Uncovering Modus Operandi in human trafficking cases with the use of court convictions

Group F2

Cas Hortensius (Student ID: 11430168), Jan-Willem Tip (Student ID: 13973061), Vera de Brouwer (Student ID: 11299711)
Data System Project 2022 University of Amsterdam

Abstract

In the Netherlands, there is an increase in the number of court cases involving the smuggling of people over the past years. Commissioned by the National Police Academy of the Netherlands, this research investigates whether modus operandi of criminal organizations can be derived from open source data. The focus of this research will be on the smuggling routes used by human traffickers, to discover whether there are any patterns or similarities in smuggling routes. With the help of text analysis techniques, publicly accessible court cases from rechtspraak.nl are investigated. Locations and means of transport are filtered out of the texts to map out a smuggling route. The routes are then compared and clustered in order to recognize patterns. Results show that there are no clear patterns or clusters to be recognized in the extracted routes from the case law.

CCS Concepts

• **Information systems** → *Information retrieval*; • **Human-centered computing** → *Information visualization*; • **Applied computing** → *Law*.

Keywords

Natural Language Processing, Court Conviction Analysis, Named Entity Recognition, Clustering, Human trafficking, Modus Operandi, Rechtspraak.nl

1 Introduction

This project is conducted in cooperation with the Information and Intelligence department of the Dutch Police Academy as one of the stakeholders. The Police Academy is the center for training, knowledge and research for the Dutch Police. As a stakeholder, the Police Academy addressed two (research) questions:

- How are criminals operating?
- How can we make intelligence driven operations more efficient?

The goal is to develop a (quantitative) method to extract (criminal) modus operandi from text and to develop means to map/visualize/interact to further analyze/interact with the data.

In this project different techniques are used for identifying patterns and/or similar ways of operating. The aim of this project is to improve the analysis process within the (criminal) intelligence process. Criminal intelligence is information compiled, analyzed, and/or disseminated in an effort to anticipate, prevent, or monitor criminal activity. The United States Army Military Police Corps [CIA [n.d.]] defines criminal intelligence in more detail; criminal intelligence is information gathered or collated, analyzed, recorded/reported and disseminated by law enforcement agencies concerning types of crime, identified criminals and known or suspected criminal groups. It is particularly useful to deal with organized crime. All phases of the intelligence process are linked and visualized by the Intelligence Cycle. The Intelligence cycle represents four (or five, depending on definition) phases which defines a systematic work process. These phases are Direction, Collection, Processing and Dissemination. In five steps representations the steps are: Planning and Direction, Collection, Processing, Analysis and Production and Dissemination. Within the Netherlands Joint Doctrine Publication [Brouwer and Scholten 2012] the processing phase consists of: Collation, Evaluation, Analysis, Integration and Interpretation. This Modus Operandi project focuses on the improvement processing phase.

Interview at Police Academy

One of the parts of this research was an interview with miss. Melanie Doleweerd, senior teacher Intelligence at the Police Academy. Goal of the interview was to get more insight information in the police intelligence organization, processes and ways how to support police analyst's in their investigations. The Police Academy is responsible for all different types of police education, including Information and intelligence courses. Examples of courses are: Threat- and Future orientation, real time intelligence and scenario building and hypotheses. The police is organized in a one National Police force. The National Police was established in 2013 due to a reorganization where all Police Counties where merged into one National Police force. Due to the previous regional approach, there are still differences in working methodology. The Police Academy tries to establish one way of work methodology. There is an increasing demand for data analysis, but also more analysts are capable to apply data science techniques. The algorithm will contribute to the analyst's tool set because the algorithm can be adjusted to different kind of criminal activities.

Problem definition

This project investigates natural language processing methodologies in order to find Modus Operandi patterns in verdict sentences. Rechtspraak.nl has an extensive databank regarding many different crimes. These crimes can differ from murder, to pick-pocketing, to human smuggling. Human Smuggling is a worldwide issue, with a total of 40.3 millions victims worldwide [Kangaspunta 2007]. In the Netherlands, between 2016 and 2020, there were 4.894 registered victims of human trafficking [Dettmeijer-Vermeulen 2021], with these numbers only expected to rise due to the Corona pandemic [Sanchez and Achilli 2020]. Human Smuggling is the recruitment, transportation, transfer, harbouring or receipt of people through force, fraud or deception, with the aim of exploiting them for profit. Men, women and children of all ages and from all backgrounds can become victims of this crime, which occurs in every region of the world. The traffickers often use violence or fraudulent employment agencies and fake promises of education and job opportunities to trick and coerce their victims [Shelley 2010].

Smuggling routes provide a great insight into how these smugglers operate. Court convictions often contain a description of these routes taken by these smugglers, however, the amount of court convictions, combined with these convictions being filled with difficult jargon, make these smuggling routes hard to detect in unstructured text. Hence, this paper will try to answer the following research question: **To what extent is it possible to identify techniques to discover MO patterns in human trafficking court sentences?**

2 Methodology

Data collection

The first step towards answering the research question is to collect the right data. A report has been made of every legal decision that is made since the beginning of 1900 and is stored on rechtspraak.nl. All these court cases are publicly accessible and can therefore be labeled as open source data. In order to obtain the correct data set, it was first decided to filter the court cases with a search term. All relevant documents returned by rechtspraak.nl for the search term 'mensensmokkel' have been included in the data set that was created for this study. Via an API call to the API of rechtspraak.nl, it was possible to create a list with unique ECLIs (European Case Law Identifier). A second API call was then made for each ECLI to request the complete content per document. Each document was returned by the API in an XML format. By using the BeautifulSoup software in Python, the relevant pieces of text could be extracted from the documents and added to the data frame. This operation resulted in a data frame containing the full text of the court case per unique ECLI.

Exploratory Data Analysis

In total, the above method resulted in a data frame of 2 columns: 'ID', containing the unique ECLIs, and the column 'Text', containing a string of the complete court case. In total, the data frame consisted of 1038 unique court cases, and more than one hundred thousand unique words. Figure 1 shows an annual amount of court cases that fall within our data set.

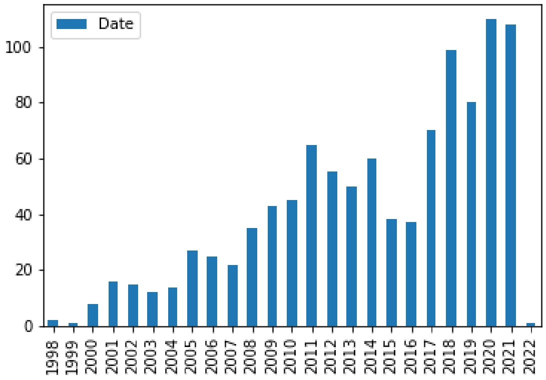

Figure 1: Annual amount of court cases in the dataset

From figure 1 can be deduced that the data set contains court cases that are made in 1998 till court cases that are made in 2022. On average, the most court cases are made in the past five years when it comes to the subject of human trafficking.

Named Entity Recognition

Named Entity Recognition (NER) is a common NLP task. A NER system is trained with predefined categories such as geological locations (Cities, Countries), names of persons, dates, etc. The system is trained on written text, of which the origin differs from blog posts to news articles. NER pipelines are trained on different sources for different languages. For the Dutch language, the NER pipeline is trained on Dutch news articles. [Srinivasa-Desikan 2018]. There are different libraries that allows for the implementation of Named Entity Recognition: GATE, OpenNLP & SpaCy. This paper will use the SpaCy library, due to the open source named-entity visualizer. [Schmitt et al. 2019] NER works by fist identifying entities in the text, and to then label these entities into different categories.

Named Entity Recognition allows for a better understanding of the convictions, with providing nationalities of the people mentioned, the countries and cities involved and other relevant locations. These can consist out of locations such as airports, highways, mountain ranges, bodies of water, etc. In addition to nationalities, countries & cities and locations, in order to be capable to discover routes in these conviction texts, it is also vital to discover *how* the victims of human trafficking are transported. Unfortunately, there are no predefined entities to detect these kind of entities. Entities that would indicate ways of transportation, would be any kind of vehicle, (plane-)tickets, boats, etc.

Extending the NER pipeline

The SpaCy pipeline luckily allows its users to train the pipelines to fulfill the users' specific needs. As mentioned earlier, in order to obtain a Modus Operandi from court documents, it is essential to obtain entities detailing transportation. SpaCy has no current entities that can detail transportation. A way to train the pipeline is by providing training data; sentences that include a new entity, with that entity being labeled manually.

```
TRAIN_DATA = [

    ("Het voertuig is doorzocht op grond
     van de Wet wapens en munitie.",
    "entities: [(4, 12, "VEHICLE")]")

    ]
```

The figure above shows an example of the data send to the pipeline, in order to try to implement the entity "VEHICLE" to the Named Entity Recognition pipeline. Unfortunately, a widely known problem with training NER pipeline is an issue called *'Catastrophic Forgetting'* of previous entity types [Monaikul et al. 2021]. In order to overcome catastrophic forgetting, and still be able to detect Vehicle entities, a data set of transportation vehicles was implemented into the pipeline, to use together with rule-based entity learning. Rule-based entity learning allows for adding entities without the risk of catastrophic forgetting, but has the risk of not having all relevant types of that entity included. In order to be able to label as many variations of transportation vehicles as an entity, lemmatization has been applied to each vehicle in the imported list. In this way, the multiple or diminishing variants of the vehicles can also be recognized by the NER model. This list is obtained from the Dutch dictionary, extracting words describing any type of vehicle. In addition, terms like 'tickets' & 'flight' were also added as a Vehicle entity, since these terms are additional ways of describing transportation methods.

Validation of NER model

A crucial part of any sort of (data)science is to test and validate one's findings. Since the new entity called Vehicle is added, it becomes necessary to test the model to see if the rule-based entity learning has a significant increase in the detection of Vehicle entities. As mentioned earlier, the origin of the imported Vehicle entities is the Dutch dictionary. It can be reasonably assumed that this data source is a valid resource. Precision, Recall & Accuracy are commonly used measurements of validation [Mayer and Butler 1993], these measurements will also be examined here. These measurements require four inputs: True Positive, True Negative, False Positive & False Negative. In the case of validating the NER model, these inputs have the following meanings:

- **True Positive:** Every entity that is correctly labeled by the system
- **True Negative:** Every entity that is correctly not-labeled by the system
- **False Positive:** Entities labeled by the system, but should not have been labeled
- **False Negative:** Entities not labeled by the system, but should have been labeled

A problem that now has to be addressed is that there is currently no 'correctly labeled' court document. In order to find any measurement input, ten court documents are labeled by hand. These documents have not been seen by the system beforehand. After manually going trough these 10 documents, the results are shown in Table 1.

Doc	TP	TN	FP	FN	Recall	Accuracy	Precision
1	65	0	3	15	0.81	0.78	1
2	99	0	6	2	0.98	0.93	1
3	112	0	1	4	0.97	0.96	1
4	93	0	4	2	0.98	0.94	1
5	52	0	1	4	0.93	0.91	1
6	81	0	6	1	0.99	0.92	1
7	37	0	3	4	0.9	0.84	1
8	38	0	1	2	0.95	0.93	1
9	42	0	5	2	0.95	0.86	1
10	53	0	8	9	0.85	0.76	1
Total	**672**	**0**	**38**	**45**	**0.94**	**0.89**	**1**

Table 1: Accuracy, Precision & Recall

First it can be observed that the True Negatives are always 0, with the Precision always being 1 as a direct result. This is because the challenging definition of a True Negative entity. Were this True Negative entity to be counted for every time a word in the court document would *not* be labeled, the amount of True Negative entities would rise to the thousands, completely invalidating the Recall, Accuracy & Precision measurements. For clarity, the Precision measurement is shown in table 1, but Precision is not used in the validation of the system. The Recall and Accuracy validation measurements are very high, which is not surprising after adding a supplemental Vehicle entity, but it does show that the results obtained are reliable.

Extracting possible routes from text

Now that the relevant entities could be recognized by the NER model, the process started to extract possible smuggling routes from the documents. To realize this, it was chosen to filter and save the following categories of entities from the texts: Geolocations, Organizations, Non-Geolocations, Nationalities and our own added entity Vehicle. This filtering is performed per court case and stored in the data frame. An example of the filtered text is as follows: *'Duitsland Zweden Irakese auto auto Maastricht ... parkeerplaats Zweden Duitsland'*. And in English: *'Germany Sweden Iraqi car car Maastricht ... parking lot Sweden Germany'*

In order to give more context to this route, it has been decided to also add the word that stands before the entity in the original text to the filter. Words such as *'from'*, *'to'* and *'by'* can add an extra dimension to a route, for example to map the start and the end of a possible route.

After these words were added, several labeled entities were found to be irrelevant to the route, due to the additional context that this entity had been given by the prefix. For example, *'Amsterdam'* is seen as a Geolocation by the NER model, and is therefore filtered. In most cases, however, this turned out to be *'rechtbank Amsterdam'* which is Dutch for 'Court of Amsterdam'. This is not relevant to the possible smuggling route and can therefore be left out of the filtered text. To achieve this goal, a list of 'prefixes' was created that were found to be irrelevant to the route. This list included words like *"investigation"* and *"court"*. When one of the words from this list was in front of a labeled entity in the original text, both the word and entity were skipped and therefore not included in the filtered text.

With this method, an attempt has been made to skip as much irrelevant information as possible and to store only the relevant information in the data frame.

Similarity

In order to find out if there are human traffic court cases with similar routes, the similarity needs to be calculated. This is done by the using Jaro Similarity: a formula used for measuring the similarity between two strings.

$$\text{Jaro similarity} = \begin{cases} 0 \text{ if m = 0} \\ \frac{1}{3}\left(\frac{m}{|s1|} + \frac{m}{|s2|} + \frac{m-t}{m},\ \text{for m! = 0}\right. \end{cases} \tag{1}$$

where m is the number of matching characters, t is half the number of transpositions & $|s1|$ and $|s2|$ are the lengths of strings s1 and s2, respectively. Essentially, the higher the Jaro Similarity, the lower the distance between two strings. Jaro Similarities are then normalized, so the similarity value will only occur between 0 and 1.

There are many other formulas available for calculating the similarity between two strings. Other examples are the Levenshtein distance, the Hamming Distance and the TFIDF distance metric [Navarro 2001]. None of these formulas however, are optimized for the uncommon string that is the description of the retrieved route. After several attempts at several similarity formulas, it turned out that Jaro Similarity had the most consistent results, and therefore Jaro Similarity is the most reliable method to use.

All obtained routes from the 1038 analyzed court documents are matched one-to-one, with the Jaro Similarity being calculated for all the document pairs. This results in a 1038x1038 symmetric matrix, with a 1 on the diagonal.

Clustering

Finally, a clustering algorithm was applied to the pairwise similarity matrix, in order to discover whether different clusters of smuggling routes exist. For the clustering process it was decided to use Agglomerative Clustering. Agglomerative Clustering is a bottom-up hierarchical clustering method, in which each document is assigned to its own cluster in the first step. Thus, with n number of documents, there are n number of clusters. Step by step, the two most similar clusters are then combined until the algorithm reaches the predetermined number of clusters [Pellegrino et al. 2021]. Agglomerative Clustering was chosen because this algorithm can handle a pairwise similarity matrix as input [Pellegrino et al. 2021].

3 Results

With the chosen methods, it was therefore mainly examined whether different routes, which were extracted from the court documents, show any kind of similarities. As mentioned earlier, an attempt was made to do this by creating a pairwise similarity matrix. This matrix was then converted with a heat map function, in order to be able to quickly observe any clusters by eye. The result of this heat map can be seen in figure 2. An important detail that should be mentioned here is that this figure is a small section of the total heat map. Figure 2 therefore shows a heat map of 10 randomly selected models that have been compared pairwise. From this map, it is not possible to read whether there are possibly different clusters or documents that strongly match when it comes to the extracted route. The complete heat map with 1038 documents also does not give a clear picture, since this map is not readable due to the large number of documents.

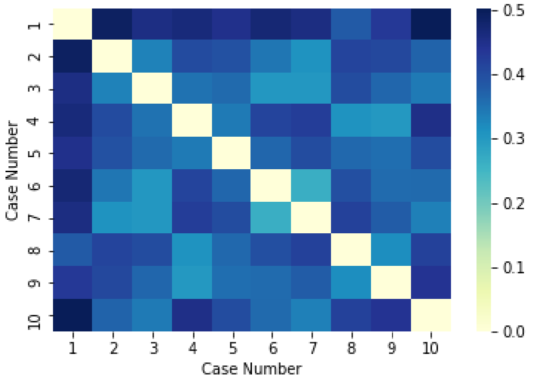

Figure 2: Small section of the complete heat map based on the similarity matrix

In addition to this heat map, Agglomerative Clustering was used to investigate whether underlying clusters of similar court cases, or extracted routes, could be discovered. The result of the clustering can be seen in figure 3. From this it can be deduced that, in addition to a number of outliers, all documents come together in a large group. Thus, no clear clusters can be observed from this clustering method.

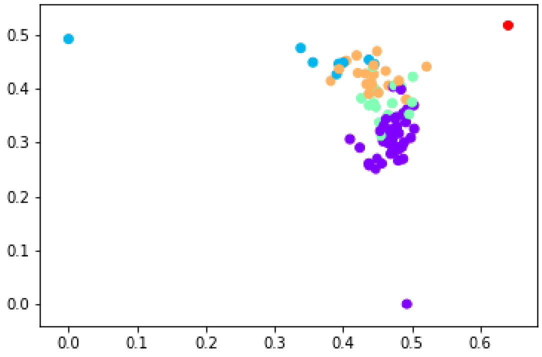

Figure 3: Agglomerative clustering of the similarity matrix

Finally, the number of court cases per year per different mode of transport was examined. This was done to discover whether there are possible trends when it comes to the use of certain means of transport. Figure 4 shows a plot of the number of court cases per year divided over the most popular categories in terms of transport. In general, it can be seen that an increase in each category is observable over the years. The peak in 2018 is particularly striking with a high number of court cases per category. The decline that follows the following year is also remarkable. In addition, from 2016 a significant increase can be observed when it comes to the number of court cases in which it was clear that transport by boat was used. Lastly, it is interesting to note that the 'train' and 'bus' categories follow the same pattern when it comes to the number of court cases per year in which these categories are mentioned as means of transport.

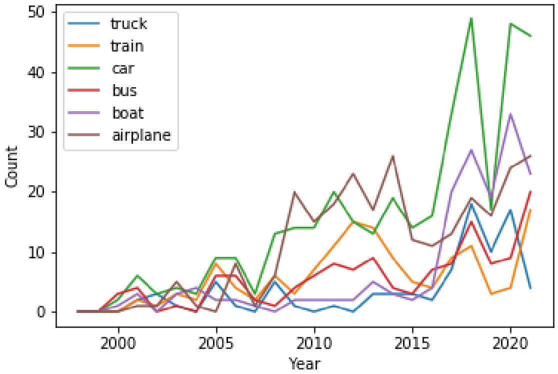

Figure 4: Number of court cases per year per popular transport category

4 Conclusion

To recap, the research question to be answered is the following: **To what extent is it possible to identify techniques to discover MO patterns in human trafficking court sentences?**. This question can be answered with looking at some conclusions derived from the research' results.

The first conclusion that can be derived, is that it is possible to add new and relevant entities, that can extract information specifically needed to analyze court documents. Using rule-based entity learning, it became possible for the Named Entity Recognition system to extract different types of transportation, ranging from cars to plane tickets. These Vehicle entities, together with other entities like Nationalities, Organizations and Geological locations, can then be grouped together to obtain the (smuggling) route described in the court conviction.

After obtaining these routes, clustering court documents in order to find similar smuggling routes proved to be infeasible, due to the layout and design of the retrieved routes. These routes consist of an entity derived by the Named Entity Recognition system, plus the word found before that entity, and many of these derived entities are frequent in many cases. As an example, in order to ensure that the court does not miss any possibility, all possible locations that could have been traveled through need to be mentioned in a court case, but these geological locations provide little to no new information about the truly traveled route.

5 Discussion

As it stands, the achieved routes currently do not provide police analysts with much extra information regarding smuggling. One problem that reoccurred often, is that the actual description of a smuggling routes, is not found in one specific area. If more court document analysis can be done, finding specific patterns in documents that tell where exactly in court documents routes are described, the quality of the derived routes can greatly increase.

Regarding the clustering of court cases, the problem mentioned in the conclusion regarding high general similarity might be tackled by implementing some sort of ‘weight’ to specific words, like ‘via’, ‘towards’, etc. These weights might create a shift in similarity, allowing the clustering algorithm to find more interesting clusters. In addition, the clustering algorithm as is, finds a predetermined amount of clusters. This is not ideal, since there might be less or more actual clusters present, compared to what the algorithm looks for.

Implementing these two described possible improvements of the model, or one of the many non-described improvements, it is of high importance that one implements as little bias as possible. Deciding which words have a higher influence in similarity scores, or discovering a pattern in the layout of a court conviction, can be very easily generalized to an extent where bias becomes too high, resulting in a very low validity of the results.

Possible extensions of model

A way to extent the model, is by adding entities to the used labels that are subject specific. For example, when deriving smuggling routes on the ocean, implement entities like seaport names, boat names, etc. This way, the context of the model decreases, but it might increase how much relevant information can be found. Consulting with experts and asking them which kind of entities they believe to be the most valuable, can prove to be quite fruitful.

Acknowledgments

To *Mellanie van Doleweerd*, Police Academy, for taking the time to answer our question regarding this case.

To *Koen van der Zwet*, TNO, for supervising our project and helping us out when required.

To *Ana Barros*, TNO, for being our stakeholder this project and showing high interest in our project.

To *Peter Bogaard*, Informaat, for teaching us how to take the stakeholders needs and interests into account.

References

P Brouwer and M Scholten. 2012. *Joint Doctrine Publicatie 2 Inlichtingen.*

CIA. [n.d.]. Intelligence Cycle. ([n. d.]).

CE Dettmeijer-Vermeulen. 2021. Mensenhandel in en uit beeld. (2021).

Kristiina Kangaspunta. 2007. Collecting data on human trafficking: Availability, reliability and comparability of trafficking data. In *Measuring human trafficking*. Springer, 27–36.

DG Mayer and DG Butler. 1993. Statistical validation. *Ecological modelling* 68, 1-2 (1993), 21–32.

Natawut Monaikul, Giuseppe Castellucci, Simone Filice, and Oleg Rokhlenko. 2021. Continual Learning for Named Entity Recognition. In *Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence*.

Gonzalo Navarro. 2001. A guided tour to approximate string matching. *ACM computing surveys (CSUR)* 33, 1 (2001), 31–88.

Maria Angela Pellegrino, Luca Postiglione, and Vittorio Scarano. 2021. Detecting data accuracy issues in textual geographical data by a clustering-based approach. In *8th ACM IKDD CODS and 26th COMAD*. 208–212.

Gabriella Sanchez and Luigi Achilli. 2020. *Stranded: the impacts of COVID-19 on irregular migration and migrant smuggling*. European University Institute.

Xavier Schmitt, Sylvain Kubler, Jérémy Robert, Mike Papadakis, and Yves LeTraon. 2019. A replicable comparison study of NER software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate. In *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*. IEEE, 338–343.

Louise Shelley. 2010. *Human trafficking: A global perspective*. Cambridge University Press.

Bhargav Srinivasa-Desikan. 2018. *Natural Language Processing and Computational Linguistics: A practical guide to text analysis with Python, Gensim, spaCy, and Keras*. Packt Publishing Ltd.