Algorithmic tools for data-oriented law enforcement

Cocx, T.K.

Chapter 2

Adapting and Visualizing Association Rule Mining Systems for Law Enforcement Purposes

Apart from a list of crimes, criminal records contain diverse demographic characteristics of offenders. This information can be a valuable resource in the constant struggle of assigning the limited police work force to the large number of tasks in law enforcement. For this purpose we try to find relations between crimes and even more important between crimes and demographic data. These relations might give insight into the deployment of officers to certain demographic areas or reveal the linkage of certain crime categories that enable legislative bodies to change policy. The nature of the criminal record database makes it hard to use a standard association detection algorithm, because it encompasses several obviously semantically strongly linked attributes, that pollute the process. We therefore propose a number of techniques, like an attribute ban or a semantic split to improve mining results for this dataset. We also provide a means to include demographically infrequent attributes, like "female", into the comparison. We conclude by showing a way of presenting the resulting trie of frequent patterns to the user, i.e.: the law enforcer.

2.1 Introduction

The notion of relations and their discovery has always been one of the core businesses of law enforcement agencies. In particular, the relations between a crime and individuals, but also the relations between evidence and individuals, e.g., a fingerprint and its owner, or between different offenders, e.g., a mob chief and his hitman, are major focus points of daily police operations. These relations are best characterized as being relations within the tactical field, for they are drawn from and applied in the tactical area of policing. These tactical relations are most often revealed by extensive forensic research or the examination of criminal records.

These records provide, however, also the possibility to reveal existing relations on a strategical level. These relations could be used to describe, and more importantly, prevent crime. This class of relations, found in these records, encompasses relations between crime types, and relations between demographic data and crimes. Revealing these relations enables strategically oriented agencies to develop strategies for the deployment of personnel and other resources.

In this chapter we demonstrate a promising framework for revealing strategic relations for criminal records. In Section 2.2 we explain some of the underlying principles and describe the nature of the criminal record database, to which we specifically suited our efforts. This approach is the main contribution of this chapter and can be found in Section 2.3 and Section 2.4.

2.2 Background

Mining frequent patterns is an area of data mining that aims to discover substructures that occur often in (semi-)structured data. The primary subject of investigation is the most simple structure: itemsets. Much effort from the scientific community has gone into the area of frequent itemset mining, that concerns itself with the discovery of itemsets that are the most frequent in the elements of a specific database. The notion of support (the number of times an itemset is contained in an element of the database) for a single itemset was first introduced by Agrawal et al. [1] in 1993. Since then, many algorithms were proposed, most notably being FP-growth, developed by Han et al. [16], and ECLAT, by Zaki et al. [37].

It might prove rewarding to apply these methods to police data to unravel underlying principles in criminal behavior. For this purpose, the database described in Appendix B seems to be best suited. Its content has been used in a number of descriptive projects [7, 8, 26], that aimed at the exploration of criminal careers (cf. Chapter 4 and Chapter 5) or the impact of its digital nature on privacy legislation.

The nature of the database or, on a larger level, the nature of crime in general, is responsible for a large number of over- or under-present attribute values. The number of males in the database is approximately 80% and almost 90% of the offenders were, not surprisingly, born in the Netherlands. In contrast, the addiction indication is only present for 4% of the entire list. In addition to this discrepancy in attribute values, there is also an inherent correlation between certain attributes that can pollute the outcome of a search. These include (semi-)aggregated attributes, e.g., a very strong relation between age of first and last crime for one-time offenders, and relations that are to be expected logically, like for example the fact that Surinam-born people are often of Surinam-descend.

In essence, the above mentioned algorithms are very well suited to the task of discovering frequent itemsets or relations from a criminal record database; they extract frequent attributes and combine them with other frequent attributes to create frequent itemsets of criminal characteristics. These sets reveal relations between crime types, e.g., a murderer is also likely to steal, and between demographic data and crime types, e.g., a crime outside one's own town is most likely theft. The mentioned methods are however not very well suited for dealing with database characteristics like over- or under-presence, which

warrants a refit of these algorithm to facilitate a better extraction of these relations. We propose a number of solutions for this task, fitted into one single approach.

2.3 Approach

For the discovery and exploration of the above mentioned relations we propose a method with five steps. First of all, the standard algorithms for searching frequent itemsets usually rely on databases with boolean attributes, hence we need to transform our database to such a format, discussed in Section 2.3.1. As a first step in our extraction algorithm we then offer the user the possibility of excluding some (range of) attributes, called an *attribute ban* (see Section 2.3.2). The third step enables the analyst to define a *semantic split* within the database, describing the virtual line between two ranges of semantically different attributes (Section 2.3.3).

The actual search for frequent itemsets or relations takes place in the fourth step, described in Section 2.3.4. In this phase of the entire process, a number of methods are used to calculate the importance or significance of a certain itemset. The results of these algorithms are then combined into a single result and the decision is made on the incorporation of this relation into the list of relevant end-results. Finally we propose a way of presenting the discovered relations to the police analyst, in a convenient way for further processing.

The entire approach is shown in Figure 2.1, where the top-down placement of the different substeps describes their chronological order.

2.3.1 Database Refit

In this chapter we use the National HKS database of the KLPD. This database, that can be viewed as a collection of criminal records, is also queried in other chapters and we will therefore refer to Appendix B for a detailed description about its contents, compilation and usage. The methods we employ to detect relations are built around databases with boolean attributes, meaning that a single attribute is either present or not present for a certain record (person). The criminal record database we use (described in Appendix B), naturally, contains no such attributes but instead has both numerical (number of crimes, age of suspect) and nominal data (a criminal is either a one-time offender, intermediate or a “revolving door” criminal).

Numerical attributes are discretized into logical intervals, e.g., a ten year period for age. The algorithm creates a new boolean attribute for each of these intervals, tagging the correct attribute true and setting the others to false.

Nominal attributes are split and a new attribute is created for each category. The category that was selected for a certain individual is set to true.

Note that this leads to a large database, column-wise, that is very sparse; most of the columns are set to false (not present) with only one of the new attributes from the original attribute set to true. Also note that this will automatically lead to an abundance of *strong negative relations*, which are relations that appear remarkably seldom together. For example, “male” and “female” should never both be true for a single individual. Since

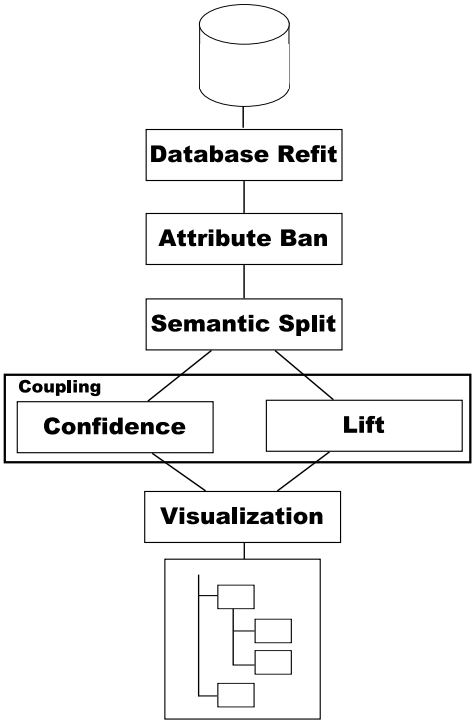

Figure 2.1: Approach for extracting relations from a criminal record database

we are not searching for these kind of relations, this does not pose a problem for the method under consideration. However, a possible extension to this approach, including such search parameters, should be able to deal with this situation (see Section 2.6).

2.3.2 Attribute Ban

In databases there are "disruptive" attributes more often than not. These attributes are on the one hand overpresent in the database while lacking significant descriptive value on the other. One could, for example, consider a plastic shopping bag from a super market. These items are sold a lot and are therefore present in a multitude of *transactions* (or rows) in a sales database. They have therefore a high chance of appearing in a frequent itemset, while their descriptive value is very low; they do not describe a customers shopping behavior, only perhaps that he or she has not brought his or her own shopping bag to the store.

There are a lot of these attributes present in the criminal record database. One of them is, for example, the *deceased* attribute. Since the database itself has only been in use for 10 years, this attribute is most often set to false, leading to an aggregated attribute (see Section 2.3.1) *alive* that is almost always present. The descriptive value of such infor-

mation to the detection process of relations between criminal behavior characteristics is quite low; the fact whether or not a person is deceased has no relevance to his criminal career or to the presence of other attributes.

To cope with the existence of such attributes in the dataset, we introduce an *attribute ban*; a set $\mathcal{B}$ of attributes or ranges of attributes that are not to be taken into account when searching for relevant relations:

$$\mathcal{B} = \{x \mid x \text{ insignificant attribute}\} \cup \{(y, z) \mid y \leq z, \forall\, q \text{ with } y \leq z, q \text{ insignificant attribute}\},$$

where the attributes are numbered $1, 2, \ldots, n$. Elements can be selected as disruptive and semantically uninteresting by a police analyst, which warrants inclusion into the set during runtime of the algorithm. This set is evaluated in a later step, when the significance of certain itemsets is calculated (see Section 2.3.4).

2.3.3 Semantic Split

A priori knowledge about the semantics of the data under consideration can be a very valuable tool in the data mining process [29]. Especially in the case of the criminal records dataset a clear semantic distinction can be made between the list of crimes on the one hand and demographic data concerning the perpetrator on the other. These two *semantic halves* are strictly separated by the numbering of attributes. In our approach, the data analyst is given the option to either use a *semantic split* by specifying the beginning attribute x of the second halve, or waive this option. From this point on, the algorithm will only combine 1 attribute of one halve (the *lower halve*) with any number of attributes from the other (the *upper halve*). The analyst can define the lower and upper halves by setting either a 1:N relation (all attributes lower than x are in the lower halve), or a N:1 relation that sets all elements greater than x as part of the lower halve. Internally, we will mark all the attributes in the lower half by inverting their sign. The semantic split x and the tagging function $\mathcal{S}$ are then defined by:

$$\mathcal{S}_x(y) = \begin{cases} -y & \textit{if } (y < x \textit{ and } 1{:}N) \textit{ or } (y \geq x \textit{ and } N{:}1) \\ y & \text{otherwise} \end{cases}$$

where y is an attribute denoted by a number.

Employing this method, the analyst can use his inside knowledge of the semantics to prohibit a multitude of relations within one semantic halve from appearing into the results. A major example of this occurs within the demographic halve of the database where people from a certain country are most often also born in that country and of that country's ethniticity. In dealing with this situation, police analysts can choose for analyzing the dataset on a 1:N basis with a semantic split between demographic and criminal data. The semantic split is evaluated during the calculation of significant relations, discussed in Section 2.3.4.

2.3.4 Detection

The actual detection of relations takes place based upon standard frequent itemset mining algorithms. The concept of *support* is the primary unit of comparison used within these techniques. The support of an itemset a ($supp(a)$) is defined as the amount of database records that contain all of the items in the itemset a. Itemsets are considered to be *frequent* when their support is above a certain *threshold*. We define the standard rating based on support for tuples of itemsets a, b as the support of the union of a and b:

$$\mathcal{R}_{\text{standard}}(a, b) = supp(a \cup b)$$

This approach suffices for standard applications, but the above mentioned concerns force our approach to resort to other comparison methods. These methods, described below, where x, y, a and b are itemsets, strive to detect itemset *interestingness* rather than frequency.

Confidence

It might be worthwhile to employ the conditional probability of a certain itemset given another itemset, thereby relinquishing the usage of support. Such a probability, called the *confidence* (of $x \rightarrow y$), is defined by:

$$C(a, b) = \frac{supp(a \cup b)}{supp(a)},$$

when $supp(a) \neq 0$.

When a certain itemset strongly implies another, the combination of itemsets may also be considered interesting. Such a combination has a high confidence for one of the two possible implications. We therefore rate the proposed new itemset on the maximum of both confidences:

$$\mathcal{R}_{\text{both}}(a, b) = max(C(a, b), C(b, a))$$

If both a certain itemset strongly implies another and the other also strongly implies the first (both confidences are high), they can easily be considered to be interesting. Usually, such a set is referred to as a *hyperclique*. If this is the case, the average of both confidences should also be relatively high. The new *candidate* for being an interesting itemset is rated in this way as follows:

$$\mathcal{R}_{\text{avg}}(a, b) = avg(C(a, b), C(b, a))$$

Lift

An itemset will certainly be interesting if its support is much higher than one would expect based upon the support of its individual member-itemsets, the subsets that comprise the itemset. The relation between expected support and actual support is the *lift* of a certain combination of itemsets. We can rate a candidate interesting itemset on this relation calculated by:

$$\mathcal{R}_{\text{lift}}(a,b) = \frac{supp(a,b)}{supp(a) \times supp(b) \ / \ rows},$$

where *rows* is the number of rows (persons) in the dataset.

Combination

For each of the four ratings mentioned above, a different threshold can (and should) be chosen. For the criminal record database, the threshold for $\mathcal{R}_{standard}$ and $\mathcal{R}_{both}$ should be relatively high due to over-presence, while the threshold for $\mathcal{R}_{avg}$ can be relatively low. Combining the four different rating results for a candidate interesting itemset can easily be done by dividing the ratings by their own respective threshold $\mathcal{T}$. The maximum of the resulting percentages will be assigned to the candidate as its score $\mathcal{P}$:

$$\mathcal{P}(a,b) = max\left(\frac{\mathcal{R}_{\text{standard}}(a,b)}{\mathcal{T}_{\text{standard}}}, \frac{\mathcal{R}_{\text{both}}(a,b)}{\mathcal{T}_{\text{both}}}, \frac{\mathcal{R}_{\text{avg}}(a,b)}{\mathcal{T}_{\text{avg}}}, \frac{\mathcal{R}_{\text{lift}}(a,b)}{\mathcal{T}_{\text{lift}}}\right)$$

If this score is higher than 1, one of the thresholds is reached and the candidate itemset is eligible for the so-called notability status.

The search for interesting itemsets (relations) starts with the itemsets of size 1. These itemsets can only be subject tot analysis by $\mathcal{R}_{standard}$, because the other rating systems require at least two itemsets to be compared. For those algorithms, all one-sized itemsets are assumed to be interesting. In the next phase of the algorithm, all itemsets that are considered interesting will be combined with each other to form candidate itemsets. When this step ends we combine the newly found interesting itemsets with all others. This process continues until there are no more new interesting combinations to be found.

Note that the semantic split and attribute ban are also taken into account when single attributes are selected to form a new candidate itemset resulting in Algorithm 1. The product of the elements of an itemset x will be denoted by $\mathcal{I}$:

$$\mathcal{I}(x) = \prod_{i \in x} i$$

This algorithm employs and yields a trie, an ordered tree data structure that is used to store an associative array, and that facilitates efficient storage and easy retrieval of interesting relations.

It may be the the case that interesting itemsets of size larger than 2 exist, where none of its children is considered to be interesting. These itemsets will not be detected by the current version of our approach because of the iterative way the itemsets are constructed and not all of the calculations adhere to the Apriori property, that states that an itemset can only be frequent if all its sub sets are also frequent. Although these itemsets are believed to be very uncommon, the results of our approach should be viewed as a good first approximation of the complete result set.

var include := true
var include2
do
  foreach interesting set x with $size(x) = 1$
    if $x \in \mathcal{B}$ **then** include := false
    foreach interesting set y with $size(y) > 1$
      if $x < 0$ **and** $\mathcal{I}(y) < 0$ **then** include2 := false **else** include2 := include
      if include2 = true **and** $\mathcal{P}(x, y) > 1$ **then** (x, y) is interesting
    endfor
  endfor
until no more new interesting itemsets

Algorithm 1: The approach for finding interesting relations

2.4 Trie Visualization

It is obviously important to produce the end results in such a way to the police analyst that he or she can immediately understand and interpret them. For this purpose we need to find a scalable (the trie is large) and fitting metaphor to describe our trie, that a non-computer scientist can easily relate to [15].

One of the few tree-related metaphors common computer users are familiar with is that of the directory or folder structure on a harddrive and more specifically the Microsoft Windows folder browse control, displayed in Figure 2.2. We propose to use this metaphor to browse the trie.

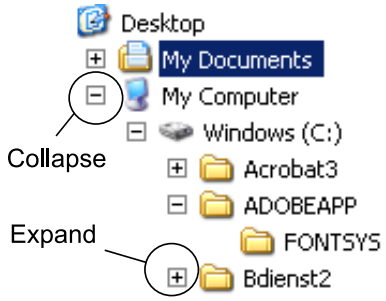

Figure 2.2: A standard Windows folder browse control

In this control, the directory structure is built up from its root, which is at the start the only node made visible to the user. If a certain node has any children, a plus sign (the *expand button*) is put next to it, which, when clicked upon, "reveals" the children of this node by showing them, indented, underneath their parent directory. After such an

operation, the expand button changes into a minus sign, which, when clicked, *collapses* the node and hides the entire subtree under its assigned node. The familiarity of this tree representation helps the police analyst in easy exploration of the data through a simple clicking interface, where each node is (part of) a discovered relation and is accompanied by its threshold reaching score.

The most important feature of this visualization is however the scalability of its content. Tries resulting from a frequent pattern search can be quite large, which makes it hard to easily browse the outcome of one's search, especially when itemsets contain more than two items. Hiding subtrees behind expand buttons enables the police analyst to limit the examination of relations to the ones that appear to be interesting, just by glancing at the first element of the itemset.

For this convenient selection of possibly interesting relations, the efficient trie needs to be transformed to its full lattice on the screen, thus making sure that each attribute that is part of *any* relation will be displayed in the root list of our control. If any of these attributes arouses the interest of the analyst, examination of this relation can start by a single click on its expand button. An example of how our method produces the end results on screen can be seen in Figure 2.3.

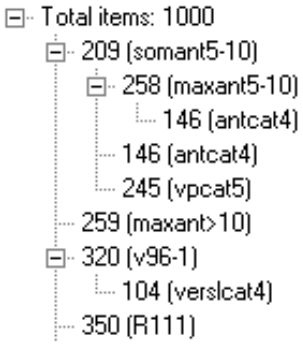

Figure 2.3: Results of a certain investigation

2.5 Experimental Results

We tested our algorithm and visualization tool on the database in Appendix B, containing approximately one million offenders and the crimes they committed. Our tool analyzed the entire database of criminals and presented the user with the resulting relations between criminal characteristics.

Some of the most notable relations have been made available to police experts in the field of strategic analysis and can contribute to policy updates in the fight against crime. Most of them were reached within a setting of either searching between crimes alone (banning all attributes in the demographic halve) or when employing a semantic split with a 1:*N* relation between demographic data and the list of crimes. The other

settings used in the experiments resulted into a list of relations that contains much jitter. Because a number of customizable settings is available, it is to be expected that the future will reveal a number of other option sets that give good results, especially after the tool has been incorporated into everyday use by the experts. Below we show some of the most remarkable and recognizable results from our experiments. The first set describes relations between police data, the second set contains demographic data as well.

Joyriding ↔ Violation of Work Circumstances ↔ Alcohol Addiction
Drug Smuggling ↔ Drug Addiction
Manslaughter ↔ Discrimination

Male ↔ Theft with Violence ↔ Possession (of weapon)
Female ↔ Drug Abuse
African Descend↔ Public Safety
Rural Areas ↔ Traffic Felonies

The confidential nature of the data used for this analysis prevents us from disclosing more detailed experimental results reached in our research.

2.6 Conclusion and Future Directions

In this chapter we demonstrated the applicability of frequent itemset mining in the analysis of criminal characteristics for strategic purposes. The tool we described compiled a list of noteworthy relations between crime types and most important demographic characteristics. The nature of a criminal record database established the need for specifically suited adaptations of standard mining algorithms to cope with over- and under-presence of and inherit relations between attributes. The end report consists of a visual, scalable and clickable version of the resulting trie and is ready to be used by police experts.

The semantic split proposed in this chapter already exploits the semantic knowledge of the analyst using the system. This can be extended to a more detailed level, a *semantic bond*, where semantic overlaps between two or more attributes can be defined. Characteristics in such a set should then not be combined in the detection phase. This way the coarse semantic split can be updated to a finer level of semantic coherence.

For this research, the search for relations was focused on positive relations, meaning, that two or more attributes appear notably often together. It may also be of interest to the law enforcer to search for attributes that appear reasonably seldom together. However, the search for those relations with our method is hindered by the boolean nature of the database, required by standard approaches, and the way we aggregate those from the original nominal or numerical attributes: aggregated attributes never appear together by definition. One way to solve this might be to join them into a semantic bond as mentioned above. Other possibilities might also be applicable.

Future research will aim at improving on the concerns mentioned above. After these issues have been properly addressed, research will mainly focus on the automatic comparison between the results provided by our tool and the results social studies reached

on the same subject, in the hope that "the best of both worlds" will reach even better analyzing possibilities for the police experts in the field. Incorporation of this tool in a data mining framework for automatic police analysis of their data sources is also a future topic of interest.