**Literature Review**

**1. The CN2 induction algorithm”, Machine Learning**

**Clark, P. and Niblett, T., “The CN2 induction algorithm”, Machine Learning, Vol. 3(4), pp. 261–283, 1999.**

The subgroup discovery, domain of application of CN2-SD, is defined as: ‘‘given a population of individuals and a property of those individuals, we are interested in finding a population of subgroups as large as possible and have the most unusual statistical characteristic with respect to the property of interest’’. The subgroup discovery algorithm CN2-SD, based on a separate and conquer strategy, has to face the scaling problem which appears in the evaluation of large size data sets. To avoid this problem, in this paper we propose the use of instance selection algorithms for scaling down the data sets before the subgroup discovery task. The results show that CN2-SD can be executed on large data set sizes pre-processed, maintaining and improving the quality of the subgroups discovered.

In data mining (Han & Kamber, 2000), the generation of representative models from data is a staple process. The models, depending on their domain of application, can be predictive or descriptive. Predictive induction has as objective the construction of a model or a set of rules to be used in classification or prediction (Chang, Lai, & Lee, 2007), while descriptive models are aimed at the discovery of individual rules which define interesting patterns in data (Yen & Lee, 2006). Subgroup discovery (SD) is situated at the intersection of predictive and descriptive induction. In the subgroup discovery task, the rules or subgroups are discovered using heuristics which tries to find the best subgroups in terms of rule coverage and distributional unusualness Sub-group discovery aims at discovering individual rules of interest, which must be represented in explicit symbolic form and which must be relatively simple in order to be recognized as actionable by potential users. The CN2-SD (Lavracˇet al., 2004) is a recent proposal in SD offering promising results.

It is an adaptation of the classification rule learner CN2 algorithm based on a sepa-rate and conquers strategy (Clark & Boswell, 1989; Clark & Niblett, 1991). The main modifications are: its covering algorithm, search heuristic, probabilistic classification of instances, and evaluation measures. The issue of scalability and the effect of increasing the size of data sets are always present in data mining (Domin-go, Gavalda´, & Watanabe, 2002; Provost & Kolluri, 1999). The scaling problem, due to large size data sets, produces situations where the CN2-SD algorithm cannot be executed. The evaluation necessities to apply the heuristic are expensive computationally and this cost is directly proportional to the size of the data set.

**2. Mining Association Rules between Sets of Items in Large Databases.**

**R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large database.In Proc. 1993 ACM-SIGMOD Int. Conf. Management of Data (SIGMOD’93), pp:207-216, Washington, DC, May 1993.**

We are given a large database of customer transactions. Each transaction consists of items purchased by a customer in a visit. We present an efficient algorithm that generates all significant association rules between items in the database. The algorithm incorporates buffer management and novel estimation and pruning techniques. We also present results of applying this algorithm to sales data obtained from a large retailing company, which shows the effectiveness of the algorithm.

Consider a supermarket with a large collection of items. Typical business decisions that the management of the supermarket has to make include what to put on sale, how to design coupons, how to place merchandise on shelves in order to maximize the profit, etc. Analysis of past transaction data is a commonly used approach in order to improve the quality of such decisions. Until recently, however, only global data about the cumulative sales during some time period (a day, week, a month, etc.) was available on the computer. Progress in bar-code technology has made it possible to store the so called basket data that stores items purchased on a per-transaction basis. Basket data type transactions do not necessarily consist of items bought together at the same point of time. It may consist of items bought by a customer over a period of time. Examples include monthly purchases by members of a book club or a music club.

The work reported in this paper could be viewed as a step towards enhancing databases with functionalities to process queries such as (we have omitted the confidence factor specification):

Find all rules that have \Diet Coke" as consequent. These rules may help plan what the store should do to boost the sale of Diet Coke. Find all rules that have \bagels" in the antecedent. These rules may help determine what products may be impacted if the store discontinues selling bagels.

Find all rules that have \sausage" in the antecedent and \mustard" in the consequent. This query can be phrased alternatively as a request for the additional items that have to be sold together with sausage in order to make it highly likely that mustard will also be sold.

Find all the rules relating items located on shelves A and B in the store. These rules may help shelf planning by determining if the sale of items on shelf A is related to the sale of items on shelf B.

**3. Deliver smarter products and services by unifying software development and IT operations.**

**IBM, “Deliver smarter products and services by unifying software development and IT operations,” Issue Date : Sep. 2009, ISBN RAW14175-USEN-00.**

At first consideration the answers to these questions are simple: Yes, the organizational separation of development and IT operations serves an economic purpose. Development and IT operations serve very distinct functions. Not only do they undertake different work streams with different tools, processes, and cultures, they are also two very distinct activities from an economic perspective. Software development focuses on creating new value and IT operations focuses on assuring the health of that value stream after it has been created. Given this distinction, specialization within development and operations makes sense. As with many undertakings, specialization underpins organizational effectiveness, creating value by allowing teams to focus on a subset of well defined objectives. Specialization, however, only works up to the point where work flows are contained within organizational boundaries. When focus is put on the on the larger goal of end-to-end service delivery where workflows must be coupled across teams, this specialization of tools, skills, and processes can expose the business to unwarranted risk. Such composite applications are truly powerful: they enable businesses to transform existing processes without having to rewrite large numbers of existing applications. That said, they tend to couple the development and IT operation processes. Because composite applications link many existing applications together in a complex way, many problems may not emerge until the application runs in a live production environment. This is not a problem with testing — good tools are available to ensure new applications work correctly— but rather it’s a question of how the new application impacts the existing environment. We increase costs because we overlap the processes in development and operations. Take the deployment of applications, for example. From an operations perspective, deployment into production is typically a well understood, controlled activity with some level of tooling, process management, and quality assurance in place. Applications are, however, often deployed—in fact, much more frequently deployed—into test environments. This process, for most testing organizations, is much less controlled, process-oriented, or automated than its operational counterpart.

**4. Learning Decision Trees Using the Area under the ROC Curve**

**Cesar Ferri-Ram´ ırez, Peter A. Flach, and Jose Hernandez-Orallo. Learning decision trees using the area under the roc curve. In Proceedings of the Nineteenth International Conference on Machine Learning, pages 139–146, Morgan Kaufmann, 2002.**

ROC analysis is increasingly being recognised as an important tool for evaluation and comparison of classifiers when the operating characteristics (i.e. class distribution and cost parameters) are not known at training time. Usually, each classifier is characterised by its estimated true and false positive rates and is represented by a single point in the ROC diagram. In this paper, we show how a single decision tree can represent a set of classifiers by choosing different labelling of its leaves, or equivalently, an ordering on the leaves. In this setting, rather than estimating the accuracy of a single tree, it makes more sense to use the area under the ROC curve (AUC) as a quality metric. We also propose a novel splitting criterion which chooses the split with the highest local AUC. To the best of our knowledge, this is the first probabilistic splitting criterion that is not based on weighted average impurity. We present experiments suggesting that the AUC splitting criterion leads to trees with equal or better AUC value, without sacrificing accuracy if a single labelling is chosen. Traditionally, classification accuracy (or error), i.e., the percentage of instances that are correctly classified (respectively incorrectly classified) has been used as a measure of the quality of classifiers. However, in many situations, not every misclassification has the same consequences, and problem-dependent misclassification costs have to be taken into account. If the cost parameters are not known at training time, Receiver Operating Characteristic (ROC) analysis can be applied (Provost & Fawcett 1997; Swets, Dawes & Monahan 2000). ROC analysis provides tools to distinguish classifiers that are optimal under some class and cost distributions from classifiers that are always sub-optimal, and to select the optimal classifier once the cost parameters are known. ROC analysis for two classes is based on plotting the true-positive rate (TPR) on the y-axis and the false-positive rate (FPR) on the x-axis. This gives a point for each classifier. A curve is obtained because, given two classifiers, we can obtain as many derived classifiers as we want along the segment that connects them, just by voting them with different weights. Consequently, any point “below” that segment will have greater cost for any class distribution and cost matrix, because it has lower TPR and/or higher FPR. According to that property, given several classifiers, one can discard the classifiers that fall under the convex hull formed by the points representing the classifiers and the points (0,0) and (1,1), which represent the default classifiers always predicting negative and positive, respectively.

**5. Logical Design of Data Warehouses from XML**

**M. Banek, Z. Skocir, and B. Vrdoljak.Logical Design of Data Warehouses from XML . In ConTEL ’05: Proceedings of the 8th international conference on Telecommunications, volume 1, pages 289–295, 2005.**

Data warehouse is a database that collects and integrates data from heterogeneous sources in order to support a decision making process. Data exchanged over the Internet and intranets has recently become an important data source, having XML as a standard format for exchange. The possibility of integrating available XML data into data warehouses plays an important role in providing enterprise managers with up-to-date and relevant information about their business domain. We have developed a methodology for data warehouse design from the source XML Schemas and conforming XML documents. As XML data is semi-structured, data warehouse design from XML brings many particular challenges. In this paper the final steps of deriving a conceptual multidimensional scheme are described, followed by the logical design, where a set of tables is created according to the derived conceptual scheme. A prototype tool has been developed to test and verify the proposed methodology. Data warehousing system is a set of technologies and tools that enable decision-makers (managers and analysts) to acquire, integrate and flexibly analyze information coming from different sources. The central part of the system is a large database specialized for complex analysis of historical data, called a data warehouse. The process of building a data warehousing system includes analysis of the data sources, design of a warehouse model that can successfully integrate them and later the construction of the warehouse according to the proposed model. Decision-makers use OLAP (OnLine Analytical Processing) tools to put queries against the warehouse in a quick, intuitive and interactive way. OLAP tools use the multidimensional data model, which enables focusing on small pieces of data, generally a few numerical parameters, that are most interesting for the decision making process. Other data in the warehouse are organized hierarchically into several independent groups, called dimensions, and used to perform calculations with the few important parameters. Data warehouses, owned by big enterprises and organizations, integrate data from heterogeneous sources: relational databases or other legacy database models, semi-structured data and different file formats. Recently, the World Wide Web, Web services and different information systems for exchanging data over the Internet and private networks have become an important data source.

**6. A multisession-based multidimensional model**

**M. Body, M. Miquel, Y. B´edard, and A. Tchounikine. A multidimensional and multi version structure for OLAP applications. In DOLAP ’02: Proceedings of the 5th ACM international workshop on Data Warehousing and OLAP, pages 1–6, New York, NY, USA,2002. ACM.**

This paper addresses the problem of how to specify changes in multidimensional databases. These changes may be motivated by evolutions of user requirements as well as changes of operational sources. The multi version-based multidimensional model we provide supports both data and structure changes. The approach consists in storing star versions according to relevant structure changes whereas data changes are recorded through dimension instances and fact instances in a star version. The model is able to integrate mapping functions to populate multi version-based multidimensional databases.

On-Line Analytical Processing (OLAP) has emerged to support multidimensional data analysis by providing manipulations through aggregations of data drawn from various transactional databases. This approach is often based on a Multidimensional Data Base (MDB). A MDB schema [1] is composed of a fact (subject of analysis) and dimensions (axes of analysis). A fact contains indicators or measures. A measure is the data item of interest. As mentioned in [2], fact data reflect the dynamic aspect whereas dimension data represent more static information. However, sources (transactional databases) may evolve and these changes have an impact on structures and contents of the MDB built on them. In the same way, user requirement evolutions may induce schema changes; eg. to create a new dimension or a new “dimension member” [3], to add a new measure,…Changes occur on dimensions as well as facts. This paper addresses the problem of how to specify changes in a MDB. The changes may be related to contents as well as schema structures. Our work is not limited to represent the mapping data into the most recent version of the schema. We intend to keep trace of changes of multidimensional structures.

**7. Transaction Management for a Main-Memory Database**

**P. Burte, B. Aleman-meza, D. B. Weatherly, R. Wu, S. Professor, and J. A. Miller. Transaction Management for a Main-Memory Database. The 38th Annual South eastern ACM Conference, Athens, Georgia, pages 263–268, January 2001.**

As part of research by members of the Department of Computer Science at the University of Georgia, we have developed a Java-based Transaction Manager that fits into the multi-layered design of MMODB, a main-memory database system. We have sought to maximize the benefits of the Java programming language and to implement transaction principles that are suitable for in-memory databases. In this paper, we examine the details of thread concurrency and resource locking protocols, our deadlock prevention scheme, and the Java-based implementation of these design decisions. We show the effectiveness of our design with performance tests that simulate typical transactions on a highly concurrent database system.

**8. Discovering business intelligence from online product reviews: A rule-induction framework.**

**W. Chung and H. Chen. Web-Based Business Intelligence Systems: A Review and Case Studies. In G. Adomavicius and A. Gupta, editors, Business Computing, volume 3, chapter 14, pages 373–396. Emerald Group Publishing, 2009.**

Online product reviews are a major source of business intelligence (BI) that helps managers and marketers understand customers’ concerns and interests. The large volume of review data makes it difficult to manually analyze customers’ concerns. Automated tools have emerged to facilitate this analysis, however most lack the capability of extracting the relationships between the reviews’ rich expressions and the customer ratings. Managers and marketers often resort to manually read through voluminous reviews to find the relationships. To address these challenges, we propose the development of a new class of BI systems based on rough set theory, inductive rule learning, and information retrieval methods. We developed a new framework for designing BI systems that extract the relationship between the customer ratings and their reviews. Using reviews of different products from Amazon.com, we conducted both qualitative and quantitative experiments to evaluate the performance of a BI system developed based on the framework. The results indicate that the system achieved high accuracy and coverage related to rule quality, and produced interesting and informative rules with high support and confidence values. The findings have important implications for market sentiment analysis and e-commerce reputation management.

As e-commerce supports higher interactivity among users with Web 2.0 applications, user-generated content posted on these sites is growing significantly. Users not only consume Web content, but also produce massive data of their participation, often affecting other users’ decisions. A study finds that more than three-quarters of the 2078 users reported that online product reviews had a significant influence on their purchase decisions (com Score, 2007). These online product reviews contain descriptions about user preferences, comments, and recommendations that serve as a major source of business intelligence (BI), helping managers and marketers to better understand customers. Management scholar Peter Drucker emphasizes that ‘‘what is value to the customer’’ may be the most important question to answer in order to realize a business’s mission and purpose (Drucker, 2003). However, the large volume of online product review data creates significant information overload problems (Bowman, Danzig, Manber, & Schwartz, 1994), making it difficult to discover BI from the reviews and to analyze customer concerns.

**9. Crowd sourcing Predictors of Behavioural Outcomes.**

**Josh C. Bongard, Member, IEEE, Paul D. H. Hines, Member, IEEE, Dylan Conger, Peter Hurd, and Zhenyu Lu. IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING YEAR 2013.**

Generating models from large data sets—and deter-mining which subsets of data to mine—is becoming increasingly automated. However choosing what data to collect in the first place requires human intuition or experience, usually supplied by a domain expert. This paper describes a new approach to machine science which demonstrates for the first time that non-domain experts can collectively formulate features, and provide values for those features such that they are predictive of some behavioural outcome of interest. This was accomplished by building a web platform in which human groups interact to both respond to questions likely to help predict a behavioural outcome and pose new questions to their peers. This results in a dynamically-growing online survey, but the result of this cooperative behaviour also leads to models that can predict user’s outcomes based on their responses to the user-generated survey questions. Here we describe two web-based experiments that instantiate this approach: the first site led to models that can predict users’ monthly electric energy consumption; the other led to models that can predict users’ body mass index. As exponential increases in content are often observed in successful online collaborative communities, the proposed methodology may, in the future, lead to similar exponential rises in discovery and insight into the causal factors of behavioural outcomes.

There are many problems in which one seeks to develop predictive models to map between a set of predictor variables and an outcome. Statistical tools such as multiple regression or neural networks provide mature methods for computing model parameters when the set of predictive covariates and the model structure are pre-specified. Furthermore, recent research is providing new tools for inferring the structural form of non-linear predictive models, given good input and output data. However, the task of choosing which potentially predictive variables to study is largely a qualitative task that requires substantial domain expertise. For example, a survey designer must have domain expertise to choose questions that will identify predictive covariates. An engineer must develop substantial familiarity with a design in order to determine which variables can be systematically adjusted in order to optimize performance.

**10. Feature Selection Based on Class-Dependent Densities for High-Dimensional Binary Data.**

**Kashif Javed, Haroon A. Babri, and Mehreen Saeed. Ieee transactions on knowledge and data engineering, vol. 24, no. 3, march 2012.**

Data and knowledge management systems employ feature selection algorithms for removing irrelevant, redundant, and noisy information from the data. There are two well-known approaches to feature selection, feature ranking (FR) and feature subset selection (FSS). In this paper, we propose a new FR algorithm, termed as class-dependent density-based feature elimination (CDFE), for binary data sets. Our theoretical analysis shows that CDFE computes the weights, used for feature ranking, more efficiently as compared to the mutual information measure. Effectively, rankings obtained from both the two criteria approximate each other. CDFE uses a fill trapper approach to select a final subset. For data sets having hundreds of thousands of features, feature selection with FR algorithms is simple and computationally efficient but redundant information may not be removed. On the other hand, FSS algorithms analyze the data for redundancies but may become computationally impractical on high-dimensional data sets. We address these problems by combining FR and FSS methods in the form of a two-stage feature selection algorithm. When introduced as a pre processing step to the FSS algorithms, CDFE not only presents them with a feature subset, well in terms of classification, but also relieves them from heavy computations. Two FSS algorithms are employed in the second stage to test the two-stage feature selection idea. We carry out experiments with two different classifiers (naive Bayes’ and kernel ridge regression) on three different real-life data sets (NOVA, HIVA, and GINA) of the “Agnostic Learning versus Prior Knowledge” challenge. As a stand-alone method, CDFE shows up to about 92 percent reduction in the feature set size. When combined with the FSS algorithms in two-stages, CDFE significantly improves their classification accuracy and exhibits up to 97 percent reduction in the feature set size. We also compared CDFE against the winning entries of the challenge and found that it outperforms the best results on NOVA and HIVA while obtaining a third position in case of GINA.

**11. Techniques, Process, and Enterprise Solutions of Business Intelligence.**

**Li Zeng, Lida Xu, Zhongzhi Shi, Maoguang Wang, and Wenjuan Wu. 2006 IEEE Conference on Systems, Man, and Cybernetics October 8-11, 2006, Taipei, Taiwan.**

Business Intelligence (BI) has been viewed as sets of powerful tools and approaches to improving business executive decision-making, business operations, and increasing the value of the enterprise. The technology categories of BI mainly encompass Data Warehousing, OLAP, and Data Mining. This article reviews the concept of Business Intelligence and provides a survey, from a comprehensive point of view, on the BI technical framework, process, and enterprise solutions. In addition, the conclusions point out the possible reasons for the difficulties of broad deployment of enterprise BI, and the proposals of constructing a better BI system. As businesses continue to use computer systems for a growing number of functions in today’s competitive, fast-evolving world, most companies face the challenges of processing and analyzing huge amounts of data and turning it into profits. They have large volumes of detailed operational data, but key business analysts and decision makers still cannot get the answers they need to react quickly enough to changing conditions because the data are spread across many departments in the organization or are locked in a sluggish technology environment. In these cases, Business Intelligence (BI) is presented, which are sets of tools, technologies and solutions designed for end users to efficiently extract useful business information from oceans of data. Nowadays, BI has been viewed as sets of powerful tools and approaches to increasing the value of the enterprise. More and more business sectors have deployed advanced BI solutions to enhance their competitiveness since it is important to the effective and efficient running of the enterprises.

**12. Support vector machine with adaptive parameters in financial time series forecasting.**

**Cao, L.J. ; Dept. of Mech. Eng., Nat. Univ. of Singapore, Singapore ; Tay, F.E.H. EEE Transactions on, On page(s): 1167 - 1178 Volume: 19, Issue: 7, July 2008**

A novel type of learning machine called support vector machine (SVM) has been receiving increasing interest in areas ranging from its original application in pattern recognition to other applications such as regression estimation due to its remarkable generalization performance. This paper deals with the application of SVM in financial time series forecasting. The feasibility of applying SVM in financial forecasting is first examined by comparing it with the multilayer back-propagation (BP) neural network and the regularized radial basis function (RBF) neural network. The variability in performance of SVM with respect to the free parameters is investigated experimentally. Adaptive parameters are then proposed by incorporating the nonstationarity of financial time series into SVM. Five real futures contracts collated from the Chicago Mercantile Market are used as the data sets. The simulation shows that among the three methods, SVM outperforms the BP neural network in financial forecasting, and there are comparable generalization performance between SVM and the regularized RBF neural network. Furthermore, the free parameters of SVM have a great effect on the generalization performance. SVM with adaptive parameters can both achieve higher generalization performance and use fewer support vectors than the standard SVM in financial forecasting.

**13. Financial time series modelling with discounted least squares back-propagation.**

**A. N. Refenes, Y. Bentz, D. W. Bunn, A. N. Burgess, and A. D. Zapranis,  "Financial time series modeling with discounted least squares back-propagation",  Neuro computing,  vol. 14,  pp.123 -138 1997.**

We propose a simple modification to the error back propagation procedure which takes into account gradually changing input-output relations. The procedure is based on the principle of Discounted least squares whereby learning is biased towards more recent observations with long term effects experiencing exponential decay through time. This is particularly important in systems in which the structural relationship between input and response vectors changes gradually over time but certain elements of long term memory are still retained. The procedure is implemented by a simple modification of the least-squares cost function commonly used in error back propagation. We compare the performance of the two cost functions using both a controlled simulation experiment and a non-trivial application in estimating stock returns on the basis of multiple factor exposures. We show that in both cases the DLS procedure gives significantly better results. Typically, there is an average improvement of above 30% (in MSE terms) for the stock return modelling problem.

**14. Stock Market Value Prediction Using Neural Networks.**

**T. Kimoto, K. Asakawa, M. Yoda, and M. Takeoka,  "Stock market prediction system with modular neural networks",  Neural Networks in Finance and Investing,  pp.343 -357 1993**

Neural networks, as an intelligent data mining method, have been used in many different challenging pattern recognition problems such as stock market prediction. However, there is no formal method to determine the optimal neural network for prediction purpose in the literature. In this paper, two kinds of neural networks, a feed forward multi layer Perception (MLP) and an Elman recurrent network, are used to predict a company’s stock value based on its stock share value history. The experimental results show that the application of MLP neural network is more promising in predicting stock value changes rather than Elman recurrent network and linear regression method. However, based on the standard measures that will be presented in the paper we find that the Elman recurrent network and linear regression can predict the direction of the changes of the stock value better than the MLP. From the beginning of time it has been man’s common goal to make his life easier. The prevailing notion in society is that wealth brings comfort and luxury, so it is not surprising that there has been so much work done on ways to predict the markets. Therefore forecasting stock price or financial markets has been one of the biggest challenges to the AI community. Various technical, fundamental, and statistical indicators have been proposed and used with varying results. However, none of these techniques or combination of techniques has been successful enough. The objective of forecasting research has been largely beyond the capability of traditional AI research which has mainly focused on developing intelligent systems that are supposed to emulate human intelligence. By its nature the stock market is mostly complex (non-linear) and volatile. With the development of neural networks, researchers and investors are hoping that the market mysteries can be unravelled.

**15. Application of a Case Base Reasoning Based Support Vector Machine for Financial Time Series Data Forecasting.**

**Pei-Chann Chang, Chi-Yang Tsai, Chiung-Hua Huang, Chin-Yuan Fan 5th International Conference on Intelligent Computing, ICIC 2009 Ulsan, South Korea, September 16-19, 2009 Proceedings**

This paper establishes a novel financial time series-forecasting model, by clustering and evolving support vector machine for stocks on S&P 500 in the U.S. This forecasting model integrates a data clustering technique with Case Based Reasoning (CBR) weighted clustering and classification with Support Vector Machine (SVM) to construct a decision-making system based on historical data and technical indexes. The future price of the stock is predicted by this proposed model using technical indexes as input and the forecasting accuracy of the model can also be further improved by dividing the historic data into different clusters. Overall, the results support the new stock price predict model by showing that it can accurately react to the current tendency of the stock price movement from these smaller cases. The hit rate of CBR-SVM model is 93.85% the highest performance among others.

**16. Self-Organizing Roles on Agile Software Development Teams.**

**Rashina Hoda,Member, IEEE, James Noble, Member, IEEE, and Stuart Marshall, Member, IEEE. IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 39, NO. 3, MARCH 2013**

Self-organizing teams have been recognized and studied in various forms—as autonomous groups in socio-technical systems, enablers of organizational theories, agents of knowledge management, and as examples of complex-adaptive systems. Over the last decade, self-organizing teams have taken centre stage in software engineering when they were incorporated as a hallmark of agile methods. Despite the long and rich history of self-organizing teams and their recent popularity with Agile methods, there has been little research on the topic within software engineering. Particularly, there is a dearth of research on how Agile teams organize themselves in practice. Through a Grounded Theory research involving 58 Agile practitioners from 23 software organizations in New Zealand and India over a period of four years, we identified informal, implicit, transient, and spontaneous roles that make Agile teams self-organizing. These roles—Mentor, Coordinator, Translator, Champion, Promoter, and Terminator—are focused toward providing initial guidance and encouraging continued adherence to Agile methods, effectively managing customer expectations and coordinating customer collaboration, securing and sustaining senior management support, and identifying and removing team members threatening the self-organizing ability of the team. Understanding these roles will help software development teams and their managers better comprehend and execute their roles and responsibilities as a self-organizing team. SELF-ORGANIZING teams have been recognized and studied in various forms—as autonomous groups in socio-technical systems as early as in the 1950s, as enablers of holographic organizations in organizational theory and as agents of knowledge creation and management around the 1980s, and as examples of entities exhibiting spontaneous order in Complex Adaptive Systems (CAS) in the 1990s. More recently, with the rise of Agile methods in the late 1990s and early 2000s, self-organizing teams took centre stage in the software engineering arena when they were incorporated as a hallmark of Agile software development. Self-organizing teams are at the heart of Agile software development Self-organizing Agile teams are composed of “individuals [that] manage their own workload, shift work among themselves based on need and best fit, and participate in team decision making”. Self-organizing teams must have common focus, mutual trust, respect, and the ability to organize repeatedly to meet new challenges. The scrum method specifically mentions self-organizing agile teams and the concept of “empowered” teams has recently been added to XP. Self-organizing teams are not only seen as enabling Agile engineering practices, but also as capturing the spirit of Agile values and principles, which focus on human and social aspects of software engineering. Self-organizing teams is one of the principles behind the Agile Manifesto and have been identified as one of the critical success factors of Agile projects.