Extracting Data from Unstructured Crime Text to Represent in Structured Occurrence Nets using Natural Language Processing

Tuwailaa Alshammari[1]

[1]School of Computing, Newcastle University, Science Square, Newcastle upon Tyne, NE4 5TG, United Kingdom

Abstract

Structured occurrence nets (SONs) are a Petri net-based formalism for representing the behaviour of complex systems, capturing concurrent events and interactions between subsystems. SONs can be used in modelling different applications such as accidents, crime and cybercrime investigation. Using graphical representations can greatly benefit investigators by revealing the causal relationships within crime events. However, identifying crime-related information that allows such modelling from unstructured resources is a challenging task. This paper proposes integrating SONs with Natural Language Processing (NLP) to extract and model crime events accurately. This is done by creating a new custom Named Entity Recognition (NER) model to identify additional crime-related entities like weapons, location, and transportation. Furthermore, we propose a syntactic pattern-based approach for verb identification aimed at generating more precise event results. This method relies on analysing manually created SON crime-related models extracted from crime documents. Our goal is to identify Acyclic Nets (ANs) and construct patterns that facilitate event extraction. The NER model has shown acceptable performance, achieving a precision rate of 83.95%, a recall of 87.45%, and an F1-score of 84.84%, suggesting its effectiveness in NER tasks.

Keywords

structured occurrence net, structured acyclic nets, communication structured acyclic net, natural language processing, event extraction, crime visualisation

1. Introduction

Structured occurrence Nets (SONs) [1, 2] are a Petri net-based formalism for representing the behaviour of complex systems consisting of subsystems that proceed concurrently and interact with each other. SONs extend the concept of an occurrence net, which represents a single 'causal history' and provides a full and unambiguous record of all causal dependencies between its constituent events. In recent years, there has been a research focus on analysing complex systems like cybercrime using SON modelling, with [3] being a notable example.

An extension of SONs is communication structured acyclic nets (CSA-nets) [4] which are based on acyclic nets (ANs) rather than occurrence nets (ONs). A CSA-net joins together two or more ANs by employing buffer places to connect pairs of events from different ANs. Connections of this kind can be either synchronous or asynchronous. When communication is synchronous, events are performed simultaneously. In asynchronous communication, events can be performed

PNSE'24, International Workshop on Petri Nets and Software Engineering, 2024

t.t.t.alshammari2@ncl.ac.uk (T. Alshammari)

either concurrently or sequentially.

It can be challenging for investigators to comprehend a crime and make decisions based on the massive amounts of information acquired during criminal investigations. In crime investigations, sources like police reports and witness statements are used to gather relevant information for analysis. Analysis of such unstructured data sources and extraction of crime events can be facilitated by the use of Natural Language Processing (NLP) techniques. According to [6], NLP originated in the 1950s. The rise of computers necessitated the development of human-machine interaction to enable computers to understand human language through the manipulation and analysis of human text and speech. NLP is a branch of artificial intelligence and linguistics that focuses on teaching computers to recognise and understand texts written in human languages. Currently, various applications such as machine translation, sentiment analysis, and chatbots use NLP. As a result, several natural language processing tools and libraries have been introduced in recent years, such as CoreNLP, NLTK and spaCy. SpaCy is an open-source natural language processing toolkit designed to help developers implement natural language processing annotations and tasks. It is a statistical model known for its ability to analyse text. spaCy provides a range of essential linguistic functionalities, including Part-of-speech (POS) tagging, dependency parsing and Named Entity Recognition (NER). NLP tools are invaluable for analysing the textual representation of natural language. In this work, we extend [5] by integrating SONs with spaCy to extract more accurate information for modelling and visualising criminal events from unstructured textual sources. Investigators often depend on various sources including police reports and witness statements. In this work, we analyse homicide stories [6] to extract information to construct a model in SONs. To be more precise, we use occurrence nets for modelling rather than more general acyclic nets.

The remainder of this paper is organised as follows: Section 2 provides an overview of the research background and related work. Section 3 introduces the basic definitions of acyclic nets and CSA-nets. Section 4 discusses the preliminary rules for mapping text to SON components. Section 4.1 explains the design, experimentation and analysis of the modelling process, presenting both manual and potential automatic models derived from the automatic extraction of an example story. Section 5 discusses the evaluation and testing results of the NER model. Finally, Section 6 concludes the paper and outlines future work.

2. Background and related work

The visual representation of crime events can be helpful for investigation and analysis. [7] outlines the creation of a tool for criminal investigations that employs Twitter data to provide contextual details about crime occurrences in a specific location. This system has been tested as a prototype in the San Francisco region, and it provides a visual representation of criminal incidents and related tweets in the area. This allows users to explore the tweets and crimes that happened before and after a crime incident, as well as to obtain information about the spatial and temporal characteristics of a crime through the internet. Furthermore, [8] highlights data mining methods such as clustering that have been effective in extracting insights from publicly accessible structured data sources such as the National Crime Records Bureau. Additionally, the paper describes an approach for retrieving data from news media through web scraping, as

well as the fundamental NLP techniques for extracting information that is not accessible through typical structured data sources.

Authors in [9] presented WoPeD (Petri net editor and simulator), which has new capabilities for combining Business Processing and natural language processing. WoPeD is an open-source Java application that allows the creation of business processes using workflow nets. The paper demonstrated algorithms for converting graphical process models to textual descriptions and vice versa. Nonetheless, the tool encounters a prevalent problem of semantic ambiguity in NLP.

SON have shown to be effective in accident, criminal and cybercrime investigations. For instance, [10] demonstrated that modelling accident behaviours using SON can help investigators comprehend how an accident occurred and trace the sequence of events leading up to its cause. Similarly, [3] proposed the use of SON features to detect DNS tunnelling during a cyber attack. The authors developed a unique method based on SONs for detecting DNS tunnellings and discussed how the data was pre-processed and eventually translated to a SON. Previous work [5] combined NLP and SONs to extract crime information suitable for modelling. The approach presented three methods to identify events: identifying root verbs, root verbs in conjunction with common crime verbs, and including all verbs to express events. Root verbs refer to the sentence's main verb which acts as the head or root of the sentence's dependency tree. The paper concluded that the most efficient of the three techniques is to combine root verbs with common verbs. However, verbs were identified from crime reports based on their frequency of occurrence, rather than according to the perspective of SON modellers. This paper builds upon the work presented in [5] in two key ways. First, we introduce a set of named entity labels and a trained custom NER model to identify crime entities more accurately. Second, we link verbs to related entities in the same sentence using dependency parsing. Both are discussed in more detail in the following sections.

3. Preliminary

3.1. Acyclic nets and occurrence nets

An acyclic net [11] can be used as a 'database' of empirical facts (both actual and hypothetical expressed using places, transitions, and arcs linking them) accumulated during an investigation. Acyclic nets can represent alternative ways of interpreting what happened, and so can exhibit (backward and forward) non-determinism. An example of acyclic net is occurrence nets, which provides a full and unambiguous record of all causal dependencies between the events it involves. An occurrence net represents a single 'causal history'.

Formally, an *acyclic net* is a triple $acnet = (P,T,F) = (P_{acnet}, T_{acnet}, F_{acnet})$, where P and T are disjoint sets of *places* and *transitions* respectively, and $F \subseteq (P \times T) \cup (T \times P)$ is a *flow relation* such that F is acyclic, and, for every $t \in T$, there are $p,q \in P$ such that pFt and tFq. Moreover, *acnet* is an *occurrence net* if, for each place $p \in P$, there is exactly one $t \in T$ such that tFp, and exactly one $u \in T$ such that pFu.

An acyclic net is *well-formed* if for every step sequence starting from the default initial marking (i.e., the set of places without incoming arcs), no transition occurs more than once, and the sets of post-places of the transitions which have occurred are disjoint. Note that all occurrence nets are well-formed.

3.2. Communication structured acyclic nets

A communication structured acyclic net [11] consists of a number of disjoint acyclic nets which can communicate through special (buffer) places. CSA-nets can exhibit backward and forward non-determinism. They can contain cycles involving buffer places. Formally, a *communication structured acyclic net (or* CSA*-net)* is a tuple $csan = (acnet_1, \ldots, acnet_n, Q, W)$ $(n \geq 1)$ such that:

1. $acnet_1, \ldots, acnet_n$ are well-formed acyclic nets with disjoint sets of nodes (i.e., places and transitions). We also denote:

$$\begin{array}{rcl} P_{csan} & = & P_{acnet_1} \cup \cdots \cup P_{acnet_n} \\ T_{csan} & = & T_{acnet_1} \cup \cdots \cup T_{acnet_n} \\ F_{csan} & = & F_{acnet_1} \cup \cdots \cup F_{acnet_n} \, . \end{array}$$

2. Q is a set of *buffer places* and $W \subseteq (Q \times T_{csan}) \cup (T_{csan} \times Q)$ is a set of arcs adjacent to the buffer places satisfying the following:
   a) $Q \cap (P_{csan} \cup T_{csan}) = \varnothing$.
   b) For every buffer place q:
      i. There is at least one transition t such that tWq and at least one transition u such that qWu.
      ii. If tWq and qWu then transitions t and u belong to different component acyclic nets.

That is, in addition to requiring the disjointness of the component acyclic nets and the buffer places, it is required that buffer places pass tokens between different component acyclic nets. In the step semantics of CSA-nets, the role of the buffer places is special as they can 'instantaneously' pass tokens from transitions producing them to transitions needing them. In this way, cycles involving only the buffer places and transitions do not stop steps from being executable.

A CSA-net $csan = (acnet_1, \ldots, acnet_n, Q, W)$ is a *communication structured occurrence net (or* CSO*-net)* if the following hold

1. The component acyclic nets are occurrence nets.
2. For every $q \in Q$, there is exactly one $t \in T_{csan}$ such that tWq, and exactly one $u \in T_{csan}$ such that qWu.
3. No place in P_{csan} belongs to a cycle in the graph of $F_{csan} \cup W$.

That is, only cycles involving buffer places are allowed.

All CSO-nets are well-formed in a sense similar to that of well-formed acyclic nets. As a result, they support clear notions of, in particular, causality and concurrency between transitions.

In this paper, we use occurrence nets rather than more general acyclic nets. However, this will change in the future work when we move to the next stages of our work where alternative statements in textual documents are taken into account.

4. Towards CSA-net components extraction

This section outlines steps taken to extract useful information from textual resources for modelling and visualising using CSA-nets. Such a visualisation can help investigators comprehend, e.g.,

the dynamics of crime incidents and the interactions among involved parties. Since CSA-nets consist of interconnected acyclic nets, our approach involves first identifying individual acyclic nets and their components, followed by the construction of communication links. Each acyclic net (*acnet*) comprises places (circle nodes) and events/transitions (square nodes), representing the progression of system execution from one state to another. SON models facilitate a visual understanding of the behaviour of complex systems to enhance comprehension.

An example of a complex system is crime investigation, which involves examining numerous variables to assist investigators in decision-making. Extracting information usually involves unstructured data, like written reports, and so presents significant challenges. Investigators often depend on various sources including police reports and witness statements.

CSA-nets provide a method for analysing these types of crimes by representing events and their relationships revealing causal links between them. However, existing CSA-net approaches lack the ability to automatically extract information from written sources and reports. To address this, we leverage statistical Natural Language Processing models by integrating SPACY with CSA-nets to automatically extract useful information from written sources and represent crime events in CSA-nets. In order to enhance the accuracy and consistency of the extraction process, the following rules have been established to guide the linking of natural language with SONs.

- ENTITIES in crime text are the representation of SON acyclic nets. An *"entity"* is a proper noun that refers to a distinct real-world object or entity, such as a person, organisation, or location.
- VERBS represents SON EVENTS/TRANSITIONS within acyclic nets.
- Shared VERBS by several entities will result in the formation of communication buffer places (communication links).

To illustrate how text is represented in SONs, consider the following phrase: *"Allen shot ... "*. Using NLP, the extractor will first identify *Allen* as an entity and then assign the tag 'PERSON'. Next, the extractor searches for verbs related to the entity *Allen*. In this case, the verb *shot* is identified as an event related to *Allen*. This results in the construction of the occurrence net shown in Figure 1.

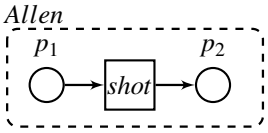

Figure 1: Simple Occurrence net.

4.1. Experiment

The aim is to automatically extract CSA-net components (transitions and acyclic nets) from free textual resources to prepare for modelling. Our initial step involved finding a method to comprehend how SON expert users identify transitions and acyclic nets. To achieve this, we engaged expert users to model stories in SONs, which we evaluated to observe the variations in the resulting models. Following a comprehensive assessment of these models, we recognised

that there is a mapping from English text to formal models, and concluded that one can represent named entities as acyclic nets and map verbs to events/transitions.

Regarding entities, we decided to create a new custom NER model and introduced new labels for the model to learn. Essential entities such as weapons, family members, and transportation are not identified by the default NER model. Therefore, we developed and trained a new custom NER model on these new labels to generate more precise acyclic nets. Subsequently, when evaluating how expert users identified transitions (verbs), we found it to be a challenging task due to the complexity of language structure. Consequently, identifying verb types was not possible, and so we examined the syntactical verb patterns in verbs manually as identified by SON expert users, resulting in the identification of patterns for recognising transitions.

Entity identification:

A customised new NER model was developed to enhance crime entity detection. The model underwent training using a dataset comprising approximately 256 concise crime stories [6]. Through an in-depth analysis of these stories and an accurate manual modelling process, we identified the need to introduce new entity labels to capture key details more effectively.

One noteworthy observation was the frequent mention of weapons, transportation, and family members, which were often undetected by default NER models due to the absence of specifically related labels. In response to these findings, we introduced new entity labels to reinforce the model's accuracy and comprehensiveness. The newly added labels encompass a range of entities including ['LAW_ENFOR', 'WITNESS', 'WEAPON', 'TRANSPORTATION', 'RELATIVE', 'PROFESSION', 'UNNAMEDPER']. (Note: 'UNNAMEDPER' is a NER tag that we propose for tagging unnamed people in the text. For instance, words like 'one woman' and 'third employee' refer to people but are not labelled. Using this tag, we can identify such ambiguous words.) This improvement aims to significantly advance the NER model's ability to recognise diverse crime-related entities, ensuring a more effective analysis of crime narratives. To train the model on these new entities, we collected approximately 334 crime stories, dividing them into 256 for training data and 78 for evaluation. This process involved several steps, including manual annotation of the testing data, training, evaluation, and finally testing of the custom NER model.

The annotation process involves human intervention for identifying and labelling named entities (such as people's names, weapons, locations) in text data according to predefined categories. Annotators highlight entities and assign the appropriate labels using annotation tools such as TagEditor [12]. These annotations provide labelled data for training and testing NER models, enabling accurate recognition and categorisation of entities.

Another challenge arises from the presence of pronouns. To handle this, we employed coreferencing models, which involve resolving pronouns and mentions to their original entity. To address this issue, we integrated NEURALCOREF [13], a SPACY compatible neural network model capable of automatically annotating and resolving coreferences. It is worth mentioning that applying coreference models to texts of considerable size can result in uncertain resolutions.

Verb identification:

Identifying verbs that accurately represent SON events from crime stories presents a significant challenge. To address this challenge, we adopted a pattern-based approach to identify verbs representing events from manually created models. To ensure precision, we examined various crime-related SON models created by expert users, selecting verbs identified by at least 50% of the modellers. Through dependency parsing, we determined the positional relationships of these verbs within sentences concerning surrounding words. Specifically, we identified the types of words and their dependency relationships preceding and following specified verbs by collecting all verbs representing events from manually created models. We then analysed the part-of-speech tags and syntactical tags of the words before and after the verbs. This enabled us to recognise syntactical patterns for event identification, reflecting the human process observed in manual SON models. This approach led to the identification of seven patterns, outlined in Table 1. For instance, the verb may be preceded by a noun, proper noun, or pronoun subject and followed by a noun, proper noun, or pronoun object. The second column in the table indicates the dependency relationship between the words around the verb and the verb itself. For instance, the noun that falls before the verb is a subject, and the noun after the verb is an object.

The pattern approach employs linguistic features to identify verbs representing events. SPACY processes text through tokenisation and linguistic annotations, assigning each token (word) with part-of-speech (POS) tags and dependency labels. These annotations facilitate the accurate identification of verbs, nouns, and other parts of speech, as well as the determination of syntactic relationships within sentences. By analysing manually selected verbs and their associated patterns from expert users, we compiled a list of patterns. If a verb pattern matches any of the patterns in this list, it is considered a SON event. Furthermore, the evaluation of verb extraction is unnecessary, as it relies more on pattern-rule-based techniques rather than statistical predictive identification.

Table 1
List of all patterns identified from the manual models created by SON users [POS]verb[POS].

No	Part-of-Speech Relationship	Dependency Relationship
1	[noun, proper noun, pronoun][EVENT][noun, proper noun, pronoun]	[nsubj, nsubjpass] [dobj, iobj, pobj]
2	[proper noun, pronoun][EVENT][verb, adverb]	[nsubj, nsubjpass] [ccomp, conj, advcl, xcomp, advmod]
3	[verb][EVENT][verb]	[advcl] [advcl]
4	[noun, proper noun, pronoun][EVENT][verb, adposition]	[dobj, iobj, pobj] [ccomp, conj, advcl, xcomp, advmod, pcomp, root]
5	[verb, adverb][EVENT][verb, auxiliary]	[advcl, advomd, ccomp][advcl, conj, ccomp, root]
6	[pronoun][EVENT][noun]	[nsubj] [relcl]
7	[noun][EVENT][verb]	[nsubj] [advcl]

Entity-verb linking:

We applied the NER model to extract entities, and then, by employing the proposed verb-pattern-based identification, we also extracted the verbs. Subsequently, we generated lists for each entity and appended all relevant verbs to each entity list. Algorithm 1 describes the comprehensive extraction process and the assignment of verbs to entities. Figure 2 shows the structure of the output lists (one list for each story), denoted as *ExtractedData_Lists*. Each list within the *ExtractedData_Lists* contains information regarding an entity and its corresponding verbs extracted from the text. For each entity and each verb associate with it, three information items are provided: the verb (v), the location of the verb within the sentence (v_{sent_loc}), and the index of the verb within the text (v_{index}). Following the extraction from each story, a list can contain multiple entities alongside their respective verbs extracted from the text.

$$\begin{aligned} \textit{ExtractedData\_Lists} = \big[& [\textit{entity}^1, v^{1,1}, v^{1,1}_{sent\_loc}, v^{1,1}_{index}, \ldots, v^{1,k_1}, v^{1,k_1}_{sent\_loc}, v^{1,k_1}_{index}], \\ & \vdots \\ & [\textit{entity}^n, v^{n,1}, v^{n,1}_{sent\_loc}, v^{n,1}_{index}, \ldots, v^{n,k_n}, v^{n,k_n}_{sent\_loc}, v^{n,k_n}_{index}] \big] \end{aligned}$$

Figure 2: Extraction output shows how lists are structured.

To transform entity lists into CSA-nets, a sequential acyclic net is constructed to represent each entity list. This involves creating a line-like structure where each verb encountered in the entity list leads to the construction of an event. The construction of acyclic net $acnet_m$ starts by creating an initial place $p^{m,0}$. Then, for every verb $v^{m,i}$ encountered in the $entity^m$ list, a corresponding event $t^{m,i}$ is constructed, followed by the insertion of a new place $p^{m,i}$, as illustrated in Figure 4.

Following the construction of acyclic nets, we use the verb information (i.e., v, v_{sent_loc}, and v_{index}) to construct the communication link later on. This will involve looping and searching for similar verb information among the entities. If such information is found, a communication link is established.

Communication identification:

Communication between different acyclic nets is possible and leads to CSA-nets. Such nets are constructed when transitions in component acyclic nets are connected using buffer places. To accomplish this, we have already extracted verbs along with their corresponding information, such as the specific location within the text and the sentence in which the verb is found. This extracted information enables us to establish communication by identifying identical verbs across different acyclic nets. Upon discovering a match, new buffer places are created to facilitate communication (synchronisation). For instance, if acyclic nets $acnet_m$ (modelling $entity^m$) and $acnet_l$ (modelling $entity^l$) are such that $v^{m,i} = v^{l,j}$, then we add two buffer places, b_{tv} and b_{vt}, and four arcs between transitions t and v (modelling $v^{m,i}$ and $v^{l,j}$, respectively) to ensure their synchronicity (the added arcs are: (t, b_{tv}), (b_{tv}, v), (v, b_{vt}), and (b_{vt}, t)). To improve readability, in the diagrams we depict such an addition of places and arcs by showing just a single buffer place

linked by dashed edges with *t* and *v*, as shown in Figure 4.

Algorithm 1 Entities and Events Extraction Algorithm

Input: Text document
Output: Lists of entities with verbs - [ExtractedData_Lists file]
Read the text file, then process the text using spaCy, **produce** Doc
Initialize *entLabels* list with *NER labels*
Initialize empty lists *names*, *verbs*, *name_info*, and *verb_info*
for each *sentence* in *doc* **do**
    for each *token* in *sentence* **do**
        if token's entity label ∈ *entLabels* **then**
            if *token* **not in** *names* **then**
                Add the *token* in *names* list
        else
            if *token* is **verb** and **in** the *verb pattern* **then**
                if *token* ∉ *verbs* **then**
                    Add *token, sent_number*, and *token index* **to** *verbs* and *verb_info*
Initialize *name_verb_mapping*{}
for each *name* in *names* **do**
    Initialize *name_info* with *name*
    for each *sentence* in *doc* **do**
        if *name* appears in *sentence* **then**
            for each *token* in *sentence* **do**
                if *token* is **verb** and *token* ∈ *verb_info* **then**
                    if *token* matches any *verb* in *verbs* **and** is in syntactical relation with *name* **then**
                        Add *token*, *sent_number*, and *token index name_info*
    Sort *name_verb_mapping* by the order of appearance
write to *FILE*

Modelling Experiment:

To illustrate the applicability of our methodology, consider the following text (STORY A): "Brenton Rhasheem Davis, 24, allegedly shot and killed Tommy Jones III, 19, with a 9mm handgun in the parking lot of a Walgreens in Columbus, Georgia. Witnesses say they saw Davis and Jones enter the store together. When they left toward the parking lot, Jones ran from Davis who then allegedly shot Jones in the leg and back. Davis then walked up to Jones, who had collapsed, and allegedly fired more shots. Jones was shot at least four times. Nine shell casing from the 9mm handgun were found at the scene. A witness added that Davis then picked up a bag Jones had been carrying and then fled the scene. It is unknown what was inside the bag. Jones had gone to the store to meet Davis in order to buy a 45-caliber handgun from him. The two men did not previously know each other. Davis was charged with one count of murder."

Here, we illustrate the diverse modelling approaches applied to STORY A. Figure 3 shows the manually created models, while Figure 4 illustrates a potential automatically extracted model based on the results of the proposed extraction method. It should be noted that humans tend to abstract and therefore selectively omit verbs they consider irrelevant. Conversely, the proposed

verb extraction method aims to identify all verbs present within the pattern, which leads to the generation of larger SON models. However, abstraction may be employed as an option. This could be explored further in future work, with the aim of utilising BSONs as an abstraction method for the produced SONs.

5. Results

Evaluation focused on the NER model using the 20% of the data (78 stories from [6]) that was manually annotated. The data used was similar to the data used to train the model. The precision indicates that roughly 83.95% of predicted entities were correctly identified, highlighting the accuracy of the model's predictions. Additionally, the recall score suggests that the model captures around 87.45% of all relevant entities in the test texts, demonstrating its ability to retrieve relevant information. The F1-score (84.84%) provides a balanced assessment of the model's performance reflecting good identification of relevant entities while minimising false positives and negatives. These results illustrate that the NER model has a good performance indicating its effectiveness across various NER tasks.

Table 2
NER model performance evaluation.

Data Set	Measure	Score %
Validation	Precision	83.95
	Recall	87.45
	F1	84.84
Testing	Precision	69.72
	F1 Recall	77.15
	F1	71.08

Then the model was tested on 54 cases related to police shootings obtained from [14], which was not part of the training dataset. The testing scores reveal acceptable performance, with a precision of 69.72% and a recall value of 77.15%. While the model effectively identifies relevant instances, the F1-score of 71.08% indicates balanced performance. Comparing validation and testing results, there are several possible reasons for the drop in performance, such as overfitting and sample size. However, in this specific case we think that data distribution may be a contributing factor. This means that the patterns learned during training may not closely match the data used in testing. In general, the model demonstrates potential for practical applications. However, depending on the specific crime type, further training and fine-tuning may be required to ensure optimal performance.

6. Conclusion and future work

A new approach to extracting useful information for SON modelling from crime reports is proposed. Initially, entities and verbs are identified to represent and describe the acyclic nets and events. New named entity labels are introduced, and a custom NER model is trained on these

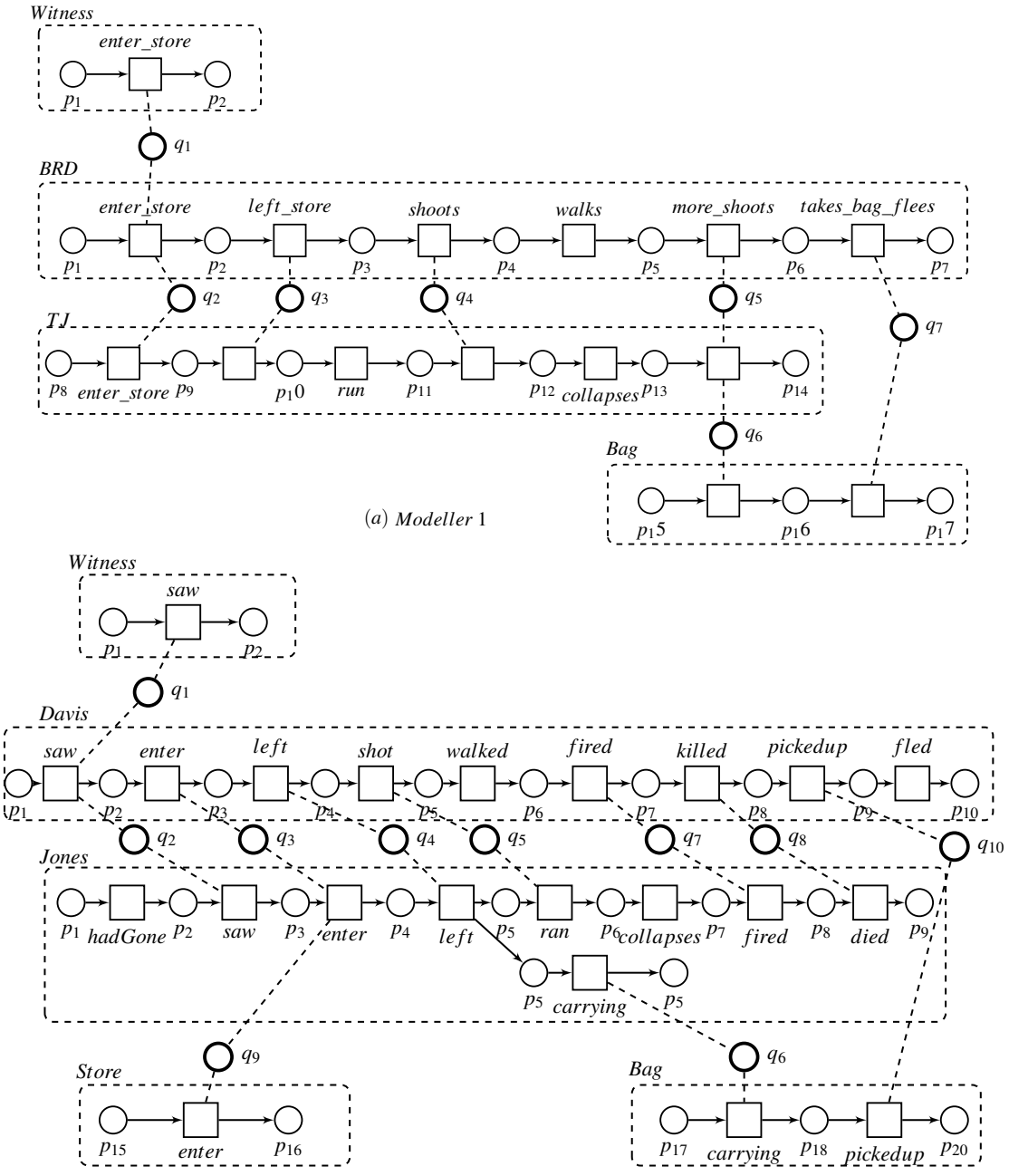

Figure 3: Two models created by different SON expert users (*Modeller* 1 and *Modeller* 2) depict the events from STORY A and demonstrate close entity identification

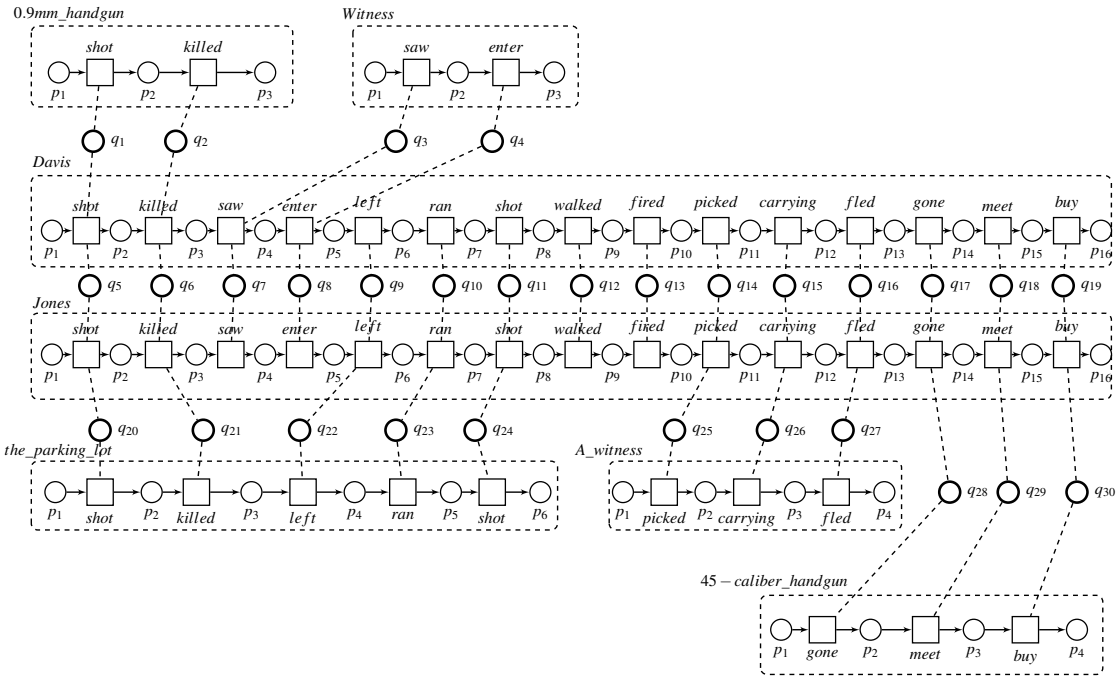

Figure 4: A potential SON model could be developed based on the results of the extraction for the same story (STORY A). The extraction approach identified more entities and verbs compared to human identification in Figure 3.

labels by annotating crime stories. Additionally, verbs are identified using syntactic patterns generated from manually modelled crime documents using CSA-nets. An algorithm for the identification and extraction process is proposed, and the experiment showed acceptable results.

Future work will focus on structurally and formally representing the extracted data and modelling it using SONCraft. Furthermore, improvements to this approach could involve developing new tagger and NER models to enhance accuracy. Additionally, we will investigate the identification and construction of concurrent and conflicting events within acyclic nets.

Acknowledgement

Appreciation is extended to the anonymous referees for their comments and suggestions which have led to improvements in the content and presentation of this paper. Furthermore, sincere gratitude is extended to Maciej Koutny and Anirban Bhattacharyya for their invaluable guidance and unwavering help throughout the development of this paper.

References

[1] M. Koutny, B. Randell, Structured occurrence nets: A formalism for aiding system failure prevention and analysis techniques, Fundam. Informaticae 97 (2009) 41–91.

[2] B. Randell, Occurrence nets then and now: The path to structured occurrence nets, in: L. M. Kristensen, L. Petrucci (Eds.), Applications and Theory of Petri Nets - 32nd International Conference, PETRI NETS 2011, Newcastle, UK, June 20-24, 2011. Proceedings, volume 6709 of *Lecture Notes in Computer Science*, Springer, 2011, pp. 1–16.

[3] T. Alharbi, M. Koutny, Domain name system (DNS) tunneling detection using structured occurrence nets (sons), in: D. Moldt, E. Kindler, M. Wimmer (Eds.), Proceedings of the International Workshop on Petri Nets and Software Engineering (PNSE 2019), volume 2424 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2019, pp. 93–108.

[4] B. Li, M. Koutny, Unfolding CSPT-nets, in: D. Moldt, H. Rölke, H. Störrle (Eds.), Proceedings of the International Workshop on Petri Nets and Software Engineering (PNSE'15), volume 1372 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2015, pp. 207–226.

[5] T. Alshammari, Towards Automatic Extraction of Events for SON Modelling, in: M. Köhler-Bussmeier, D. Moldt, H. Rölke (Eds.), Petri Nets and Software Engineering 2022 co-located with the 43rd International Conference on Application and Theory of Petri Nets and Concurrency (PETRI NETS 2022), Bergen, Norway, June 20th, 2022, volume 3170 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 188–201. URL: https://ceur-ws.org/Vol-3170/paper11.pdf.

[6] Violence Policy Center, [online] Available at: https://vpc.org/. (2022).

[7] P. Siriaraya, Y. Zhang, Y. Wang, Y. Kawai, M. Mittal, P. Jeszenszky, A. Jatowt, Witnessing crime through tweets: A crime investigation tool based on social media, in: F. B. Kashani, G. Trajcevski, R. H. Güting, L. Kulik, S. D. Newsam (Eds.), Proceedings of the 27th ACM International Conference on Advances in Geographic Information Systems, SIGSPATIAL 2019, Chicago, IL, USA, November 5-8, 2019, ACM, 2019, pp. 568–571.

[8] S. Chakravorty, S. Daripa, U. Saha, S. Bose, S. Goswami, S. Mitra, Data mining techniques for analyzing murder related structured and unstructured data, American Journal of Advanced Computing 2 (2015) 47–54.

[9] T. Freytag, P. Allgaier, Woped goes NLP: conversion between workflow nets and natural language, in: W. M. P. van der Aalst et. al (Ed.), Proceedings of the Dissertation Award, Demonstration, and Industrial Track at BPM 2018, volume 2196 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2018, pp. 101–105.

[10] B. Li, Visualisation and Analysis of Complex Behaviours using Structured Occurrence Nets, Ph.D. thesis, School of Computing, Newcastle University, 2017.

[11] M. Alahmadi, S. Alharbi, T. Alharbi, N. Almutairi, T. Alshammari, A. Bhattacharyya, M. Koutny, B. Li, B. Randell, Structured acyclic nets, CoRR abs/2401.07308 (2024). doi:10.48550/ARXIV.2401.07308. arXiv:2401.07308.

[12] I. TagEditor, Tageditor annotation tool, 2020. URL: https://github.com/d5555/TagEditor, accessed: (2022).

[13] NeuralCoref, Neuralcoref 4.0: Fast coreference resolution in spacy with neural networks, https://github.com/huggingface/neuralcoref, 2022.

[14] Fatal Encounters, [online] Available at: https://fatalencounters.org/. (2024).