

Towards Automatic Extraction of Events for SON Modelling

Tuwailaa Alshammari¹

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

Abstract

Data visualization is the process of transforming data into a visual representation in order to make it easier for human to comprehend and derive knowledge from. By offering a detailed overview of crime events, data visualization technologies have the potential to assist investigators in analysing crimes.

This paper proposes a new model that takes advantage of statistical natural language processing technologies to extract people's names and relevant events from crime documents for SON modelling and visualisation. The proposed extractor is examined and evaluated. It is argued that it achieved reasonable results when data is extracted for SON modelling when compared with human extraction.

Keywords

structured occurrence net, structured acyclic nets, communication structured acyclic net, natural language processing, event extraction, model building

1. Introduction

Structured occurrence nets (SONs) [1, 2] are a Petri net based formalism for representing the behaviour of complex systems consisting of interdependent subsystems which proceed concurrently and interact with each other. This extends the concept of an occurrence net, which represents a single 'causal history' and provides a full and unambiguous record of all causal dependencies between the events it involves. An example of a complex system is (cyber)crime and its computer based representation and analysis gained considerable research attention in recent years using, in particular, the SON model [3].

An extension of SONs are the 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. The nature of such connections can be synchronous or asynchronous. In a synchronous communication, events are executed concurrently, whereas in asynchronous communication, events may be executed concurrently or sequentially.


One of the main challenges in conducting effective criminal investigations is the overwhelming amount of data, which makes it challenging for investigators to comprehend the crime and, therefore, make decisions. In particular, investigators rely on a variety of sources of information during criminal investigations, including police written reports and witness statements, which may contain information that needs to be extracted and analysed. This is performed by connecting

PNSE'22, International Workshop on Petri Nets and Software Engineering, Bergen, Norway, 2022

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



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

and analysing the different aspects of the crime in order to comprehend the causality behind crime events.

Natural Language Processing (NLP) can help in the analysis of such unstructured data sources and extraction of crime events. NLP began in the 1950s, according to [5]. With the invention of the computer, it became necessary to build human-machine interaction relationships in order to teach computers how to interpret real human language through manipulation and analysis of human text and speech. NLP is defined as a sub-field of Artificial Intelligence and Linguistics that focuses on teaching computers to recognise and interpret text, statements, and words written in human languages. NLP is now employed in a number of applications, including machine translation, sentiment analysis, and chatbots. As a result, several natural language processing tools and libraries have emerged in recent years, including CORENLP, NLTK, and SPACY (used in the work presented here). SPACY [6] is an open-source natural language processing toolkit that was created to assist developers in implementing natural language processing annotations and tasks. It is a statistical model that excels at data extraction and text preparation for deep learning. As with other NLP libraries, SPACY has a variety of valuable linguistic features, including Part of Speech (POS) tagging, a dependency parser, and a Named Entity Recognizer (NER). This paper proposed integration of SONS with NLP in the extraction and modelling of crime events. The idea is to extract useful information from unstructured written sources in order to analyse and visualise in SONS.

This paper is organised as follows. Section 2 provides an overview of research background on SON and NLP. Section 3 presents basic definitions concerning SONS. Section 4 presents the extraction and modelling of events using SONS done by human. We then introduce TEXT2SON, our proposed automatic extraction approach. Section 5 discusses and compares the results and shortcomings for both manual and automatic extraction. Section 6 concludes the paper and provides an overview of future work.

2. Background

In this section, we look at the related work that have been used to attempt to solve the mentioned challenges through the use of graph analysis and visualisation, NLP, and data extraction for crime data. In particular, the work done in crime modelling in SONS as well as crime data extraction using various NLP techniques.

SONs demonstrated promising results for accident, criminal and (cyber)crime investigations. [7] demonstrates an explicit capture of the accident behaviours for multiple sub-systems by modelling it in SONS. It showed the ability of SONS to aid an investigator in comprehending how the accident occurred and tracing the sequence of events leading up to the accident cause. Moreover, [3] suggests the use of SON features to detect DNS tunneling during an actual attack. A unique method for detecting DNS tunneling based on SONS has been created and implemented. Additionally, pre-processing of data and a set of experiments were discussed.

The paper [8] introduced new WoPeD capabilities for integrating NLP and Business Processing. WoPeD (Petri net editor and simulator), is an open source Java application for building business processes using workflow nets. Algorithms have been presented for converting graphical process models into textual process descriptions and vice versa. However, the tool suffers from the

common issue of semantic ambiguity in natural language processing (in both directions).

News and social media are taking considerable focus in information extraction and classification. [9] describes the development of a crime investigation tool that leverages Twitter data to aid criminal investigations, by providing contextual information about crime occurring in a certain location. A prototype has been implemented in the San Francisco region. This system provides users with a spatial view of criminal incidents and associated tweets in the area, allowing them to investigate the various tweets and crimes that occurred prior to and following a crime incident, as well as obtaining information about the spatial and temporal characteristics of a crime via the web. [10] presents data mining techniques, such as clustering, which have been shown to be useful in extracting insights from publicly available structured data (National Crime Records Bureau). Additionally, an approach for retrieving data via web scraping from news media has been presented, as well as the essential NLP techniques for extracting significant information that is not available through typical structured data sources.

A continuing work on Simple Event Model ontology [11] discusses populating instances extracted from crime-related documents, aided by an SVO (Subject, Verb, Object) algorithm that extracts events using hand-crafted rules. The study employs the SVO algorithm to generate SVO triples by parsing crime-related words using the MALTPARSER5 dependency parser and then extracting SVO triples from the parsed sentences.

3. Preliminaries

In this section, we recall basic definitions of the SON model needed in the rest of the paper.

Acyclic nets and occurrence nets

A acyclic net is 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 has happened, and so may 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.

Communication structured acyclic nets

A communication structured acyclic net consists of a number of disjoint acyclic nets which can communicate through special (buffer) places. CSA-nets may 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 $csan = (acnet_1, \dots, acnet_n, Q, W)$ ($n \geq 1$) such that:

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

$$\begin{aligned} P_{csan} &= P_{acnet_1} \cup \dots \cup P_{acnet_n} \\ T_{csan} &= T_{acnet_1} \cup \dots \cup T_{acnet_n} \\ F_{csan} &= F_{acnet_1} \cup \dots \cup F_{acnet_n} . \end{aligned}$$

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) $Q \cap (P_{csan} \cup T_{csan}) = \emptyset$.
 - b) For every buffer place q :
 - i. There is at least one transition t such that tWq .
 - 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 $csan = (acnet_1, \dots, acnet_n, Q, W)$ 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 and CSO-nets rather than more general acyclic nets and CSA-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. Extraction and Modelling

Crime can be conceptualised as a complex evolving system characterised by the occurrence of numerous relevant and linked variables. Such systems require the examination and comprehension

of behaviour to assist investigators in the decision-making process. Investigators typically rely on a variety of sources, including written police reports and/or witness statements. CSA-nets provide a distinctive method for analysing such types of crimes via representing events and chain of events in order to uncover causal relationships between them. Also, CSA-nets can assist in better comprehension and visualisation of events. Our work relies on integrating NLP techniques with CSA-nets in order to extract useful information from written sources, and representing crime events through CSA-nets. This integration aims at the development of an automatic extraction tool (TEXT2SON) for criminal cases leveraging statistical NLP models.

4.1. Human extraction: an experiment

Extracting information from unstructured data, such as written investigation reports, aims to extract valuable information that could aid investigators in analysing and comprehending the dependencies between crime events. CSA-nets are one of the potential techniques for visually representing data in order to assist investigators in analysing and identifying causality among these occurrences.

The existing CSA-net approach lacks the ability to automatically extract information from (unstructured) written sources and reports. Figure 2 illustrates the outcome of extracting information and representing it by three expert SON users. The experiment was focused on a short fragment of a crime story displayed in Figure 1. The users were asked to extract and represent crime events as a SON model. In addition, we were interested in observing the style of human extraction and modelling processes in order to determine the consistency of the models and the amount of time spent.

ROSS AND SPICER HAD PLAYED A GAME OF DICE. ROSS LOST AND HE WAS UPSET,
ACCORDING TO POLICE RECORDS. THE NEXT DAY HE, WEARING BODY ARMOR,
RETURNED TO SPICER'S HOME AND FATALLY SHOT SPICER

Figure 1: A short fragment of a crime story

The users extracted the following verbs from the sentences: play, lost, leaves, wearing, goes/returned, and shoot. Nevertheless, not all users agreed on the exact model design and in the terms of wording. For example, Modeller1 extracted only three verbs (play, lost, and shot), whereas Modeller3 extracted five verbs noting the insertion of verbs that were not explicitly mentioned in the sentences. Modeller3 added the words (leaves and goes) that may be explained by the human capacity to comprehend and express events differently.

Despite minor representational discrepancies (for example, the extent of information provided by different modellers), the experiment revealed semantically similar models. In comparison to other modellers, Modeller1 extracted just enough data. Modeller1 extracted and presented the offense in a very straightforward manner by extracting two entities and three verbs. Modeller2, on the other hand, added an additional entity, ON: DICE, that the other two modellers did not,

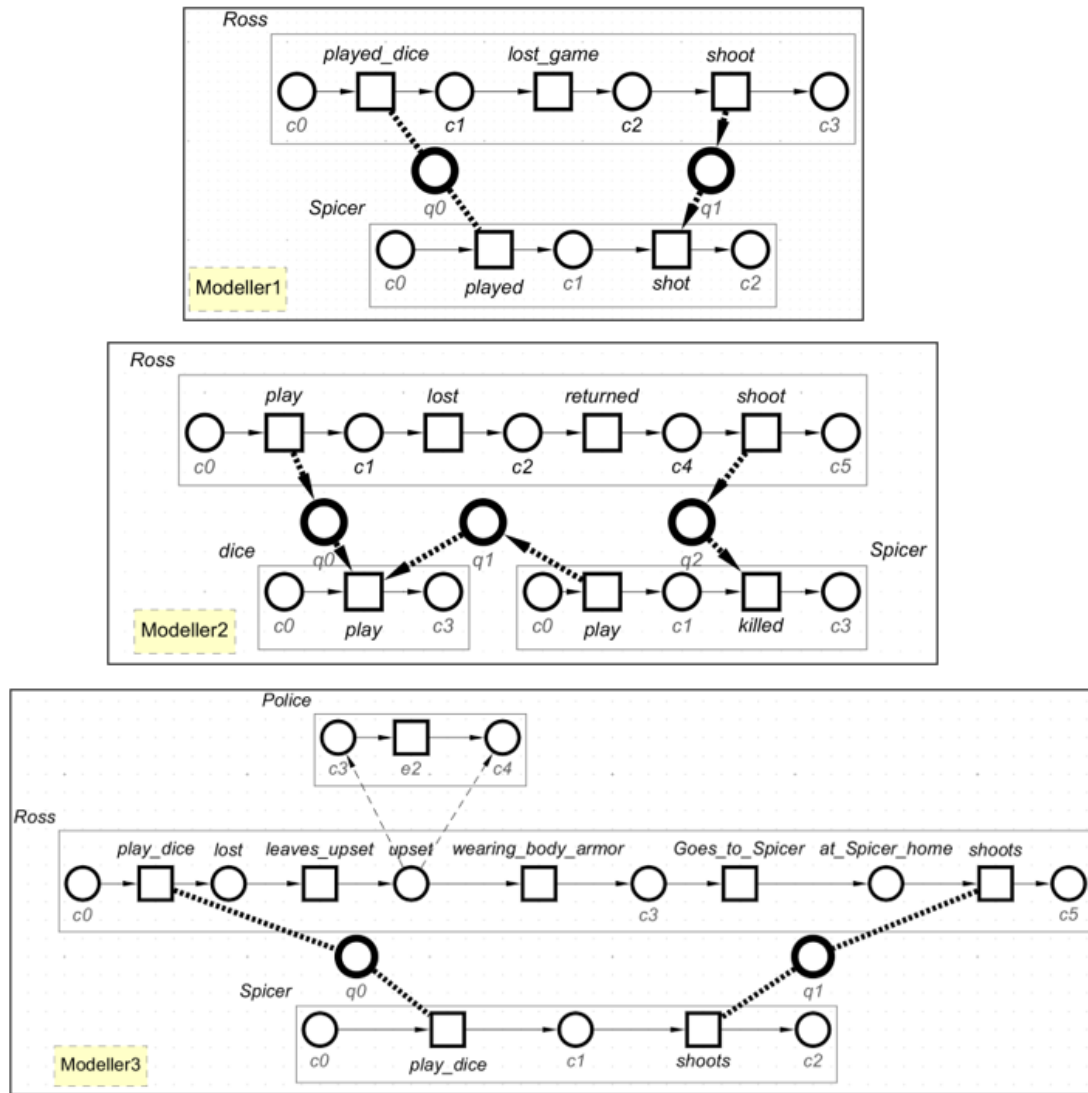


Figure 2: Human modelling done by SON expert users

instead inserting DICE into the play event, PLAYED_DICE and PLAY_DICE, by Modeller1 and Modeller3, respectively.

4.2. Automatic extraction: TEXT2SON

With the above in mind, our goal is to develop a tool to extract crime events and relationships between crime events, and build a model in SONS for behaviour analysis and visualisation. We have applied three methods of extraction to identify and evaluate the most accurate method

compared to human extraction presented in the previous section. Initially, we only considered extracting the main verb that SPACY's parser nominates as the main verb of a sentence, tagged as ROOT. ROOT tags appear once in every sentence representing the main word carrying the meaning of the sentence is (usually a verb). We then considered extracting more information, by evaluating the most frequently occurring verbs in the reserved data set (note that from the evaluation of around 570 crime stories we compiled a list of most frequently occurring verbs). We then tested the second method by extracting both ROOT verbs and verbs that match common verbs. Finally, we considered extracting all other verbs present in the text, including verbs tagged as ROOT.

4.2.1. Terminology

Tokenization: the process of splitting words in a sentence into a series of tokens.

Part-of-Speech (POS): assigns part of speech tags.

Dependency Parsing: links tokens (words) as they grammatically appear in the sentence and assigns them parsing tags that shows their relationships, i.e., subject, object, conjunction, etc.

Named Entity Recognizer (NER): is a model where entities are identified within a text and tagged with name types.

Coreferencing: resolve pronouns and mentions to the original names they refer to.

ROOT verbs: main verbs appear in sentences and are predicated by the PARSER as the primary word from which the sentence is parsed.

Occurrence Net (ON): an acyclic net which provides a full and unambiguous record of all causal dependencies between the events it involves.

4.2.2. Main assumptions and preliminary rules

In order to extract data automatically, we propose to extract entities PEOPLE and verbs. In our methodology, we consider entities as representations for ONS, and verbs as representations of EVENTS within the ONS. We will also consider that the shared EVENTS between different ONS represent potential synchronised communications, and we connect them formally using channel places.

A number of assumptions has been put in place in order to carry out the extraction. We first assume that verbs tagged as ROOT verbs represent events (and each occurs exactly once). Then, since ROOT verb appears only once in a sentence, we assume that events are represented by either verbs tagged as ROOT verbs, or are verbs in most frequently occurring verbs list. This is due to the possibility of more than one event occurring in the same sentence.

4.2.3. The Extractor

The proposed TEXT2SON extractor utilises the statistical models provided by SPACY. The algorithm in Figure 3 extracts people names present in a text, by searching for names in every sentence. Following this phase, we analyse every sentence by searching for presence of people

Algorithm 1 Extraction

```

1: Input: text document
2: Output: Lists of entities with events - [entities_with_verbs file]

3: Step 1: read document Text
4: Step 2: pass the Text into spaCy NLP pipeline
5: Step 3:
6: Create entity_check[] list                                ▷ entities list to avoid duplication
7: Create allOns[][] list
8: for S in Text do                                           ▷ every sentence in the Text
9:   for word in S do                                           ▷ every word in the sentence
10:    if word is ent & not in entity_check[] then
11:      add word to entity_check[]
12:      create new_list[] & add allOns[][]
13:      add word in new_list[0]                                ▷ add the entity in position 0
14:    else
15:      continue
16:    end if
17:  end for
18: end for
19: for list_value[] in allOns[][] do
20:  get list_value[0]
21:  for word in S do
22:    if word equals list_value[0] then
23:      for word in S do
24:        if word equals root then
25:          add to list_value[]
26:        end if
27:      end for
28:    end if
29:  end for
30: end for

```

Figure 3: Extractor design

names in each sentence and the occurring verbs labelled as ROOT. Once found, they are grouped together in a list. This process is repeated until the text reaches its end, resulting in lists of people names and their associated events (verbs). This is required because we regard names to be the representations of occurrence nets (i.e., each name is associated with exactly one ON), and verbs to be the representations of events.

Figure 4(a) demonstrates the tools used to create the TEXT2SON extractor and their respective versions. We utilised SPACY, a Python library that makes use of pipeline packages with key

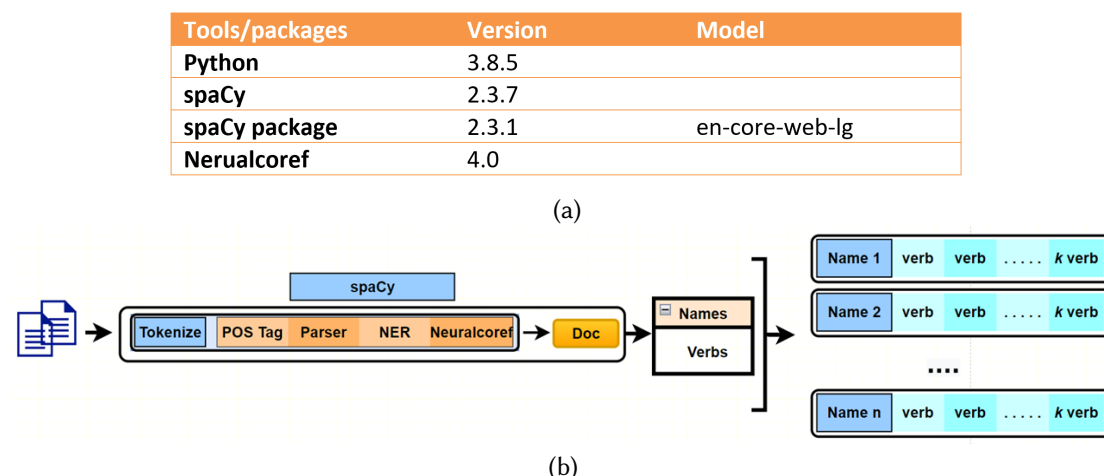


Figure 4: Tools for extractor design (a); and extractor design (b)

linguistic features such as a TAGGER, PARSER, and NER.

Figure 4(b) illustrates the steps involved in the extraction process. To begin with, text data is fed into SPACY’s pipeline, where it is tokenised into individual tokens. The tokens are then passed to SPACY’s tagger, which assigns anticipated POS tags, depending on the pre-trained model predictions. The parser next assigns tags indicating the relationships between the tokens. The NER assigns labels to identified entities, which may be individuals, organizations, or dates, to mention a few. However, for this part of the research, we are interested only in the PEOPLE tag. Additional tags will be included in future versions of the tool.

Among the difficulties is resolving the pronouns to the correct previously named individual. To address this issue, we integrated NEURALCOREF [12], a SPACY compatible neural network model capable of annotating and resolving coreferences. Figure 5 shows how crime example sentences from Figure 1 are modified after applying NEURALCOREF. More precisely, we can observe the replacement of the pronoun HE with the person’s name ROSS.

ROSS AND SPICER HAD PLAYED A GAME OF DICE. ROSS LOST AND **ROSS** WAS UPSET, ACCORDING TO POLICE RECORDS. THE NEXT DAY **ROSS**, WEARING BODY ARMOR, RETURNED TO SPICER’S HOME AND FATALLY SHOT SPICER

Figure 5: The example text after applying coreference resolution using NEURALCOREF

4.2.4. Formalisation of the construction

In this section we provide a formal description of the steps undertaken by the proposed extraction method and construction procedure, for the first of the proposed event extraction methods (the

remaining two are similar).

We assume that the NLP stage generated, from a given text of k sentences (written in a natural language), an *extracted* sequence:

$$\text{ExtractedText} = \text{ExtractedS}_1 \text{ ExtractedS}_2 \dots \text{ExtractedS}_k$$

such that each ExtractedS_i is a pair $(\text{Entities}_i, \text{event}_i)$, where Entities_i are the entities associated with the i -th sentence, and event_i is the root verb of i -th sentence. Moreover, let

$$\text{Entities} = \text{Entities}_1 \cup \dots \cup \text{Entities}_k = \{\text{ent}_1, \dots, \text{ent}_n\}$$

be the set of all the entities. Then, for every entity ent , let $\text{events}(\text{ent})$ be the sequence of events

$$\text{events}(\text{ent}) = x_1 \dots x_k$$

where $x_i = \text{event}_i$ if ent belongs to Entities_i , and otherwise x_i is the empty string. Intuitively, $\text{events}(\text{ent})$ is the ordered sequence of events in which entity ent ‘participated’, and such a sequence can be used to provide a time-line for this entity. Following this observation, for each entity ent with $\text{events}(\text{ent}) = \text{ev}_1 \dots \text{ev}_l$, we construct an occurrence net $ON_{\text{ent}} = (P, T, F)$, where:

$$\begin{aligned} P &= \{p_{(\text{ent}, \text{ev}_i)} \mid i = 1, \dots, k\} \cup \{p_{\text{ent}}\} \\ T &= t_{(\text{ent}, \text{ev}_1)}, \dots, t_{(\text{ent}, \text{ev}_k)} \\ F &= \{(p_{(\text{ent}, \text{ev}_i)}, t_{(\text{ent}, \text{ev}_i)}) \mid i = 1, \dots, l\} \cup \\ &\quad \{(t_{(\text{ent}, \text{ev}_i)}, p_{(\text{ent}, \text{ev}_{i+1})}) \mid i = 1, \dots, l-1\} \cup \{(t_{(\text{ent}, \text{ev}_l)}, p_{\text{ent}})\}. \end{aligned}$$

Finally, for each pair $(t, t') = (t_{(\text{ent}, \text{ev})}, t_{(\text{ent}', \text{ev}')})$ of transitions in T , where ent is different from ent' , we add channel places $q = q_{(t, t')}$ and $q' = q_{(t', t)}$ together with the arcs

$$(t, q) \quad (q, t') \quad (t', q') \quad (q', t)$$

to enforce synchronisation between t and t' . One can then show that the result is a CSO-net which can be used for analysis and visualisation.

5. Discussion

In order to evaluate our modelling approach, we used human modelling and compared it with the proposed extractor. We have conducted manual extraction and modelling experiments with expert SON users (researchers). They all produced similar outcomes in terms of model explaining the case, but (not surprisingly) in various forms. These models are not fundamentally dissimilar in terms of meaning, but rather in terms of the amount of information displayed. Figure 2 illustrates the human expert modelling for the example sentences in Figure 1.

All of SON models shown here convey the same narrative because all of the modellers reported or modelled the semantics (meaning) of the sentences in the example sentences. However, different modellers incorporated varying amounts of information (EVENTS and ONS) in their models based on their judgments. This, however, may indicate a lack of modelling consistency due to the volume of data presented in the experiment. Another issue is the amount of time

required for such modelling. To construct a model, spending time on reading, comprehending to extract the crime events, and then modelling is inevitable.

To address these issues, we employed three extraction methods and compared them to the human extraction described in Section 4.1. At first, we considered only the main verb that SPACY's parser indicates as the main verb of the sentence tagged with ROOT. ROOT tags appear once in every sentence representing the main word carrying the meaning of the sentence which is usually a verb. Then we considered extracting ROOT verbs alongside a list of common verbs used in criminal reporting. This list was compiled after analysing approximately 570 crime stories from *The Violence Policy Center*¹ website for the most frequently occurring verbs. Finally, we considered extracting ROOT verbs as well as all other verbs in the text.

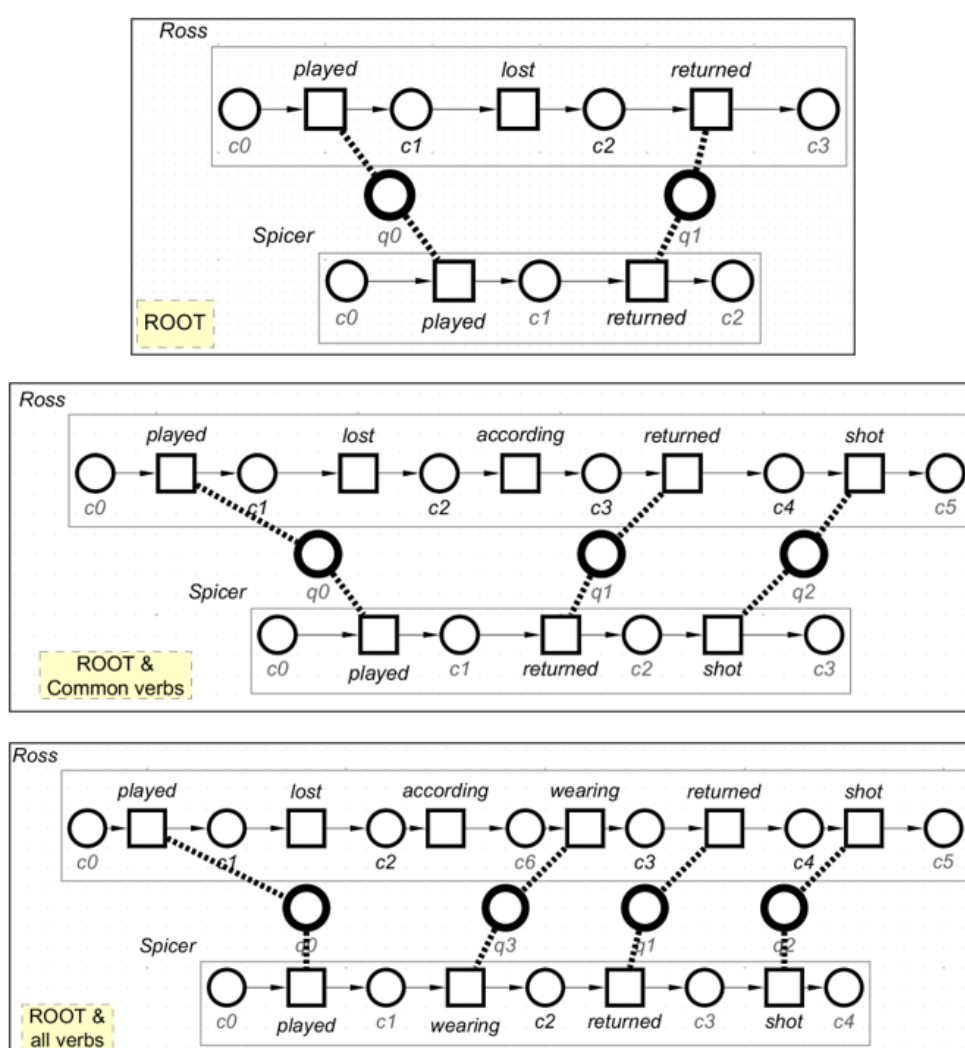


Figure 6: Extractor automated extraction, manually modelled for visualisation purposes

¹<https://vpc.org/>

TEXT2SON showed a significantly faster extraction process compared to human extraction. In our observations, we estimated that the time spent for manual extraction and modelling by the three expert users was on average 8 minutes. However, the automatic extractor handled the extraction process in about 6 seconds. But this was only the extraction phase time as we are working on the development of an automatic modeller that will function in conjunction with the extractor.

The extractor demonstrated lower accuracy by associating verbs with unrelated entities. The assumption is that if the verb is a ROOT verb and appears in a sentence with other entities, we can reasonably presume that they are linked. However, because each phrase can have only one ROOT verb, which may or may not contain the intended crime verb, a modeller may infrequently choose another verb for extraction. The automatic modeller does not recognize linking of such verbs that relate to a single distinct entity unless that entity is the only entity contained within a sentence.

However, when we fed the extractor a list of common crime verbs, it performed notably better in terms of extracting events. Yet, it maintains a connection between the newly extracted verbs and all other entities in the sentence. This is not necessarily accurate, as previously discussed, because a NON-ROOT verb can refer to a single entity. This leads to the establishment of an inaccurate communication link between the two entities.

Another challenge is the amount of information displayed in visualised models in comparison to human modelling. Human modellers frequently augment the information offered in events with additional words. Comparing Figures 2 and 6, we noticed that people tend to add more words to events, such as PLAYED DICE, LOST GAME, LEAVE UPSET, and GOES TO SPICER'S events. This addition provides further description for the model, which aids visualisation. On the other hand, the automatic extractor extracts only one token for each identified event. We are currently investigating and enhancing the extractor by including information that a human modeller would incorporate.

To facilitate comparison, the following table lists all the verbs extracted or selected for modelling by the various modellers. We can see that nearly all extraction methods agree on the verbs PLAYED, LOST, and SHOT, which may express the essential shooting events. Extracting only the ROOT produced satisfactory results except for the omission of the shooting incident, which we think to be noteworthy. As previously stated, some sentences may contain verbs other than the ROOT verb. This is one of the shortcomings of the proposed approach, which prompted us to experiment with two additional methods: extracting common verbs, and extracting all verbs. In comparison with the other two approaches, we see that extracting common criminal verbs alongside the root verb (second approach) produced more steady and acceptable result.

Modeller1	Modeller2	Modeller3	ROOT	ROOT and common verbs	ROOT and all verbs
played	play	play	played	played	played
lost	lost	leaves	lost	lost	lost
shoot(shot)	return	wearing	returned	according	according
-	shoot(kill)	goes	-	returned	wearing
-	-	shoots	-	shot	returned
-	-	-	-	-	shot

This provides an opportunity for further enhancement and development of the tool. We are now working on improving the model by incorporating word relationships such as subject and object via the use of the dependency parser. Additionally, we are creating and training the NER model, as well as introducing new labels for criminal extractions for SON modelling.

6. Conclusion and future work

We discussed the initial steps toward automatic SON data extraction using NLP prediction techniques. We used SPACY and included several of its models without modification. Specifically, the TOKENIZER, POS, PARSER, and NER. Additionally, we used NEURALCOREF to resolve mentions.

We developed our algorithm to extract people's names and events associated with them. Then we illustrated how the algorithm works by feeding TEXT2SON a text passage to extract events automatically in three different approaches. We then used expert SON users to extract and model the entities and events to verify and validate the result obtained from the tool. We compared human extraction to the final output produced by our automatic modeller.

The ongoing work focuses on improving automatic extraction and developing an automatic modeller, as well as integrating both with SONS. Among the ongoing and future works are the following:

- Developing and integrating an automatic modeller and examining it using a larger data set.
- Utilising the dependency parser in SPACY to extract events associated with the extracted entities. In our approach, we leveraged the sentence's main verb, ROOT, to express events regardless of their relationship to other entities in the sentence.
- Investigating the effect of various human extraction behaviours on SON modelling. We will look for commonalities in human extraction behaviour in order to assess a broader understanding of human extraction for the purpose of SON modelling.
- Developing a new NER model by training the model on a new set of data and introducing new distinct NER labels suited for crime extraction.

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] P. M. Nadkarni, L. Ohno-Machado, W. W. Chapman, Natural language processing: an introduction, *J. Am. Medical Informatics Assoc.* 18 (2011) 544–551.
- [6] spaCy, <https://spacy.io>, 2022.
- [7] B. Li, Visualisation and Analysis of Complex Behaviours using Structured Occurrence Nets, Ph.D. thesis, School of Computing, Newcastle University, 2017.
- [8] 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.
- [9] 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.
- [10] 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.
- [11] G. Carnaz, V. B. Nogueira, M. Antunes, Knowledge representation of crime-related events: a preliminary approach, in: R. Rodrigues, J. Janousek, L. Ferreira, L. Coheur, F. Batista, H. G. Oliveira (Eds.), 8th Symposium on Languages, Applications and Technologies, SLATE 2019, June 27-28, 2019, Coimbra, Portugal, volume 74 of *OASICS*, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, pp. 13:1–13:8.
- [12] NeuralCoref, Neuralcoref 4.0: Fast coreference resolution in spacy with neural networks, <https://github.com/huggingface/neuralcoref>, 2022.