

Big Data Investigation: Chicago Crime Dataset with Hadoop MapReduce

Emilio Singh Francois van Greunen Henry Wandera Flip Nothling Tafara Freddie Hove
University of Pretoria University of Pretoria University of Pretoria University of Pretoria University of Pretoria
u14006512 u18349863 u17253129 u71230191 u18278150
u14006512@tuks.co.za fgreunen@gmail.com u17253129@tuks.co.za u71230191@tuks.co.za u18278150@tuks.co.za

Abstract—Big Data presents a number of challenges to traditional information extraction techniques due to its characteristics such as volume and velocity. Therefore, in order to more efficiently analyse Big Data, new techniques need to be applied. This report examines a Big Dataset, Chicago Crime statistics (2001-present), in terms of how it can be analysed with Hadoop MapReduce to perform efficient data analysis. This analysis will present distributed data mining algorithms alongside conventional, more shallow, data processing strategies to demonstrate how Hadoop and MapReduce can dramatically improve dataset analysis.

I. INTRODUCTION

MapReduce is a term used to describe an approach to data processing, typically in the context of Big Data, that relies on a distributed or parallel processing model [1]. As a programming model, MapReduce allows for a large dataset to be processed with the relatively simple concept of (key,value) pairings being used to parse records in a dataset. Apache Hadoop, in particular, is one of the most common implementations of the MapReduce model [2]. It is considered in the context of this report.

The Chicago Crime Incidents dataset is such a big dataset [3]. It is examined with MapReduce-based techniques as an examination of how useful MapReduce-based techniques can be in a Big Data environment.

This report examined the Chicago Crime Incident dataset using MapReduce-based processing techniques and presents a general discussion of the processes, approaches, and results of such an operation. Section II will look at the basic preliminaries of the dataset and provide a broad overview of the data as well as investigation questions. Section III will discuss the analysis methodologies, including the algorithms used. In Section IV, a general overview of the results is presented. Finally, the report is concluded in Section V.

II. DATASET PRELIMINARIES

1-2 2 notes: In this section, a broad overview of the dataset is presented. The objective is to provide the reader with an explanation of the dataset, highlight some of the interesting characteristics of the dataset, a general review of what the dataset contains, and which attributes from the dataset might be interesting to the rest of the processes described in this report.

A. Dataset Overview

The Chicago Crime Incident dataset is maintained and collected on behalf of the Chicago Police Department as part of their Citizen Law Enforcement Analysis and Reporting system (CLEAR) [4]. As an incident dataset, it records the details of incidents of crime that take place in the city with a primary aim of documenting the most pertinent details of the incident, i.e. it does not provide a full

documentation of a crime. Importantly, the dataset has been cleaned and anonymised to protect the identities of the victims.

It does, however, provide a sanitised account of the crime in terms of broadly where it happened, when, and a broad classification of the criminal activity of the incident. Geolocation information is also provided, but this has been subject to a transformation to protect the original location of the incident, as a means of ensuring the privacy of the victim. However, the transformation keeps the incident to the same city block so as to still maintain some usability in terms of analytics. The dataset has the potential to be updated, for example if additional details of an incident are uncovered, or an arrest is made.

The dataset is summarised in Table I.


Table I
CHARACTERISTICS OF THE DATASET

Size of Dataset	≈6,600,000 records
Number of Attributes	22
Attribute Types	Mixed: Categorical and Geolocation Data
Age	2001-2018 (Ongoing)

B. Feature Selection

Feature selection is a process to select a subset of the attributes (variables or features) in the dataset for further processing, based on their relevancy to the problem or task at hand [5]. While algorithms exist to perform automatic feature selection, they are not considered in this report. Instead, the feature selection consists of removing attributes that can already be adequately expressed in the dataset, or which do not seem to have any interesting information.

Below is a full list of the final attributes with a brief motivation for their inclusion:

-  This is the incident identifier. Each criminal incident is given a unique ID. Although adding little direct information, this unique code can be necessary for MapReduce tasks as it provides a key that uniquely identifies each of the incidents. It is also used to split the dataset into training and test sets for the naive Bayes algorithm discussed later.
- Location Description: The location description essentially describes, in general terms, where the crime took place. Without relying on specific location coordinates or an address, it gives useful information that can be used to categorise the types of places where certain crimes might occur. It also provides potential classifications for the geo-location data that could be used to describe regions of the city in terms of the crimes that occur there.

- **Arrest:** This is a boolean variable. It simply indicates whether or not an arrest was made for that particular incident. This is a simple yet informative feature that can be used to help measure police effectiveness over time, in terms of determining the arrests of the department over time, or through determining which crimes are more likely to see arrests.
- **Domestic:** This is a boolean variable. It indicates whether the crime incident is one involving domestic violence. This is an important social feature as domestic violence statistics are generally hard to come by due to under-reporting [6]. Therefore, anything that can be used to determine how serious the domestic violence situation in Chicago is, can be valuable.
- **District, Ward, Beat:** These are related, but distinct categorical variables that refer to a police district, administrative ward and police beat (attended by one patrol car) respectively. These all provide broad categorical labels to where the crime occurred in the city, each at a different level of granularity. This ranges from large districts served by a lot of police resources to a beat where a single patrol car has jurisdiction.
- **Year:** This is simply the year in which the incident occurred. It will span from 2001, the earliest records in the dataset, to the present as the dataset is live and continuously being added to as crimes occur. It provides a good level of abstraction by enabling for crimes to be divided by years. This gives a wider view of the criminal incidents that might occur in the city, over a year, rather than smaller periods like a week or month which can be subject to many unpredictable fluctuations.
- **Geolocation Data:** This feature gives the coordinates for the location in the city where each crime incident occurred. The coordinates are subject to a transformation to shift and rotate them such that they remain within the city block for privacy concerns. Nonetheless, this is useful information relating incidents to geographic areas. This takes the form of two coordinates, an x and y value per incident.

3-4

2 notes:

C. Investigative Questions

Several questions are investigated. The purposes of these questions are to help define and orient whatever data extraction processes and algorithms will follow, as well as to hypothesise about what useful information might be extractable from the dataset. This list is by no means exhaustive, as the process of investigation is iterative. Discoveries during the investigation can always prompt further questions and renewed investigation; the aim is to provide broadly the objectives of the processes to follow. Sub-questions are posed to aid with clarity:

5-6

2 notes:

- 1) Are there any relationships between the attributes that describe a criminal incident?
 - a) If there are relationships between attributes, can these be formulated into descriptive rules that can describe an incident in terms of some given characteristics and an outcome?
 - b) Could these descriptive rules be used to infer underlying information about the characteristics of certain kinds of crime?
- 2) What types of crime are most common in each of the districts?
 - a) Do these most common crimes fluctuate over the years in each district or do they remain relatively stable?
 - b) How does the occurrence of the most common crime in a district compare to that of the least common crime?

- 3) Are there any topological relationships, independent of strict geography, between where crimes occur and the types of crime that happens?
 - a) Is it possible to cluster criminal incidents to reveal areas of high crime frequency?
- 4) What will the classification performance be for a simple classifier that classifies whether or not a crime has led to an arrest (or will lead to one)?
 - a) Which attribute is the most influential in predicting an arrest based on specific attributes of a crime?
- 5) Can basic aggregations and summations of attributes in the dataset reveal useful information?
 - a) Which attributes should be extract in this way?
 - b) How does the volume of the dataset influence this sort of processing?

1) *Algorithmic Motivation:* Considering the questions above, below are motivations for algorithms that could help resolve them.

- 1) The use of an Apriori association rule mining algorithm [7, 8] is proposed to answer Question 1. Association rules can be very descriptive and mining such a large dataset is normally a processing challenge. Using MapReduce and an Apriori association rules miner could extract descriptive rules. The algorithm allows implementation in a distributed processing environment, because the process is iterative and commutative, meaning the processing can occur out of direct order, i.e. in parallel. The final results can be combined together again without concern for how this would affect the outcome.
- 2) For answering Question 2, a MapReduce algorithm can be constructed to partition the dataset into year slices, and then from there, gather and aggregate a combination of fields together that represent the information required. It is a simpler solution but can work quickly to answer a relatively important question.
- 3) In order to answer Question 3, a Self Organising Map (SOM) is proposed [9]. As a type of neural network, a SOM is applicable here because it can be used to preserve topological ordering across a training process and train in an unsupervised or supervised way. That is, we can first use the SOM as a form of discovery tool, determining if there are any inherent topological relationships and then later, apply labels to the trained SOM and extract further information about the distribution of criminal incidents, all independent of the strict geographic position of the incidents. Importantly, the SOM algorithm can be applied to the dataset using a batch approach, which will allow for it to operate on such a large dataset more easily than a online algorithm.
- 4) For Question 4, a naive Bayes classifier is created to determine how accurately such a classifier can predict an arrest.
- 5) Question 5 can be answered using MapReduce's native aggregation potential.

III. ANALYSIS METHODOLOGY

In this section, a methodology is presented for the types of algorithms considered in this report. The methodology will cover broadly how the algorithm will work as well as discuss, where relevant, how the algorithm parameters can influence the results it produces.

A. Map Reduce Summation and Aggregation

Map Reduce shines in its ability to perform simple aggregations and summations of attributes. Much of the data can be aggregated,

for example through frequency counts, to produce a summarised version of the information contained within the dataset using map and reduce functions and the Hadoop File System. The process would play to the strengths of MapReduce, being simple operations that are fundamentally stateless, and also condense the dataset into something more manageable for visualisation and analysis. All that needs to be determined are which attributes to aggregate in this way. In particular some examples of the types of aggregation would be:

- Number of crimes per type per district per year
- Total number of crimes involving domestic violence without arrests

B. Apriori Algorithm

The Apriori algorithm is a frequent item set mining algorithm for association rule learning [7]. The algorithm functions through the determination of frequent item sets in a transactional database and works iteratively, growing the size of the item sets as it does. Statistical measures like confidence and support are used as stopping conditions [8, 10].

Support in this context is defined as:

$$\text{support}(X) = \frac{|\{t \in T; X \subseteq t\}|}{|T|} \quad (1)$$

where X refers to an item set and T is the full dataset. Confidence is defined as:

$$\text{conf}(X \Rightarrow Y) = \frac{\text{support}(X \cup Y)}{\text{support}(X)} \quad (2)$$

where $X \Rightarrow Y$ is an association rule and X and Y are item sets.

A distributed Apriori algorithm was created, based on the Apriori algorithm and MapReduce, to run on a Hadoop MapReduce system. This will be termed MR_Apriori for reference. The general algorithm is a 3 phase algorithm. In each phase, a separate job is scheduled in order to compute a final set of items which can then be turned into association rules with a calculated confidence level. The interpretation of the rules is left to manual analysis, but the rules are subject to 2 minimums: minimum support and minimum confidence, two thresholds for judging whether rules are sufficiently supported with evidence. The support is set to 6600 records per rule and the confidence is set to 80%.

Additionally, the algorithm partitions the dataset along a time dimension, by dividing the overall dataset into time slices of 3 years each, for a total of 6 slices. The reason for this is introduce more exploration potential in the rules mining process. Through this technique, more subtle rules can be extracted as they do not need to be prevalent throughout the entire dataset. More rules will be produced through this process that can be pruned later, rather than too few that have the potential of not offering valuable insights.

Importantly, the choice of minimum support is most important to the output. The minimum confidence merely provides a quality measure that can be used to measure how accurate a rule is by how often it is correct in all instances that the rule is pertinent to in the dataset. However, the minimum support will have a huge impact on the algorithm performance because it determines how many rules are selected. A very low support minimum produces a larger number of rules but these rules are also more specific, since they can lack sufficient evidence of generalisation. A higher support will produce

far fewer rules but generally these rules are evident across large sections of the dataset. The values chosen are expected to produce rules of length (number of antecedents per consequent) of about 2 to 5. Larger rules are selected over smaller rules as these are more descriptive and therefore offer more insight than shorter rules.

C. SOM

The MapReduce SOM specified here uses a batch implementation. Not only does this prevent large memory requirements, it also removes the need for both a learning rate and a neighbourhood radius, removing two parameters that would need to be tuned. What remains is the size of the map and the criteria for judging convergence. This SOM is implemented to measure convergence by either reaching a maximum number of epochs, or if the difference in the weight updates between one epoch to the next becomes sufficiently small $W_{current} - W_{old} \leq \epsilon$ where ϵ is 1×10^{-9} and W_{old} $W_{current}$ refers to the average weight updates made during prior and current epoch respectively.

Another facet of the SOM is the process of neuron labelling. After the SOM has been trained, data with labels, that we can choose such as whether a crime was violent or not, can be passed into the trained map and used as labels for neurons. In this process, the trained map can preserve the topological order of labelled data and demonstrate clusters, for example, amongst the labelled nodes that represent regions in the feature space that correspond to labels we have just assigned.

Through experimentation, a map size of 5 by 5 neurons is chosen to give a sizeable map that would still be relatively compact enough given the few dimensions of the available geo-location data. Much larger maps are unlikely to show any more significant relationships while increasing the performance overheads.

D. Naive Bayes Classifier

We opine that a criminal activity should lead to an arrest. There is value in being able to predict which crime incident is likely to lead to an arrest being made. An arrest prediction, as well as the crime's attributes likely to effect an arrest, could be used to improve arrest rates.

A naive Bayes classifier is a probabilistic classifier. This classifier is called naive because it assumes independence of input features, i.e. the independent variables. Therefore, we made a simplifying and possibly consequential assumption that input features are independent of one another, i.e. an input feature can only be related to the target feature (dependent variable). This is unlikely for this data set but there is evidence that even when it is not true, the classifier is still surprisingly accurate and competitive with other modern classifiers [11].

A simple naive Bayes classifier was trained to classify an arrest into two classes - True or False. The following input features were used to classify a crime as leading to an arrest:

- Domestic - True/False
- Primary Type - A categorical description, e.g. 'battery'
- Location Description - A categorical description, e.g. 'apartment'
- Ward - An integer, e.g. 25

To train the model, 75% of the crimes were used, and 25% of the crimes were used to evaluate the model. Applying the

modulo/remainder operator on the crime's unique numeric *ID* field resulted in this split, i.e. if the *ID* modulo (%) 4 is equal to zero, the instance belongs to the test set, and to the training set otherwise. This split allowed the original data file to be processed without splitting it into separate training and test files, which would take significant disk space. Instead, both the training and test phases by the above logic to determine whether a particular line instance belongs to either the training or the test set.

The model was trained in a MapReduce process, line by line. The computed probabilities of the training set - the trained classification model - were serialized and written to a file. By simply reading the file's contents, the model could be evaluated on the test set and utilised independent of the training or MapReduce process.

IV. RESULTS

This section presents a short discussion of the results from the algorithms. Advanced visualisations of the results are not included, as they are presented in a later report. The purpose here is twofold: firstly to broadly discuss the output of the algorithms with examples where necessary and also to present findings in terms of insights or other useful information that has been extracted. A later report will include the visualisations of these findings but it is important to discuss them nonetheless.

A. Overview of Findings

1) *MapReduce Aggregation and Summation*: The results of this sort of processing is a standard MapReduce output in most senses. It would be a collection of results, a list of keys representing some derived attribute and a value, such as a count. There is no need to perform an additional validation process; the accuracy of the results is the accuracy of the information contained within the dataset. The output is not directly comparable to the output of the other processes as its purpose was merely to collect information from the dataset that is not readily apparent. The value of course in this output is in being able to visualise it more easily than the original dataset. Therefore no results from this process are presented here as they are best suited for the later report.

2) *Apriori Algorithm*: The output of the Apriori algorithm is more subjective than the simple information extraction. Firstly, the two parameters (support and confidence minimum thresholds) will heavily influence what sorts of results are produced. This further requires additional subjective input as choosing which rules that are chosen, out of many, requires some discretion to choose rules that are meaningful and sensible. There is the implication that some rules might be produced that are statistically supported, in terms of support and confidence, but offer little insight or meaning is there and has to be treated with care.

Since the rules that are produced, at the end of the algorithm, all have sufficient support and confidence to back them up, the final choice of the rules has some subjective bias. Rules that relate to certain types of crime, arrests, domestic violence events and locations of crime are more important, it is felt, as these kinds of rules can be used to describe events and their likely outcomes. For example, if a rule has the consequent of an arrest, then the antecedents describe what conditions are likely to produce an arrest. This is the kind of useful information that can be used to adjust policing strategies to produce better outcomes, such as more arrests and better crime prevention.

In order to demonstrate the success of the algorithm, a few rules are presented below along with their support and confidence levels. This is by no means all the rules created, merely a sample as demonstration of the successful mining operation:

Rule	Support	Confidence
DOMESTIC _{NO} ,SIDEWALK,BATTERY \Rightarrow No Arrest	9024	0.83
NARCOTICS,SIDEWALK,DOMESTIC _{NO} \Rightarrow Arrest	29492	0.99

3) *SOM*: The output, here, of the SOM is more about the visualisations of the neuron map rather than anything else. As the SOM was not used in a predictive capacity, the unsupervised learning done was performed to as an exploration of the dataset. There are three primary outputs of the SOM of particular value to this investigation. The first is the U-Matrix. The U-Matrix is simply a matrix visualisation of the distances between a neuron and its neighbours. This is helpful because it can help us to recognise clusters, and boundaries in the topological structure of the SOM by how far apart the neurons are to each other. Since we can label the SOM, it will also then be able to tell us where clusters exist, presumably, between the label information we add to the SOM. The second piece of information is a BMU (Best Matching Unit) matrix. Here we are again visualising the SOM except this time we are representing the number of times a neuron was chosen as the BMU. With this, we can determine the most dense locations of criminal activity as those neurons will have the most matching patterns. Finally, a labelled SOM map itself is produced that can demonstrate, visually, the relationships between the various labels assigned to the SOM after training.

4) *Naive Bayes Classifier*: Unlike with the other algorithms whose output will be data, representing extracted knowledge of some kind, the output of this algorithm is really a predictive model that tries to capture the relationship between criminal activity and arrests as a probabilistic relationship. Consequently, evaluating the output of the naive Bayes algorithm is more about evaluating the statistical success of the classifier and whether the results it produces are actually useful for the purposes of understanding how likely arrests are.

There are no real parameters to tune in the conventional sense but the size of the training/testing datasets will play a factor in how well the model actually generalises in practice. With the current split, the results show enough evidence of being robust to not justify further experimentation although the possibility of a more optimal divide between data used for training and data used for testing, through a process like k-fold cross-validation exists as well.

B. Presentation of Findings

1) *Naive Bayes Classifier*: The full naive Bayes classifier's performance is presented in Table III. This is the performance for the classifier with all 4 features used as input. The difference in performance between training and tests sets is negligible, which indicates that the model generalizes well. Accuracy is high at 84%. Precision is also very high at 0.99, which means that when the classifier labels a crime as an arrest, it is very accurate in doing so. However, recall is very low at 0.43, which means that the classifier was not very good at identifying positive cases, i.e. arrests (although it was very accurate when an arrest was identified/predicted - precision). The area under the receiver operating curve (ROC_AUC) is good, at 0.714, which indicates that the classifier is superior to a random classifier, but quite inferior to the perfect classifier (at 1.00).

Table II provides a comparison of classifier performance when individual features are used as inputs. They are compared with the full classifier, i.e. the classifier with all 4 features as input. The Domestic and Ward features are not good classifiers on their own, and are no better than a random classifier. The Location Description classifier is only very slightly better than a random classifier. The Primary Type feature is the most significant individual feature, and is comparable with the full model (All). In fact, its metrics indicate a better performance than the full model, which suggests that the Primary Type feature is the most significant predictor of arrest.

These results answers the question of how accurate a simple classifier can be. A naive Bayes classifier is quite accurate and very precise, but lacks good recall. In other words, the trained classifier is fair, but not entirely trustworthy. Additionally, the Primary Location feature is the most important feature of the 4 features considered when it comes to classification. Note, in Tables II and III Location refers to Location Description and Primary refers to Primary Type.

Figure 1. Confusion Matrix for the Full Naive Bayes Classifier

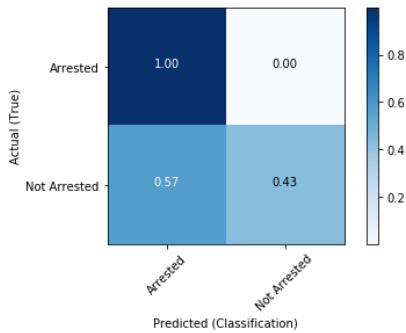


Table II
NAIVE BAYES INDIVIDUAL RESULTS

Metric	Domestic	Primary	Ward	Location	All
Accuracy	72.370%	86.366%	72.370%	72.531%	84.130%
Precision	0.000	0.933	0.000	0.735	0.993
F1-Score	0.000	0.689	0.000	0.018	0.599
Recall	0.000	0.546	0.000	0.009	0.429
ROC AUC	0.500	0.765	0.500	0.504	0.714

Table III
NAIVE BAYES CLASSIFIER RESULTS

Metric	Training Set	Test Set
Instances	4,550,904	1,517,438
Accuracy	84.18%	84.13%
Precision	0.994	0.993
F1-Score	0.599	0.599
Recall	0.429	0.429
ROC AUC	0.714	0.714

2) *Apriori Algorithm*: Presented below are a list of the observations that were uncovered through the Apriori mining process. These reflect potential relationships between certain outcomes and incident attributes. The list considers the important relationships that were discovered but would be cumbersome to visualise. The span of time in which the relationship is observed is provided as well:

- 2001-2018: Motor vehicle theft that occurs on a street is likely to not end with an arrest

- 2001-2015: Crimes involving narcotics that were reported on the a street was likely to result in an arrest
- 2004-2018: Criminal Damage(property damage) was most likely directed against locales in the street (public access)
- 2007-2012: Crime reported in Ward 012 was unlikely to result in a successful arrest.
- 2010-2018: Crime reported in Ward 008 was unlikely to result in a successful arrest if that crime occurred in the street.
- 2010-2012: Theft in Ward 018 and 019 was unlikely to result in an arrest.
- 2013-2018: Deceptive Practice crime was unlikely to result in an arrest.
- 2016-2018: Theft of items in restaurants were likely to not result in an arrest.

There are number of important extrapolations to make from these observations. Firstly, as a city with a large driving population, theft of motor vehicles being the most prominent crime fits to the environment. That most of the crimes involving motor vehicles being stolen from street, that is not parked in a garage or similar, reflects the lack of space in the crowded city which means secure parking facilities are not often available. Worryingly, the observation also shows a failure to make arrests and therefore subsequently recover lost vehicles. Another very long running observation is that narcotic crime that occurred in the street was likely to produce an arrest. This suggests that active police observation, as well as the availability of evidence of the narcotics on hand, means an easier potential arrest. That both of these crimes are so long spanning, shows that efforts to combat both are not as successful as might be believed. Wards 012, 019, 008 and 012 are identified as problematic policing areas. Crime that occurs in these areas tends to go without arrests. Theft and street-related crime go hand in hand in these wards. Unusually, in more recent years, there are other types of crime coming into more frequency. In particular, deceptive practice and theft in restaurants are both crimes that emerge as more frequent only in the latter portion of the dataset.

A central theme in the results, and of course in the mining, is to examine what crime and arrests. The effectiveness of policing is difficult to objectively measure and so a greater emphasis is placed on understanding crime and arrests as an available method to quantify how effective policing might be.

V. CONCLUSION

This report presented a number of techniques for dataset processing in a Big Data environment and as shown, these techniques were able to extract a wide variety of information about crime in the city of Chicago. While this report did not present all of the findings, it did present a number of the most pertinent ones demonstrating the potential for new insight into crime and criminal incidents with MapReduce and Hadoop.

REFERENCES

- [1] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters," *Commun. ACM*, vol. 51, no. 1, pp. 107–113, Jan. 2008.
- [2] T. White, *Hadoop: the definitive guide*. OReilly, 2015.
- [3] "Crimes — 2001 to present — city of chicago — data portal." [Online]. Available: <https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2>

- [4] T. Douglas, "Chicago police cut crime with major upgrades to analytics and field technology." [Online]. Available: <https://bit.ly/2vvNBK7>
- [5] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning with applications in R*. Springer, 2017.
- [6] B. Strong and T. F. Cohen, *The marriage and family experience: intimate relationships in a changing society*. Cengage Learning, 2017.
- [7] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in *Proceedings of the 20th International Conference on Very Large Data Bases*, ser. VLDB '94. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1994, pp. 487–499. [Online]. Available: <http://dl.acm.org/citation.cfm?id=645920.672836>
- [8] M. Hahsler, B. Grün, and K. Hornik, "Introduction to arulesmining association rules and frequent item sets," 2006.
- [9] T. Kohonen, "The self-organizing map," *Neurocomputing*, vol. 21, no. 1, pp. 1 – 6, 1998. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0925231298000307>
- [10] J. Hipp, U. Güntzer, and G. Nakhaeizadeh, "Algorithms for association rule mining — a general survey and comparison," *SIGKDD Explor. Newsl.*, vol. 2, no. 1, pp. 58–64, Jun. 2000.
- [11] H. Zhang, "The optimality of naive bayes," *AA*, vol. 1, no. 2, p. 3, 2004.

Big Data Investigation : Chicago Crime Dataset with Hadoop MapReduce

Singh, Emilio; Greunen, Francois Van; Wandera, Henry

-
- 01 Unknown Unknown Page 1
18/9/2018 21:26
The ID does not contribute any information for analysis, unless it has some time or other dimension. I suppose it should be removed from the datasets before they are used in an algorithm.
-
- 02 Unknown Unknown Page 1
18/9/2018 21:23
-
- 03 Unknown Unknown Page 2
18/9/2018 21:30
It seems to me that months are important. I saw a time series on Github (Henry? Francois?) with a strong seasonal pattern. Fewer crimes in their winter. There may be a similar pattern in weekdays.
-
- 04 Unknown Unknown Page 2
18/9/2018 21:27
-
- 05 Unknown Unknown Page 2
18/9/2018 21:34
This may be right but when the report is written, all the questions are known and can be included. Just a suggestion.
-
- 06 Unknown Unknown Page 2
18/9/2018 21:31
-
- 07 Unknown Unknown Page 3
18/9/2018 21:53
Does naive Bayes calculate norms? If so then Ward should perhaps be a factor, otherwise it will calculate distances based on the integer values.

18/9/2018 21:52

18/9/2018 21:22

This will read the input file twice and should increase the processing time. Isn't it better to split the files?

18/9/2018 21:21