exhaustive we had to make a few guidelines 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 organization ('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*. URLs (also e.g. `Facebook.com`), bank account numbers, email addresses, telephone numbers and usernames were marked as *miscellaneous*.

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 around 85 tokens (1–5 sentences). 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.

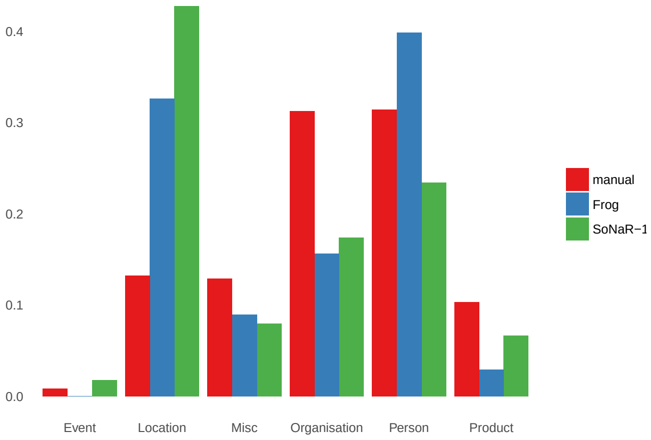

Figure 2: Proportion of entity types in Frog's algorithm output, the manual reference set, and the algorithm training set SoNaR.

The evaluation set of 250 crime reports contains a total of 23,294 tokens, containing 1,191 named entity tokens, the manual reference set. In comparison, Frog's NER-module detected a total of 839 named entity tokens. Note that various named entity tokens occur multiple times (e.g., *Facebook*). 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 manually annotated entity types of the SoNaR corpus (used to train the Frog NER-module) as a reference. The distributions are relatively similar, indicating that the prior probabilities for the entity types are unlikely to negatively affect NER results in the current police corpus. Most noticeable is the relatively low proportion of locations and the high proportion of organisations in the manual reference annotations.

3 RESULTS

In Table 1 the performance of the algorithm compared to the manual annotation is shown. 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. The F-score in the most strict condition equals 0.38. While obviously higher

	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.

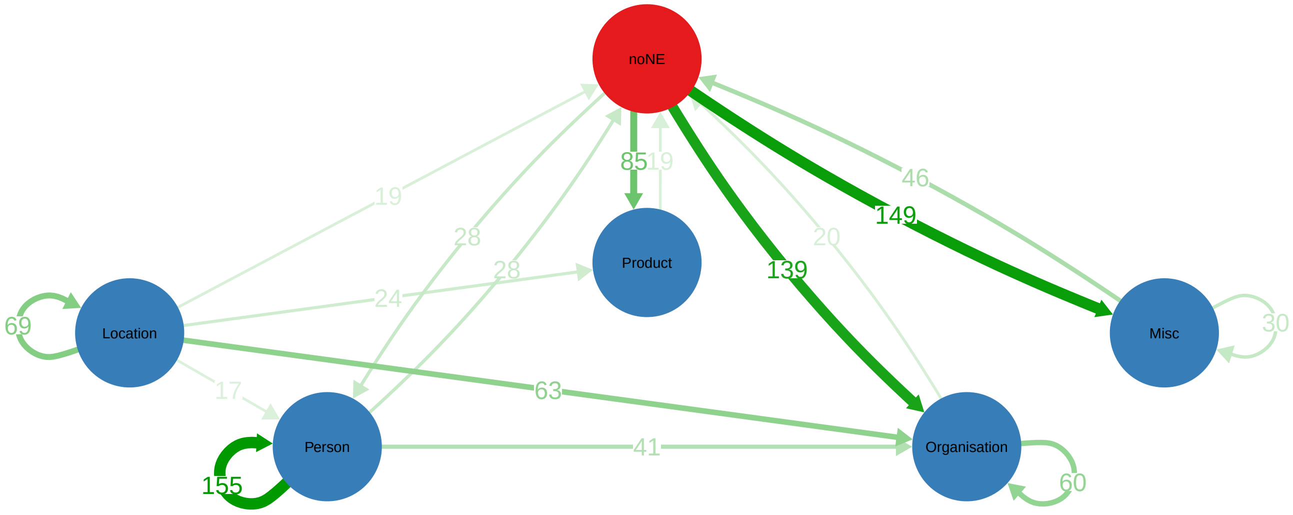

Figure 3: Graph showing type confusion between types recognized by the algorithm and manually assigned types.

than a random baseline, this is considerably lower than the F-score of 0.80 for the NER-module that was reported in [6], where the algorithm was evaluated on a test set similar to the training set (i.e. both sets were part of the SoNaR corpus). 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 or scope 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 application, while these entity types are not included in the development of the NER module (see Sections 3.1 and 4 for further discussion). The performance on items for which the algorithm was originally intended will therefore be closer to previously reported values.

In Figure 3, a graph is presented to visualise errors in main type assignment. The arrows indicate the number of times an entity type recognized by Frog is categorised differently by the experts (e.g., Frog recognised 63 entities as *Location* while the expert annotators labeled these entities as *Organisation*). A threshold of 15 is used for clarity purposes. A special class here is the *noNE* type, for which outgoing arrows indicate named entities recognized by the experts but not by Frog, and incoming errors indicate entities falsely recognized by Frog according to the experts. Self-directed arrows indicate classifications deemed correct. The data underlying this graph, including the node sizes, can be found in the Supplementary Material. 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. 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 has the largest amount of items which are not present in the reference set.

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

Figure 4: Examples of classification errors

3.1 Error analysis

Figure 4 provides an illustration of the classification output of Frog and the errors that are observed. A number of main issues can be identified as the source of algorithm errors. Table 2 lists a selection of properties for incorrectly recognized entities, which may explain the results of the algorithm. For example, several company or brand

property	amount	proportion
brand or company name	156	0.21
capitalization 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 2: Selection of properties of incorrectly recognized items.

names are not recognized 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) that are missing from the original training set of the algorithm are, unsurprisingly, not recognized 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 are more difficult to address. For example, the errors related to incorrect punctuation and capitalization (18% of all errors) will remain challenging for NER methods in general.

4 DISCUSSION

The results as presented in Section 3 indicate that the performance of the current state-of-the-art algorithm for Dutch NER is not adequate on real world data. Indeed, the NER algorithm produces many incorrect results, and a performance improvement is desired. 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 eDiscovery goal of extracting information from text and other goals (e.g. linguistic) [11]. From an eDiscovery perspective, entity recognition is a starting point to identify items of interest in a text. If the police and public prosecutior want to make a case, for example, references to the criminal, the fraud victim, and the traded product must be identified, as well as details on bank transactions, contact information, advertisement id numbers 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 the 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 in the eDiscovery process, further processing is necessary to identify domain-specific roles in the text (see, e.g., [4]). 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 eDiscovery than for NER as such, where a type error is considered as a lack in precision.

5 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 data.

With respect to the larger project context in which the current research takes place [1], we are working on defining exactly which information in the complaints and reports is relevant for the further investigation and legal follow-up. For example, complainants input the name and the bank account of the alleged counterparty into a field in the form (in which also the report is input). However, as evidenced by the Dutch police and prosecutors they often forget to mention the product or the way in which they stayed in contact with the counterparty (email, Whatsapp), while this is essential information from a legal perspective. We also aim to integrate a NER-module in the systems of the police, so that it can directly be used on the 200-300 complaints coming in every day. Thus, if performance is sufficiently high the system can automatically ask a complainant for further information (e.g. 'which product did you try to buy?'). Furthermore, the feedback from users that is gathered in this way can serve to further evaluate and train the system.

6 ACKNOWLEDGEMENTS

This research has been funded by the Dutch National Police in the project 'Intelligence Amplification for Cybercrime' (IAC).

REFERENCES

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

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

[3] Ada Brunstein. Annotation guidelines for answer types. Linguistic Data Consortium, 2002.

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

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

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

[7] Frog, an advanced natural language processing suite for Dutch. https://languagemachines.github.io/frog/.

[8] Ralph Grishman and Beth Sundheim. Design of the MUC-6 evaluation. In *Proceedings of the 6th Message Understanding Conference*, pages 413–422. ACL, 1996.

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

[10] David Nadeau and Satoshi Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007.

[11] Douglas W Oard, Jason R Baron, Bruce Hedin, David D Lewis, and Stephen Tomlinson. Evaluation of information retrieval for e-discovery. *Artificial Intelligence and Law*, 18(4):347–386, 2010.

[12] Nelleke Oostdijk, Martin Reynaert, Véronique Hoste, and Ineke Schuurman. The construction of a 500-million-word reference corpus of contemporary written Dutch. In *Essential speech and language technology for Dutch*, pages 219–247. Springer, 2013.

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

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

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

7 SUPPLEMENTARY MATERIALS

A total of 1,191 named entities have been identified manually from the 250 crime reports. Of these 1,191 named entities, 63 times Frog has made a scope error where the named entity was undercomplete, e.g., two named entities were collapsed into one. Overcomplete errors occurred 4 times. Together, these scope errors compose 6% of the errors, and have been ignored in analyses.

Table 3 provides the numbers as graphically presented in Figure 2. The marginals in this table indicate the relative number of recognised named entities, as presented in Figure 3.

Manual Frog	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

Table 3: Named entity types recognized by the Frog NER-module and recoded by experts.