Faculty of Engineering and Information Technology
University of Technology, Sydney

Coupling Analysis in Educational Data

A thesis submitted in partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

by

Mu Li

December 2015

CERTIFICATE OF AUTHORSHIP/ORIGINALITY

I certify that the work in this thesis has not previously been submitted for a degree nor has it been submitted as part of requirements for a degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are indicated in the thesis.

Signature of Candidate

Acknowledgments

First and foremost, I would like to express my sincere gratitude to my supervisor Prof. Longbing Cao for his continuous support, patience, motivation, enthusiasm, and immense knowledge. His guidance was tremendously valuable over the course of my PhD research. I could not have had a better advisor and mentor.

My sincere thanks is also extended to Dr. Jinjiu Li for providing me with his continuous support over the duration of my PhD. Without his professional guidance and consistant help, this thesis would not have been possible.

Many thanks to my fellow labmates at the Advanced Analystics Institute: Chuming Liu, Yin Song, Junfu Yin for the stimulating discussions, and sleepless nights that we shared whilst working together on a range of deadlines, and for all the fun we had over the last four years.

My sincere appreciation goes to the team leader, Zhigang Zheng for his kindness and expert help with the project. I am also grateful to my team members, Allen Lin, Chunming Liu and Wei Cao, for their hard work in the project teams.

Fianlly, my deep and abiding gratitude is extended to my wife and my parents, for their unconditional love and support over the course of my PhD candidature.

Mu Li
March 2015 @ UTS

Contents

List of Figures

List of Tables

Abstract

Educational data analysis refers to techniques, tools, and research designed to automatically extract meaning from large repositories of data generated by or related to people's learning activities in educational environments. It is a research field which focus on helping policymakers and administrators understand how analytics and data mining can be applied for the purposes of educational improvement. Unfortunately, most research on educational data only by applying the existing machine learning or data mining algorithms, very few publications have discussed the character of the data itself. Traditional data mining algorithms have disadvantages, in that most of them assume the independent and identically distributed (IID) of data objects, attributes, and values. However, real world data usually contains strong couplings among values, attributes and data objects, and this represents a considerable challenge to existing methods and tools. This thesis focuses on utilizing coupling analysis in educational data analysis tasks. In particular, it focuses on two educational data analysis tasks: student performance prediction, and student social media sentiment analysis.

The student performance prediction task is firstly examined. This thesis begins with the most straightforward method which integrates coupling similarities as the distance for a weighted k-nearest centroid classifier. This method considers not only the intra-coupled similarity within an attribute but also the inter-coupled similarity between attributes. Computational cost is high for coupling analysis. Hence, a more efficient method is proposed that selects the centroid objects instead of all objects in the nearest neighbor search process. Furthermore, integrating support vector machines with coupled similarity. The original SVMs is designed for numerical data. This thesis develops a novel pairwise SVMs that use the coupled similarity metric as a kernel between data objects with nominal attributes. The experiment result shows the two proposed methods outperform the traditional SVMs and other popular classification methods on various public data sets, and the student performance prediction task.

Secondly, the student social media sentiment analysis is examined. Unlike linguistic methods, this thesis learns how to classify student sentiment by applying data mining on the labeled historical data. Most previous research employs the vector-space model for text representation and analysis, however, the vector-space model does not utilize the information about the term to term relationships. In other words, the traditional text mining techniques assume the relations between term to term are independent and identically distributed (IID). This thesis introduces a novel term representation by involving coupling relations between neighbors. This coupling representation provide much richer information which enables us to create a coupled similarity metric from document to document, and a coupling document similarity based k-nearest centroid classifier applied to the classification task. Experiments verify that the proposed approach outperforms the classic vector-space based classifier and displays distinct advantages and richness in terms of student social media sentiment analysis tasks.

Finally, due to the complexity of the proposed algorithm and the enormous amount of the educational related data source, a scalable educational data mining platform is in great demand. Hence, with the help of the Spark cluster, a novel coupling similarity based learning approach has been proposed to cater for the big data learning problem by parallelizing the coupled similarity calculation process. Further, the parallel k-NN for classification and k-Means for the clustering task has been proposed. Compared to the original algorithms, the experimental results show that the proposed methods not only outperforms the clustering and classification performance of the baselines, but also represent a huge improvement on the data scale in terms of the time efficiency. Accordingly, the proposed framework has already been implemented, a scalable educational data analysis platform with coupling analysis will serve to meet a host of future challenges.

Chapter 1

Introduction

1.1 Background

Education ensures a bright future for humankind. Educational data analysis is a technique to enhance the existing education methods, processes and concepts. At the same time, coupling analysis is a novel data analysis concept which aims to learn the relation and interaction that connects two or more aspects. Analyzing coupling relationships is fundamental but challenging. Coupling analysis has great potential in terms of building a deep understanding of educational data and handling challenges that haven't been addressed well by the existing machine learning or data mining tools. The fusion of the coupling analysis and the educational data is the primary objective and contribution of this thesis. This chapter firstly introduces the detailed concepts of coupling analysis and educational data analysis; secondly, it provides an example to illustrate the advantages of applying coupled similarity to the student data set; and thirdly, it points out the limitations and challenges of existing methods along with outlining the contribution of this thesis.

1.1.1 Coupling Analysis

Most of the current algorithms and methods in statistics, data mining and machine learning are built on the assumption of the IIDness(Identically Independent Distributed). IIDness assumes the IID in every perspective of the data source, e.g. the objects, attributes and values. (Cao 2014) summarized a coupling learning system for complex interactions. The coupling analysis had the primary purpose of revealing the implicit coupling relations among the data in all aspects. In general, coupling is a kind of relationship between two or more aspects. The following relation can be identified as a coupling relationship: co-occurrence, neighborhood, dependency or linkage. There are many examples of these aspects, e.g. data objects, object clusters, and object attribute values. The coupling relation can be represented as follows:

- Entity coupling: The coupling relation of entities is the basic concept of coupling relations. This kind of coupling relation exists within and between data sources, data objects, data attribute and data values. Moreover, intra-coupling is the coupling relation within an entity, and the inter-coupling appears between entities.

- Attribute coupling: the attribute coupling is about the entity attribute distribution, structure, dynamics, logic or linkages. Intra-attribute coupling is the coupling relations between the values of an attribute. Moreover, one attribute is more or less related to another attribute, and this leads to the inter-attribute coupling between attributes.

In coupling analysis, there are more perspectives that can be considered. These include the problem domain, data presentation, knowledge representation, and coupling characteristics.

- Problem domain perspective, the coupling relation may present in the cross-domain. For example, the college entrance examination result in one school may affect another schools entry rates to university. However, most researchers only consider the aspects within one specific domain.

- Data format perspective, the coupling relation may appear in different data formats. For instance, numerical data, categorical data, graphic, and textual. The coupling relation within the same data formats and the coupling relation between different data formats should also be considered.

- Knowledge representation perspective, this coupling relation may present in terms of the semantic, syntactic, or graphic attributes. For example, in the text mining task, the term to term relation is often built on grammatical relations.

- Coupling characteristics perspective, the coupling relation may appear in terms of a visible or invisible way, quality or quantity priority, and shallow or deep analysis.

This thesis discusses coupling analysis mostly from the similarity perspective, and similarity analysis is a widely used criteria for many machine learning and data mining algorithms. For example, a typical aspect of these usages is clustering whereby the similarity is defined in terms of the following levels: clusters to clusters, attributes to attributes, data objects to data objects, and attribute values to attribute values. The similarity between the clusters is built on top of the similarity of the data objects, and the similarity of the data objects is derived from the similarity of the attribute values. The basic definition of the similarity is the more the resemblance between each other, the larger the similarity is. The similarity measure can be categorized into two types, the similarity for the nominal attribute and the similarity for continuous data. The main research within this thesis is based on the nominal data which has a finite number of values for each attribute.

1.1.2 Educational Data Analysis

Educational data analysis uses the machine learning and data mining technology, tools, and the design of automatic process to extract meaning from

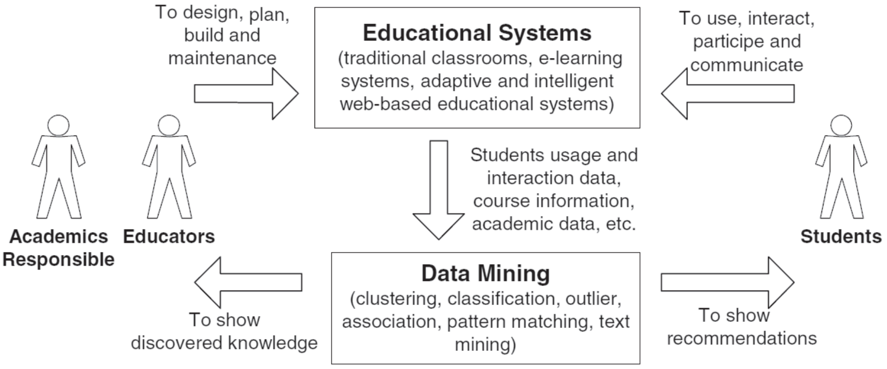

To design, plan, build and maintenance
Educational Systems
(traditional classrooms, e-learning systems, adaptive and intelligent web-based educational systems)
To use, interact, participe and communicate
Students usage and interaction data, course information, academic data, etc.
Academics Responsible
Educators
Students
Data Mining
(clustering, classification, outlier, association, pattern matching, text mining)
To show discovered knowledge
To show recommendations

Figure 1.1: The Work Flow of Educational Data Analysis

the data generated by people's learning activities in the educational environment. For example, some learning management systems (LMS) track information, such as when each student seeks learning object access, how many times they visit the object, and how many minutes the learning object is displayed on the student's computer screen. As another example, intelligent tutoring systems record the data of every student who seeks solutions to problems; the time is takes them to submit their data, whether the solution matches the expected outcomes of the student, the time that has passed since the data was submitted, the order of the solution components in the interface, and so on. The precision of the data in even a relatively short session computer-based learning environment may produce a lot of process data to be analyzed. Figure 1.1 shows the work flow of the Educational data analysis. This thesis focus on two research areas of the educational data analysis, student performance prediction and student sentiment analysis.

Performance Prediction

The objective of prediction is to estimate the unknown value of a variable that describes the student. In education, the values typically predicted are performance, knowledge, score and mark. These values can be numerically continuous values or categorical discrete values. Regression analysis finds the relationship between a dependent variable and one or more independendent variables (Draper, Smith & Pownell 1966). Classification is a procedure in which individual items are placed into groups based on quantitative information regarding one or more characteristics inherent in the questions and based on a training set of previously labeled items (Espejo, Ventura & Herrera 2010). Prediction of a students performance is one of the oldest and most popular applications of data analysis in education, and different techniques and models have been applied. An example (Figure 1.2) can be introduced to illustrate the application of student performance prediction. A university strives to improve the average student performance due to the seriously competitive nature of the educational market. This university proposed the notion of giving an early warning to the high risk student before the student actually takes the final examination. However, the challenge is how to define a high risk student and how to find them. A very simple rule was put forward that an international student proceeding towards a bachelor's degree is a high risk student. However, this results in the university needing to call more than 7000 students each year which is very costly in terms of human resources and is simply not efficient. However, if it has been applied the data analysis technique to their problem(e.g. using data mining or machine learning algorithms to learn the model from the historical student data that already has the exam results, and then use the model to predict student performance) this allows for early intervention to only the most high risk students. This intuitive strategy significantly reduces the number of students, and considerably reduces the cost to the university.

Sentiment Analysis

Sentiment analysis has been defined as the computational study of opinions, feelings and emotions expressed in texts (Liu 2010). For the sake of simplifying the development of an emotion recognition tool, to avoid complex and potentially controversial definitions of emotions and sentiments. In this respect, a simplified definition of emotion can be defined as ”a personal positive or negative feeling or opinion”. An example of a sentence transmitting

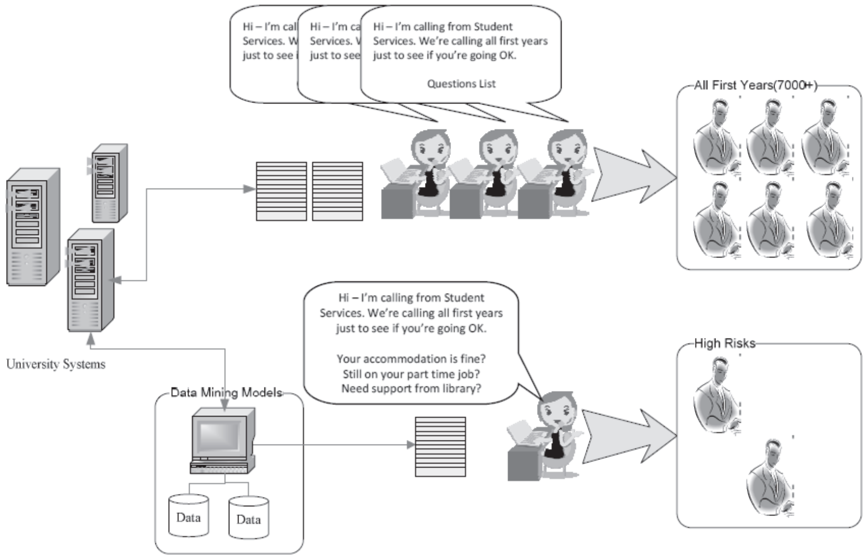

Hi – I'm cal
Services. W
just to see i
Hi – I'm ca
Services. V
just to see
Hi – I'm calling from Student
Services. We're calling all first years
just to see if you're going OK.
Questions List
All First Years(7000+)
University Systems
Hi – I'm calling from Student
Services. We're calling all first years
just to see if you're going OK.
Your accommodation is fine?
Still on your part time job?
Need support from library?
High Risks
Data Mining Models
Data
Data

Figure 1.2: Application of Student Performance Prediction

a positive sentiment would be "I love it!", Whereas "It is a terrible movie" transmits a negative sentiment. A neutral sentiment does not express any feeling. Most of the works in this research area focus on classifying texts according to their sentiment polarity, which can be positive, negative or neutral (Pang & Lee 2008). Therefore, it can be considered as a text classification problem since its goal consists of categorizing documents within classes by means of algorithmic methods. This thesis show a very simple example that is produced by the algorithm that was developed from this thesis (Figure 1.3). The proposed method learns the textural features of six million labeled sentiment tag's data from Twitter's public access data, and does the sentiment analysis from the text content posted by students from social media and university internal online media like student forums. The current sentiment indicator and historical trends are all accessible by students and the university management team. This method may result students and management staffs making a better decisions or creating a adaptable plans.

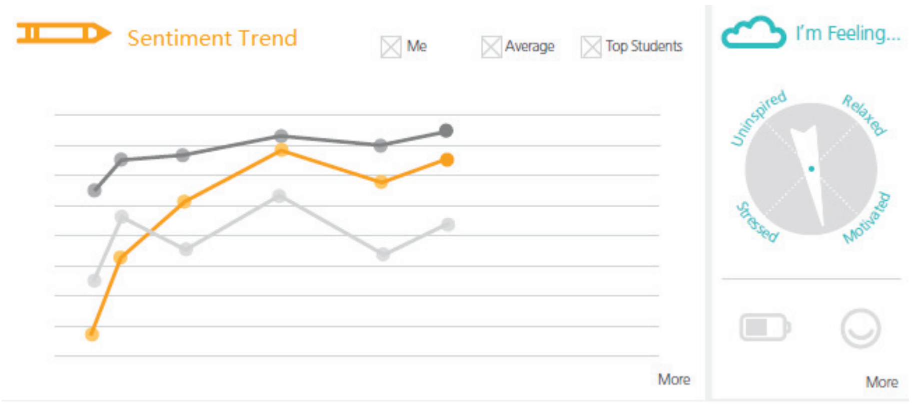


Figure 1.3: Application of Student Sentiment Analysis

1.1.3 Coupling Analysis in Educational Data

After understanding the basic concepts introduced above, this thesis discusses a simple example to illustrate the concept of applying coupling analysis on educational data sets. Most of the previous research has treated the data with independent identical distribution(iid)

Taking an educational data (Romero & Ventura 2010) for example. This data aims to analyze student demographical features and behaviors on campus and assesses the risk of failing in the target courses. In table 1.1, six students are divided into two classes, one is high risk and the other is low risk. As shown in the table, there are three categorical features: country, educational background and economic status. If it uses the overlap similarity (Li & Li 2010) to measure the similarity between the countries "*China*" and "*Japan*'" this could be 0. However, if it have more information about the student, *China* and *Japan* share similarities like location, culture and characteristics. Therefore, students from these two countries may be more similar than other countries. If it considers existing features' information like their co-relations for the similarity measurement, the metric should be better than a binary judgement 1 or 0. Take previous examination results as an example where the Chinese and Japanese student may both be good at maths and bad at English. If it takes the previous exam result's impact on

Table 1.1: Student Information Table

ID	Country	Education	Math	English	Performance
1	USA	TAFE	M	H	M
2	Australia	TAFE	L	H	L
3	China	HighSchool	H	L	H
4	China	Bachelor	H	L	H
5	Japan	Bachelor	H	L	H
6	Japan	HighSchool	H	L	H

the similarity of their nationality into account, the Chinese may be similar to Japanese. Moreover, if every feature has considered this coupling relation, the similarity between students could be much more reasonable and accurate than simply the sum of the overlapped value count. This kind of similarity can also be found in other example, e.g. the similarity between "*Bachelor*" and "*HighSchool*" is equal to that of "*TAFE*" and "*HighSchool*" and this is because of the overlap similarity. Intuitively, the similarity of the first pair should be greater since they drop into the same class H.

This example shows that it is possible to analyze the similarity between categorical variables properly rather than just comparing the binary matches. Furthermore, numeric distance cannot capture the significant correlations among nominal values. It is worthwhile to design an effective and efficient method to measure the similarity among nominal variables.

This coupling concept can be applied in most areas of Education data, but it does not have enough resources and time to focus on all of them. This thesis chooses two representative targets in educational data analysis which straightforwardly apply the classification and text mining algorithms. As discussed before, the first one is the prediction of the student performance and the second one is student sentiment analysis. There is enormous research have been done in this two areas, the detail literature review will in the next chapter, the following section will discuss the previous research and point out their strengths and their weaknesses.

1.2 Research Issues and Objectives

For the student performance prediction task, most of the researchers just directly apply the existing data mining techniques on their educational data. This approach may not be a sophisticated way to get a good prediction result. The main limitations and challenges of the current research on the prediction of student performance are detailed below:

- Existing research is too specific to a small domain, and most of the research is limited to a particular context. There is very limited research which attempts to create a standard student performance prediction process and method which can be widely used.

- The traditional classification method is usually built on the assumption of data is the independent identically distributed(iid), not only for the data objects but also for the features. These methods treat all the features separately, e.g. the Naive Bayesian classifier or logistic regression classifier. The relation between the features and data objects has not been considered. The assumptions are too strong and the reality and complexities of the student performance prediction task are not properly addressed. Indeed, it is not possible to support depth analysis interior insights and to get comprehensive output.

- A large proportion of the data source for the student performance prediction is categorical data. Most of the classification algorithms are designed for numerical data, such as SVM and logistic regression. Unfortunately, the state of art research hasn't addressed the problem. It has just directly utilize the existing classification method. Nevertheless, a few works have done the pre-processing that transforms the categorical data into numerical data. This transformation may not present the original data correctly as it can lose the unique information for the categorical data.

- For the student sentiment analysis task, most of researchers follow the concept from the traditional linguistic perspective. They make handcraft features or meta-data to transform the text into a sentiment metric. It heavily relied on the context of the particular domain. In a clear trend that the machine learning or data mining based method has begun to play the more and more important role in the sentiment analysis task.

- In relation to sentiment analysis, there are very few works focus on student sentiment. The works found are usually adaptations of already presented methods for text sentiment analysis. They just use some machine learning methods to classify movie reviews or online forums.

- In recent years, there has been an increasing amount of information delivered through social networks. However, a few researchers are focusing on applying sentiment analysis to these data, except (Go, B-hayani & Huang 2009) (Pak & Paroubek 2010). The recent research work on the student sentiment analysis task lacks the social media data of the student from applications like Twitter or Facebook. Nowadays, social media represents a data source which has an enormous, accessible volume of information for the educational purposes. Social media data also contains more characteristic features than traditional data sources.

- Lastly, for both student performance prediction task and text sentiment analysis task, most of the previous research has only experimented on tiny data sets. However, most of the modern universities have an enormous number of students. If it has been considered the historical data for training purposes, the data set becomes even bigger. Moreover, if researchers want to undertake a regional analysis or national analysis, the current method cannot handle the size of the data.

1.3 Research Contributions

For the sake of tackling the research limitations and challenges as mentioned above, this thesis propose targeted research objectives from the following perspective.

Coupled similarity based classifier for student performance prediction

By adapting the coupling analysis concept, this thesis propose the coupled similarity base student performance prediction algorithm. The coupled similarity for the categorical data framework reveals the relation between each attribute and data objects. It is like a feature learning schema that can capture the characteristics of the data itself. It prevents problems in the previous research that are sometimes too specific for a certain context. However, it does not mean that the algorithm can handle everything entirely automatically, the data pre-process, and human's expertise are still essential to the final prediction result. Furthermore, the proposed methods are under the assumption of non-IIDness, which means it does not assume that the data is independent identical distributed. This is a break through as it can produce much richer information than the existing method. So too, as the coupled similarity is designed for categorical data, it can prevent the shortage of the similarity metric for the categorical data.

Coupled similarity based text mining for student sentiment analysis

This thesis proposes a novel term to term coupling relation analysis framework that can learn the relationship between each term automatically. The system is carefully designed and it learns the term to term coupling relation from the student social media posts. It successfully solves the problem of the handcraft features which can only adapt to a particular domain or data set. Furthermore, this thesis collects a considerable amount of student Twitter data which can serve to reveal the true sentiment of the student.

This thesis mainly concentrates on how to integrate the coupling concept into the educational data analysis domain. In such a way, it aims to improve the outcome of the previous work. The proposed coupling analysis in educational data related publications is listed in Appendix A. The research contributions are listed individually as below:

1.3.1 Coupled Similarity Based K-centroid Classifier

This is the first step which employs the coupling analysis concept in the educational data. A novel method called the coupled similarity based k-centroid classifier is proposed. In general it contains the following advantages:

- A classifier that utilize the coupling similarities and apply them with a weighted k-nearest classifier to do the student performance prediction from the source data.
- A novel metric which indicates three perspectives of coupling relations: the intra-coupling of values within one attribute, the inter-coupling of values that involves all the attributes, and the object coupled similarity which serves as a metric for the similarity between objects.
- An improved k-centroid classifier which significantly boost the efficiency of the student performance prediction task.
- Experimental evaluations demonstrate the effectiveness of the proposed method and outperform the existing student performance prediction task. The proposed method has been successfully used in the real world project titled "student at risk alert system".

1.3.2 Coupled Similarity Based Pairwise SVM Classifier

This is a further step to explore the effectiveness of the coupling concept and the coupled similarity to categorical data. It successfully integrates the state of art classification algorithm with the proposed coupled similarity which enhances the classification performance significantly. The detailed contribution can be described as follows:

- Propose an appropriate strategy which fuses the coupled similarity into the SVM optimization process. It combins both the advantages of the SVM learning schema and the coupled similarity for learning the data relations.
- Develops a caching tactics which dramatically improve the time efficiency of the proposed classification method.
- Conceivably, this technique will be used in the student performance prediction task. And the experiment result illustrates the proposed method outperform the classic prediction method on both the educational data and the UCI public data sets. The proposed method also has been successfully applied in the real world system and is named ”killer subjects analysis”.

1.3.3 Coupled Term to Term Relations Extractor

The information about the student is not just stored in the organized structured data sources. There is a vast amount of public unstructured data sources which also contribute to the educational data analysis tasks. Social media like Twitter and Facebook could be the largest and most easily accessible data source. This thesis have imported more than six million Twitter posts from the public for training, and test the model on the students who follow any particular department or institution within our university. The traditional text mining technologies assume the IID(independent identically distributed) of the terms(words). This thesis follows the coupling concept by proposing two novel strategies that extract the coupling relations from term to term. The main contribution listed below:

- This work proposes a neighborhood co-occurrence based coupling relation criteria by accumulating a sliding window as a vector feature for each term. This criteria based similarity extracts much more characteristic features of the term than the term frequency based method.

- Moreover, a coupled similarity based on neural network has also been conducted. Like the progressive of the proposed two classifiers, the neural network based coupled similarity involves the learning process which has a superior ability to capture more precisely the relation criteria than the first strategy.
- This thesis creates a novel sentiment metric by just using the Twitter tags. The learning algorithm with the coupled similarity from term to term, automatically builds the model from the training data. There are no human expertise and handcraft features required. The proposed algorithm has been implemented as a function in a ”student dashboard” system.

1.3.4 Large Scale Coupling Analysis Framework

Due to the complexity of the aforementioned algorithms, it requires considerable time to be executed. Furthermore, as the student related data set grows larger and larger, it becomes necessary to develop a framework that can handle both the large data set and complex learning algorithms. This thesis proposes a platform for large-scale coupling analysis, and the principal contributions of this part are as follows:

- Proposal of a novel map-reduce based coupled similarity computation algorithm which can run on the Spark cluster. It is a nearly linear scalable algorithm that can handle very a large data set if there are sufficient computational resources. It can be applied in any similarity based algorithms even though this thesis is mainly focused on the educational data.
- Proposal of a scalable coupled similarity based k-NN classifier. It can be directly applied to the student performance prediction task without

any modifications.

- Proposal of a novel scalable clustering method based on coupled similarity. It addresses a novel divergence strategy for the categorical data with coupled similarity. The experiment shows that it can run on a data set of more than four million records while no previous research for clustering categorical data has been done that scale.

- Development of a sophisticated framework with scalability on both similarity computation and the learning algorithms which are already be used in the real world educational data analysis tasks.

1.4 Thesis Organization

The structure of the research work in this thesis is exhibited in Figure 1.4. The thesis is organized as follows.

Chapter 1 provides an introduction to the basics of coupling analysis and the concept of educational data analysis. It describes the current research limitations and challenges and discusses the motivation and objectives. Moreover, it introduces the research issue and main contributions.

Chapter 2 comprehensively reviews the related literatures, summarizes the recent research directions and categorizes them.

Chapter 3 proposes a novel coupled similarity based k-centroid classifier which is utilized in the student performance prediction task. It considers both the prediction accuracy and the efficiency.

Chapter 4 refines the k-centroid classier and introduces a coupled similarity base pairwise SVM classifier which leads to a better prediction accuracy. It has also been used in the real world student performance prediction task.

Chapter 5 puts forward the concept of employing coupling analysis in the text mining domain in order to examine students' sentiment in the social media. The experiment shows the proposed method outperform the classic methods on both public data and student related data sources.

Chapter 6 transforms the previous research into the big data platform which makes a scalable coupling analysis framework for educational data analysis.

Chapter 7 concludes the thesis and outlines the scope of future work.

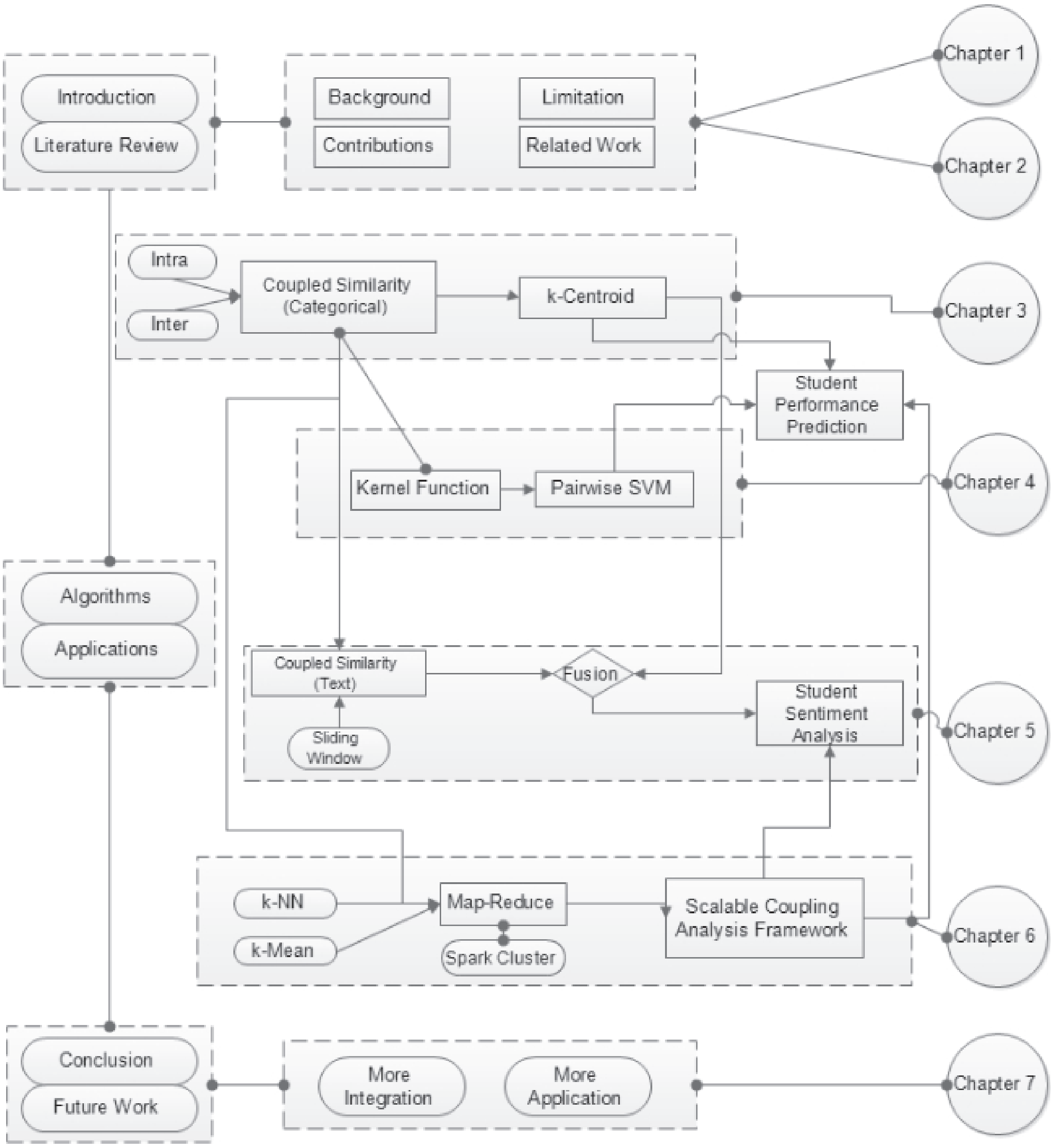


Figure 1.4: The Profile of the Research Work of This Thesis

Chapter 2

Literature Review and Foundation

2.1 Foundation

2.1.1 Coupled Similarity for Categorical Data

The coupling analysis on categorical data proposed by (Wang, Cao, Wang, Li, Wei & Ou 2011) is the main foundation for this thesis. In this thesis, in order to make the method more appropriate for analyzing real world data, the coupled similarity criterion was used to add both intra-coupled similarity and inter-coupled similarity to to measure the coupling relation between data features. The coupling distance (COS) measure was based on considering both Intra-coupling and Inter-coupling Attribute Value Similarities ($IaAVS$ and $IeAVS$), which capture the attribute value frequency distribution and feature dependency aggregation with a high learning accuracy.

Formal definition

In this section, ***Coupling Attribute Value Similarity (CAVS)*** is used in terms of both intra-coupling and inter-coupling value similarities. When considering the similarity between attribute values, "intra-coupling" indicates the involvement of attribute value occurrence frequencies within one feature, while "inter-coupling" means the interaction of other features with this attribute. For example, the coupling value similarity between B_1 and B_2 concerns both the intra-coupling relationship specified by the repeated times of values B_1 and B_2: 2 and 2, and the inter-coupling interaction triggered by the other two features (a_1 and a_3).

Suppose there is the ***Intra-coupling Attribute Value Similarity (IaAVS)*** measure $\delta_j^{Ia}(x, y)$ and ***Inter-coupling Attribute Value Similarity (IeAVS)*** measure $\delta_j^{Ie}(x, y)$ for feature a_j and $x, y \in V_j$, then $CAVS$ $\delta_j^A(x, y)$ is naturally derived by simultaneously considering both of them.

Definition 2.1 *The **coupling Attribute Value Similarity (CAVS)** between attribute values x and y of feature a_j is:*

$$\delta_j^A(x, y) = \delta_j^{Ia}(x, y) \cdot \delta_j^{Ie}(x, y) \tag{2.1}$$

where δ_j^{Ia} and δ_j^{Ie} are IaAVS and IeAVS, respectively.

Intra-Coupling Interaction

According to (Gan, Ma & Wu 2007), it is a fact that the discrepancy of attribute value occurrence times reflects the value similarity in terms of frequency distribution. Thus, when calculating attribute value similarity, consider the relationship between attribute value frequencies on one feature needs to be considered, intra-coupled similarity, which is defined as follows:

Definition 2.2 *Given an information table S, the **Intra-coupling Attribute Value Similarity (IaAVS)** between attribute values x and y of feature a_j is:*

$$\delta_j^{Ia}(x, y) = \frac{|g_j(x)| \cdot |g_j(y)|}{|g_j(x)| + |g_j(y)| + |g_j(x)| \cdot |g_j(y)|}. \tag{2.2}$$

where $|g_j(x)|$ denotes the frequency of value x in attribute j.

Hence, by taking into account the frequencies of categories, an effective measure ($IaAVS$) has been captured to characterize the value similarity in terms of occurrence times.

Inter-Coupling Interaction

In terms of *IaAVS*, the intra-coupled similarity is significant, i.e., the interaction of attribute values within one feature a_j. This does not, however, involve the couplings between other features $a_k (k \neq j)$ and feature a_j when calculating attribute value similarity. Accordingly, this dependency aggregation, i.e., inter-coupling interaction is discussed.

Definition 2.3 *The **Inter-coupling Relative Similarity based on Power Set (IRSP)** between attribute values x and y of feature a_j based on another feature a_k is:*

$$\delta^P_{j|k}(x, y) = \min_{W \subseteq V_k} \{2 - P_{k|j}(W|x) - P_{k|j}(\overline{W}|y)\}, \tag{2.3}$$

where $\overline{W} = V_k \backslash W$ is the complementary set of a set W under the complete set V_k, and $P_{k|j}(W|x) = \frac{|g_k(W) \bigcap g_j(x)|}{|g_j(x)|}$

According to the above discussion, define the similarity between the jth attribute value pair (x, y) on top of these four optional measures can be naturally defined by aggregating all the relative similarities on features other than attribute a_j.

Definition 2.4 *Given an information table S, the **Inter-coupling Attribute Value Similarity (IeAVS)** between attribute values x and y of feature a_j is:*

$$\delta^{Ie}_j(x, y) = \sum_{k=1, k \neq j}^{n} \alpha_k \delta_{j|k}(x, y), \tag{2.4}$$

where α_k is the weight parameter for feature a_k, $\sum_{k=1}^{n} \alpha_k = 1$, $\alpha_k \in [0, 1]$, and $\delta_{j|k}(x, y)$ is one of the inter-coupling relative similarity candidates.

Coupled Similarity

After specifying *IaAVS* and *IeAVS*, a coupled similarity between objects is built based on *CAVS*. Then, considering the sum of all these *CAVS*s analogous to the construction of Manhattan dissimilarity (Gan et al. 2007). Formally, there is:

Definition 2.5 *Given an information table S, the* ***Coupling Object Similarity (COS)*** *between objects u_{i_1} and u_{i_2}:*

$$COS(u_{i_1}, u_{i_2}) = \sum_{j=1}^{n} \delta_j^A(x_{i_1 j}, x_{i_2 j}), \tag{2.5}$$

where δ_j^A is the CAVS *measure defined in (3.4), $x_{i_1 j}$ and $x_{i_2 j}$ are the attribute values of feature a_j for objects u_{i_1} and u_{i_2} respectively, and $1 \leq i_1, i_2 \leq m$, $1 \leq j \leq n$.*

This coupled similarity metric involves both attribute value frequency distribution and attribute dependency aggregation in measuring attribute value similarity of the categorical data. It is a powerful tool for any distance or similarity based algorithms and the following subsections are the classifications that can be integrated and utilized in the student performance prediction task.

2.1.2 Classification Method

This section discusses the classification method related to the student performance prediction. This section firstly introduces the basic k nearest neighbor method and its variation k centroid classification method; secondly, it presents the support vector machines; finally, it discusses the pairwise support vector machines which adapt to the coupled similarity characteristics.

K-nearest Neighborhood

The easiest and most straightforward method of applying the coupled similarity to the classification task is the k-Nearest Neighbours algorithm. In data mining, the k-Nearest Neighbours algorithm is a non-parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression. In general, it can be defined as follows:

$$\begin{aligned} &\widehat{y}(x) = y^*_x \\ &where \\ &n^* = \arg\min_n dist(x, x_n) \end{aligned} \tag{2.6}$$

In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbours, with the object being assigned to the class most common among its k nearest neighbours (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbour. In k-NN regression, the output is the property value for the object. This value is the average of the values of its k nearest neighbours. k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally, and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms.

Both for classification and regression, it can be useful to weight the contributions of the neighbors so that the nearer neighbors contribute more to the average than the more distant ones. For example, a standard weighting scheme consists in giving each neighbor a weight of 1/d, where d is the distance to the neighbor.

The neighbours are taken from a set of objects for which the class (for k-NN classification) or the object property value (for the k-NN regression) is known. This can be thought of as the training set for the algorithm though no explicit training step is required.

A shortcoming of the k-NN algorithm is that it is sensitive to the local structure of the data. The algorithm has nothing to do with the k-means and is not to be confused with it, although it is another popular machine learning technique.

The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples.

In the classification phase, k is a user-defined constant, and an unlabeled vector (a query or test point) is classified by assigning the label which is most frequent among the k training samples nearest to that query point.

A commonly used distance metric for continuous variables is Euclidean distance. For discrete variables, such as for text classification, another metric can be used, such as the overlap metric (or Hamming distance). In the context of gene expression micro-array data, for example, k-NN has also been employed with correlation coefficients such as Pearson and Spearman. Often, the classification accuracy of k-NN can be improved significantly if the distance metric is learned with specialized algorithms such as Large Margin Nearest Neighbor or Neighbourhood components analysis.

In this thesis, the coupled similarity is used as the distance in the k-NN algorithm which is concerned with both intra-coupling and inter-coupling perspective. Because the complexity of the k-NN algorithm is $O(n^2)$, and the complexity of the coupled similarity is $O(n^2R^3)$. n is the number of instances, and R is the distinct value of the data, respectively. Hence, the brute force k-NN with coupled similarity is not sophisticated and there should be some way to improve the efficiency. Especially in the student performance prediction task, the number of students in a standard university should be more than most the UCI classification data source. The next section will illustrate the k-nearest centroid classification method which is much faster in the completion of the classification task.

K-nearest Centroid

A nearest centroid or nearest prototype classifier is a classification model that assigns to the label of the class of training samples whose mean (centroid) is closest to the observation. Briefly, the method computes a standardized mean (centroid) for each class. Nearest centroid classification takes the instance profile as a new sample and compares it to each of this class centroid. The class that the centroid is closest to, over a selected distance, is the predicted class for that new sample. Intuitively, in this thesis, the coupled similarity is used as the distance metric. By doing this, the time consuming classification period can be reduced from $n*N$ to $n*C$, where n is the number of instances that need to be classified, N is the number of instances of the training samples and C is the number of centroid by pre-processing the training samples. In most situations, the $C << N$, which means the time cost for the classification task can be dramatically reduced by applying the k-centroid method.

The K-nearest centroid classification method can be defined as follows, given labeled training samples: $\{(\vec{x_1}, y_1), (\vec{x_2}, y_2), ...(\vec{x_N}, y_N)\}$ with class labels $y_i \in Y$. Then computation of the centroid for each class can occur,

$$\vec{c_l} = \frac{1}{C_l} \sum_{i \in C_l} \vec{x_i} \tag{2.7}$$

where C_l is the set of instances of the training samples belonging to class $l \in Y$.

The classification function can be defined as:

$$\widehat{y} = \arg\min_{l \in Y} dist(\vec{c_l}, \vec{x}) \tag{2.8}$$

Though the K-nearest centroid classification method has its advantages, it has been designed for numerical data and it require computing the mean of several instances. One of the following chapters will discuss how to manipulate the numerical data based K-centroid method into the coupled similarity based method and apply it to the student performance prediction tasks.

In spite of the improvement of the classification efficiency, the classifier itself is still simple and naive, not using the state of art classification method. Hence in the next section the modern classification method will be discussed and also variations in order to utilize the coupled similarity in the student performance prediction task. It significantly enhances the classification accuracy by using the kernel method and the max-margin optimization process.

Support Vector Machines

Support vector machines (SVMs) are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by an apparent gap that is as wide as possible. New examples are then mapped into that same space and predicted as belonging to a category based on which side of the gap they fall.

In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

More formally, a support vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training data point of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier.

Whereas the original problem may be stated in a finite dimensional space, it often happens that the sets that must be classified are not linearly separable in that space. For this reason, it is proposed that the original finite-dimensional space be mapped into a much higher-dimensional space, presumably making the separation easier in that space. To keep the computational load reasonable, the mappings used by SVM schemes are designed to ensure that dot products may be computed easily in terms of the variables in the original space, by defining them in terms of a kernel function $k(x, y)$ selected to suit the problem. The hyper-planes in the higher-dimensional space are defined as the set of points whose dot product with a vector in that space is constant. The vectors defining the hyper-planes can be chosen to be linear combinations with parameters α_i of images of feature vectors that occur in the data base. With this choice of a hyperplane, the points x in the feature space that are mapped into the hyperplane are defined by the relation:

$$\sum_i \alpha_i k(x_i, x) = constant \tag{2.9}$$

Note if $k(x, y)$ becomes small as y grows further away from x, each term in the sum measures the degree of closeness of the test point x to the corresponding data base point x_i. In this way, the sum of kernels above can be used to measure the relative nearness of each test point to the data points originating in one or the other of the sets to be discriminated. Note the fact that the set of points x mapped into any hyperplane can be quite convoluted as a result, allowing much more complex discrimination between sets which are not convex at all in the original space. This strategy has the potential to ensure that the kernel function can be adapt to the varying the character of the data set.

Linear SVM

Given some training data $\mathcal{D}$, a set of n points of the form

$$\mathcal{D} = \{(\mathbf{x}_i, y_i) \mid \mathbf{x}_i \in \mathbb{R}^p,\, y_i \in \{-1, 1\}\}_{i=1}^{n} \tag{2.10}$$

where the y_i is either 1 or -1, indicating the class to which the point $\mathbf{x}_i$ belongs. Each $\mathbf{x}_i$ is a p-dimensional real vector. It is need to be found the maximum-margin hyperplane that divides the points having $y_i = 1$ from those having $y_i = -1$. Any hyperplane can be written as the set of points $\mathbf{x}$ satisfying $\mathbf{w} \cdot \mathbf{x} - b = 0$

where $\cdot$ denotes the dot product and $\mathbf{w}$ the normal vector to the hyperplane. The parameter $\frac{b}{\|\mathbf{w}\|}$ determines the offset of the hyperplane from the origin along the normal vector $\mathbf{w}$.

If the training data are linearly separable, it can be selected two hyperplanes in a way that they separate the data and there are no points between them, and then try to maximize their distance. The region bounded by them is called ”the margin”. These hyper-planes can be described by the equations $\mathbf{w} \cdot \mathbf{x} - b = 1$,and $\mathbf{w} \cdot \mathbf{x} - b = -1$.

Geometrically, the distance between these two hyper-planes is $\frac{2}{\|\mathbf{w}\|}$, so $\|\mathbf{w}\|$ needs to be minimized. As data points have to be prevented from falling into the margin, the following constraints were added: for each i either $\mathbf{w} \cdot \mathbf{x}_i - b \geq 1 \qquad$ for $\mathbf{x}_i$ of the first class or $\mathbf{w} \cdot \mathbf{x}_i - b \leq -1 \qquad$ for $\mathbf{x}_i$ of the second. This can be rewritten as: $y_i(\mathbf{w} \cdot \mathbf{x}_i - b) \geq 1, \quad$ for all $1 \leq i \leq n$. This is put together to get the optimization problem: Minimize (in $\mathbf{w}, b$) $\|\mathbf{w}\|$ subject to (for any $i = 1, \dots, n$) $y_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1$.

Primal Form

The optimization problem presented in the preceding section is difficult to solve because it depends on $\|\mathbf{w}\|$, the norm of $\mathbf{w}$, which involves a square root. Fortunately it is possible to alter the equation by substituting $\|\mathbf{w}\|$ with $\frac{1}{2}\|\mathbf{w}\|^2$ without changing the solution . This is a quadratic programming optimization problem. More clearly:

$\arg\min_{(\mathbf{w},b)} \frac{1}{2}\|\mathbf{w}\|^2$

subject to (for any $i = 1, \dots, n$)

$y_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1$.

By introducing Lagrange multipliers $\boldsymbol{\alpha}$, the previous constrained problem can be expressed as

$$\arg\min_{\mathbf{w},b} \max_{\boldsymbol{\alpha} \geq 0} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n} \alpha_i [y_i(\mathbf{w} \cdot \mathbf{x_i} - b) - 1] \right\} \tag{2.11}$$

looking for a saddle point. In doing so all the points which can be separated as $y_i(\mathbf{w} \cdot \mathbf{x_i} - b) - 1 > 0$ are insignificant as the corresponding α_i is set to zero.

This problem can now be solved by standard quadratic programming techniques and programs. The ”stationary” KarushCKuhnCTucker condition implies that the solution can be expressed as a linear combination of the training vectors

$\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x_i}$. Only a few α_i will be greater than zero. The corresponding $\mathbf{x_i}$ are exactly the support vectors, which lie on the margin and satisfy $y_i(\mathbf{w} \cdot \mathbf{x_i} - b) = 1$. From this it can be derived that the support vectors also satisfy

$\mathbf{w} \cdot \mathbf{x_i} - b = \frac{1}{y_i} = y_i \iff b = \mathbf{w} \cdot \mathbf{x_i} - y_i$ which allows the definition of offset b. The b depends on y_i and x_i, so it will vary for each data point in the sample. In practice, it is more robust than average over all N_{SV} support vectors, since the average over the sample is an unbiased estimator of the population mean:

$b = \frac{1}{N_{SV}} \sum_{i=1}^{N_{SV}} (\mathbf{w} \cdot \mathbf{x_i} - y_i)$

Dual Form

Writing the classification rule in its unconstrained dual form reveals that the maximum-margin hyperplane and therefore the classification task is only a function of the support vectors, which are the subset of the training data that lie on the margin.

Using the fact that $\|\mathbf{w}\|^2 = \mathbf{w} \cdot \mathbf{w}$ and substituting $\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x_i}$, one can show that the dual of the SVM reduces to the following optimization problem:

Maximize (in α_i)

$$\tilde{L}(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)$$

subject to (for any $i = 1, \ldots, n$)

$$\alpha_i \geq 0$$

and to the constraint from the minimization in b

$$\textstyle\sum_{i=1}^{n} \alpha_i y_i = 0$$

Here the kernel is defined by $k(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j$.

W can be computed thanks to the α terms:

$$\mathbf{w} = \textstyle\sum_i \alpha_i y_i \mathbf{x}_i$$

Soft margin

If there exists no hyperplane that can split the "yes" and "no" examples, the Soft Margin method will choose a hyperplane that splits the examples as cleanly as possible, while still maximizing the distance to the nearest cleanly split examples. The method introduces non-negative slack variables, ξ_i, which measure the degree of misclassification of the data x_i

$$y_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1 - \xi_i \quad 1 \leq i \leq n.$$

The objective function is then increased by a function which penalizes non-zero ξ_i, and the optimization becomes a trade off between a large margin and a small error penalty. If the penalty function is linear, the optimization problem becomes:

$$\arg\min_{\mathbf{w},\xi,b} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i \right\}$$

subject to (for any $i = 1, \dots n$)

$$y_i(\mathbf{w} \cdot \mathbf{x_i} - b) \geq 1 - \xi_i, \qquad \xi_i \geq 0$$

Using the hinge function notation like that in MARS, this optimization problem can be rewritten as $\sum_i [1 - y_i(w \cdot x_i + b)]_+ + \lambda\|w\|^2$, wherein let $[1 - y_i(w \cdot x_i + b)]_+ = [\xi_i]_+ = \xi_i, \quad \lambda = 1/2C$.

This constraint along with the objective of minimizing $\|\mathbf{w}\|$ can be solved using Lagrange multipliers as done above. One then has to solve the following problem:

$$\arg\min_{\mathbf{w},\xi,b} \max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i [y_i(\mathbf{w} \cdot \mathbf{x_i} - b) - 1 + \xi_i] - \sum_{i=1}^{n} \beta_i \xi_i \right\}$$

with $\alpha_i, \beta_i \geq 0$.

Maximize (in α_i)

$$\tilde{L}(\alpha) = \textstyle\sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j)$$

subject to (for any $i = 1, \ldots, n$)

$$0 \leq \alpha_i \leq C,$$

and

$$\textstyle\sum_{i=1}^{n} \alpha_i y_i = 0.$$

The key advantage of a linear penalty function is that the slack variables vanish from the dual problem, with the constant C appearing only as an additional constraint on the Lagrange multipliers. For the above formulation and its huge impact in practice, Nonlinear penalty functions have been used, particularly to reduce the effect of outliers on the classifier, but unless care is taken the problem becomes non-convex, and thus it is considerably more difficult to find a global solution.

The Support Vector Machines are a sophisticated solution for the classification and regression task. However, integrating the coupled similarity with the SVM is a vital problem because the x_i in our training data sets are not numerical features, it cannot use the function directly. Though it can be defined the coupled similarity as the kernel function to prevent the computation of the x_i to others, the w is still reliant upon the x_i and y_i. Hence, it is still hard to implement the coupled similarity into traditional SVM. Fortunately, (Brunner, Fischer, Luig & Thies 2012) proposed the pairwise support vector machines that suitable for our aim. A later chapter will introduce the Pairwise Support Vector Machines to tackle this problem.

2.1.3 Text Mining

This section will discuss the basic foundation of text mining. These methods will be used in the later chapter for the student social network sentiment analysis. The basic text mining concept and TF-IDF definition will be discussed in the following.

Text Mining Concept

Text mining (also referred to as text data mining, roughly equivalent to text analytics), refers to the process of deriving high-quality information from text. High-quality information is typically obtained through the devising of patterns and trends through means such as statistical pattern learning. Text mining usually involves the process of structuring the input text, deriving patterns within the structured data, and finally evaluation and interpretation of the output. High quality in text mining usually refers to some combination of relevance, novelty, and intrigue. Typical text mining tasks include text categorization, text clustering, concept or entity extraction, production of granular taxonomies, sentiment analysis, document summarization, and entity-relation modeling.

Text analysis involves information retrieval, lexical analysis to study word frequency distributions, pattern recognition, tagging/annotation, information extraction, data mining techniques including link and association analysis, visualization, and predictive analytics. The overarching goal is, essentially, to turn text into data for analysis, via application of natural language processing (NLP) and analytical methods.

A typical application is to scan a set of documents written in a natural language and either model the document set for predictive classification purposes or populate a database or search index with the information extracted.

TF-IDF

TFCIDF, short for term frequency inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining. The TFCIDF value increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus, which helps to adjust for the fact that some words appear more frequently in general.

Variations of the TFCIDF weighting scheme are often used by search engines as a central tool in scoring and ranking a document's relevance given a user query. TFCIDF can be successfully used for stop-words filtering in various subject fields including text summarization and classification.

One of the simplest ranking functions is computed by summing the TF-CIDF for each query term; many more sophisticated ranking functions are variants of this simple model.

Term frequencyCInverse document frequency TFCIDF is calculated as

$$\mathrm{tfidf}(t, d, D) = \mathrm{tf}(t, d) \times \mathrm{idf}(t, D)$$

Vector Space Model

Vector space model or term vector model is an algebraic model for representing text documents as vectors of identifiers, such as, for example, index terms. They are used in information filtering, information retrieval, indexing and relevancy rankings.

Definitions

Documents and queries are represented as vectors.

$$d_j = (w_{1,j}, w_{2,j}, \ldots, w_{t,j}) \; q = (w_{1,q}, w_{2,q}, \ldots, w_{n,q})$$

Each dimension corresponds to a separate term. If a term occurs in the document, its value in the vector is non-zero. Several different ways of computing these values, also known as (term) weights, have been developed. One of the best known schemes is TF-IDF weighting

The definition of term depends on the application. Typically terms are single words, keywords, or longer phrases. If words are chosen to be the terms, the dimensionality of the vector is the number of words in the vocabulary (the number of distinct words occurring in the corpus).

Vector operations can be used to compare documents with queries.

Applications

Relevance rankings of documents in a keyword search can be calculated, using the assumptions of document similarities theory, by comparing the deviation of angles between each document vector and the original query vector where the query is represented as the same kind of vector as the documents.

In practice, it is easier to calculate the cosine of the angle between the vectors, instead of the angle itself:

$$\cos\theta = \frac{\mathbf{d_2} \cdot \mathbf{q}}{\|\mathbf{d_2}\| \, \|\mathbf{q}\|} \tag{2.12}$$

Where $\mathbf{d_2} \cdot \mathbf{q}$ is the intersection (i.e. the dot product) of the document and the query vectors, $\|\mathbf{d_2}\|$ is the norm of vector d_2, and $\|\mathbf{q}\|$ is the norm of vector q. The norm of a vector is calculated as such:

$$\|\mathbf{q}\| = \sqrt{\sum_{i=1}^{n} q_i^2} \tag{2.13}$$

As all vectors under consideration by this model are elementwise nonnegative, a cosine value of zero means that the query and document vector are orthogonal and have no match (i.e. the query term does not exist in the document being considered).

The term-specific weights in the document vectors are products of local and global parameters. This model is known as the term frequency-inverse document frequency model. The weight vector for document d is $\mathbf{v}_d = [w_{1,d}, w_{2,d}, \ldots, w_{N,d}]^T$, where

$$w_{t,d} = \mathrm{tf}_{t,d} \cdot \log \tfrac{|D|}{|\{d' \in D \,|\, t \in d'\}|}$$

and $\mathrm{tf}_{t,d}$ is term frequency of term t in document d $\log \tfrac{|D|}{|\{d' \in D \,|\, t \in d'\}|}$ is inverse document frequency. —D— is the total number of documents in the document set; $|\{d' \in D \,|\, t \in d'\}|$ is the number of documents containing the term t.

Using the cosine the similarity between document dj and query q can be calculated as:

$$\mathrm{sim}(d_j, q) = \frac{\mathbf{d_j} \cdot \mathbf{q}}{\|\mathbf{d_j}\| \, \|\mathbf{q}\|} = \frac{\sum_{i=1}^{N} w_{i,j} w_{i,q}}{\sqrt{\sum_{i=1}^{N} w_{i,j}^2} \sqrt{\sum_{i=1}^{N} w_{i,q}^2}} \tag{2.14}$$

2.2 Literature Review of Education Data Mining

In this section will discuss the related work of the Educational data mining, and whilst its the main contribution of this thesis is focused on the student performance prediction and the student sentiment analysis, the efficacy of research into the whole educational data mining domain is also critical. A comprehensive understanding of educational data mining paradigms result in more specific research. Educational data mining is an emerging research area. It develops methods that explore the value of the data in an educational context.Educational data mining uses computational approaches to analyze educational data in order to answer some critical educational questions. This section summarises the most relevant research and study in this field to date.

Educational data mining is the application of data mining techniques to deal with the educational data. The objective is to analyze these types of data to reveal the hidden value of the educational data and to enhance the education concepts and methods. Data mining can be defined as the process involved in extracting interesting, interpretable, useful and novel patterns or information from data (Fayyad, Piatetsky-Shapiro & Smyth 1996). It has been applied for many years in business, science and government policy. It sifts through volumes of data like airline passenger records, census data and the supermarket scanner data to produce market research reports (Jiawei & Kamber 2001). However, the educational data has not been extensively investigated. Educational data mining focuses on developing methods to explore the unique types of data in educational settings. By using these methods, a better understanding of students and the environments in which they learn (Baker et al. 2010) can be obtained. Furthermore, the increase in both instrumental educational software and state databases of student information have been created, and there are large repositories of data now reflecting how students learn (Koedinger, Cunningham, Skogsholm & Leber 2008). Concurrently, the use of the internet in education has created a new context known as e-learning or web-based education which also generates a large amount of information about teaching-learning interaction (Castro, Vellido, Nebot & Mugica 2007). All the information provides a gold mine of educational data (Mostow & Beck 2006). Educational data mining seeks to use these data repositories to get a better understanding of learners and learning, and to develop novel approaches that utilise data mining theory as a practice that can benefit students. Educational data mining has emerged as a research area in recent years for researchers all over the world. There are different and related research areas such as:

OffLine Education

This category of Educational data mining tries to transmit knowledge and skills based on face-to-face contact and also study psychologically on how humans learn. Psychometric and statistical techniques have been applied to data from student behavior performance, curriculum, etc. that was gathered in classroom environments

E-Learning

This category of Educational data mining also called as Learning Management System (LMS). E-learning provides online instruction and LMS also provides communication, collaboration, administration and reporting tools. Web Mining (WM) techniques have been applied to student data stored by these systems in log files and databases.

Intelligent Tutoring

Intelligent Tutoring System(ITS) and Adaptive Educational Hypermedia System (AEHS) are an alternative to the just-put-it-on-the-web approach by trying to adapt teaching to the needs of each particular student. Data Mining has been applied to data picked up by these systems, such as log files, user models, etc.

The Educational data mining process converts raw data coming from educational systems into useful information that could potentially have an enormous impact on educational research and practice. This process does not differ much from other application areas of data mining like business, genetics, medicine, etc. Because it follows the same steps as the general data mining process: pre-processing, data mining and post-processing. However, in this paper the term data mining is used in a larger sense than the original traditional Data mining definition. That is, and this thesis is going to describe not only Educational data mining studies that use typical Data mining techniques such as classification, clustering, association rule mining, sequential mining, text mining, etc. but also other approaches such as regression, correlation, visualization, etc. that are not considered to be Data mining in a strict sense. Furthermore, some methodological innovations and trends in Educational data mining such as discovery with models and the integration of psychometric modeling frameworks are unusual Data mining categories or not necessarily universally seen as being DM. From a practical viewpoint Educational data mining allows, for example, to discover new knowledge based on students usage data in order to help validate evaluate educational systems. To potentially improve some aspects of the quality of education and to lay the groundwork for a more efficient learning process (Romero, Ventura & De Bra 2004). Some similar ideas were already successfully applied in e-commerce systems, the first and most modern application of data mining, in order to determine clients interests so as to be able to increase online sales. However, there has been comparatively less progress in this direction in Education to date, although this situation is changing, and there is currently an increasing interest in applying data mining to the educational environment. Even so, there are some important issues that differentiate the application of Data Mining specifically to education from how it is applied in other domains:

Objective

The objective of data mining in each application area is different. For example, in business the primary purpose is to increase profit, which is tangible and can be measured in term of amounts of money, number of customers and customer loyalty. But Educational data mining has both applied research objectives, such as improving the learning process and guiding students learning; as well as pure research goals, such as achieving a deeper understanding of educational phenomena. These goals are sometimes difficult to quantify and require their unique set of measurement techniques.

Data

In educational environments, there are many different types of data available for mining. These data are unique to the educational area, and so it have intrinsic semantic information, relationships with other data and multiple levels of meaningful hierarchy. Some examples are the domain model, used in ITS and AEHS which represents the relationships among the concepts of a particular subject in a graph or hierarchy format. For another instance, the q-matrix that shows relationships between items questions of a test quiz system and the concepts evaluated by the test. Furthermore, it is also necessary to take pedagogical aspects of the learner and the system into account.

Techniques

Educational data and problems have some particular characteristics that require the issue of mining to be treated in a different way. Although most of the traditional Data mining techniques can be applied directly, others cannot and have to be adapted to the specific educational problem at hand. Furthermore, specific data mining techniques can be used for specific educational problems.

2.2.1 Student Performance Prediction

A comparison of machine learning methods has been carried out to predict success in a course in Intelligent Tutoring Systems (Hämäläinen & Vinni 2006). Other comparisons of different data mining algorithms are made to classify students based on Moodle usage data (Romero, Ventura & García 2008); to predict student performance based on features extracted from logged data (Minaei-Bidgoli, Kashy, Kortemeyer & Punch 2003) and to predict University students academic performance (Ibrahim & Rusli 2007).

Different types of neural network models have been used to predict final student grades (using back-propagation and feedforward neural networks) (Gedeon & Turner 1993); to predict the number of errors a student will make (using feed-forward and backpropagation)(Wang 2002); to predict performance from test scores (using backpropagation and counter-propagation) (Fausett & Elwasif 1994); to predict students marks (pass or fail) from Moodle logs (using radial basis functions) (Calvo-Flores, Galindo, Jiménez & Pineiro 2006) and for predicting the likely performance of a candidate being considered for admission into the university (using multilayer perceptron topology) (Oladokun, Adebanjo & Charles-Owaba 2008).

Bayesian networks have been used to predict student applicant performance (Hien & Haddawy 2007); to model user knowledge and predict student performance within a tutoring system (Pardos, Heffernan, Anderson & Heffernan 2007); to predict a future graduates cumulative Grade Point Average based on applicant background at the time of admission (Hien & Haddawy 2007); to model two different approaches to determine the probability a multi skill question has of being corrected (Pardos, Beck, Ruiz & Heffernan 2008) and to predict future group performance in face-to-face collaborative learning (Stevens, Soller, Giordani, Gerosa, Cooper & Cox 2005); to predict end-of-year exam performance through student activity with online tutors (Ayers & Junker 2006) and to predict item response outcome (Desmarais & Gagnon 2006).

Different types of rule-based systems have been applied to predict student performance (mark prediction) in an e-learning environment (using fuzzy association rules) (Nebot, Castro, Vellido & Mugica 2006); to predict learner performance based on the learning portfolios compiled (using key formative assessment rules) (Lara, Lizcano, Martínez, Pazos & Riera 2014); for prediction, monitoring and evaluation of student academic performance (using rule induction) (Ogor 2007); to predict final grades based on features extracted from logged data in an education web-based system (using genetic algorithm to find association rules) (Shangping & Ping 2008); to predict student grades in LMSs (using grammar guided genetic programming) (Zafra

& Ventura 2009); to predict student performance and provide timely lessons in web-based e-learning systems (using decision tree) (Chan 2007); to predict online students marks (using an orthogonal search-based rule extraction algorithm) (Etchells, Nebot, Vellido, Lisboa & Mugica 2006).

Several regression techniques have been used to predict students marks in an open university (using model trees, neural networks, linear regression, locally weighted linear regression and support vector machines) (Kotsiantis & Pintelas 2005); for predicting end-of-year accountability assessment scores (using linear regression prediction models) (Anozie & Junker 2006); to predict student performance from log and test scores in web-based instruction (using a multi-variable regression model); for predicting student academic performance (using stepwise linear regression) (Golding & Donaldson 2006); for predicting time to be spent on a learning page (using multiple linear regression) (Arnold, Scheines, Beck & Jerome 2005); for identifying variables that could predict success in colleges courses (using multiple regression) (Martinez 2001); for predicting university students satisfaction (using regression and decision trees analysis) (Thomas & Galambos 2004); for predicting exam results in distance education courses (using linear regression) (Myller, Suhonen & Sutinen 2002); for predicting when a student will get a question correct and association rules to guide a search process to find transfer models to predict a students success (using logistic regression) (Freyberger, Heffernan & Ruiz 2004); to predict the probability a student has of giving the correct answer to a problem in an ITS (using a robust Ridge regression algorithm) (Cetintas, Si, Xin & Hord 2009); for predicting end-of-year accountability assessment scores (using linear regression) (Anozie & Junker 2006), to predict a students test score (using stepwise regression) (Feng, Heffernan & Koedinger 2005) and to predict the probability that the students next response has of being correct (using linear regression) (Beck & Woolf 2000).

Finally, correlation analyses have been applied together to predict web-student performance in on-line classes; to predict a students final exam score in online tutoring (Pritchard & Warnakulasooriya 2005) and for predicting high school students probabilities of success in university (Campbell & Dickson 1996).

2.2.2 Sentiment Analysis

The earliest researchers dealing with sentiment analysis consisted of classifying words or phrases according to semantic issues and date from the late 1990s (Hatzivassiloglou & McKeown 1997). Linguistic heuristics or preselected sets of seed words were used. The results obtained in those works served as the basis for classifying entire documents, considering that the average semantic orientation of the words in a review may be an indicator of whether the text is positive or negative (Turney 2002). The appearance of WordNet (Miller 1995) and, in general, of annotated corpora, increased the production in this research area. On one hand, WordNet is useful because it allows knowing the semantic relationships between different words. Therefore, with a reduced set of polarity words, every word could be labeled as positive, negative or neutral through its relationships. On the other hand, corpora and, in particular, the Treebanks, are very useful. They are corpora with the syntactic structure labeled, and are of great help for training the analyzers in order to label the words automatically.

One of the first works that used the term ”sentiment analysis” as it be known currently, it was that presented in (Das & Chen 2007), which analyzes messages written in stock boards in order to extract the market sentiment. Currently, many of the works in this area focus on document classification based on the sentiment expressed on it. One of the best known domains is that of reviews (Dave, Lawrence & Pennock 2003). Review websites are examples of especially useful sources for sentiment analysis. Other application areas in which sentiment analysis can be very useful are: recommendation systems, flame detection, sensitive content detection for advertising, humanCcomputer interaction, business intelligence , prediction of hostile or negative sources, classification of citizens opinions on a law, broadcasting based on the receiver sentiment, dynamic adaptation of daily tools, such as e-mail, marketing or politics.

In general, accuracy is strongly influenced by the context in which the words are used (Turney 2002). For instance, the sentence "You must read the book" is positive in a book review but is negative if the review is about films.

Additionally, the position of words in text is an interesting factor to consider, since a word at the end of a sentence can change the polarity completely (Pang & Lee 2008). For example the sentence "This book is very addictive", it can be read in one sitting, but I have to admit that it is rubbish begins with the word "addictive" and the expression "one sitting", which are positive in the context of book reviews, but it finish with the word "rubbish" that is negative. Although the sentence contains two positive tokens against one negative, it should be marked as negative because the final word nullifies all the previous ones.

Another issue to be considered is the presence of figures of speech in the analyzed text. Some of them, such as the irony, can change the whole polarity of a text. Sometimes they are difficult to detect even for a human being, if additional information is not provided. Recent work in natural language processing focuses on the detection of these figures, such as (Reyes, Rosso & Buscaldi 2012), that build a training dataset of messages written in Twitter with the hash tag "#irony" in order to set a model with machine-learning techniques.

With respect to the techniques used for sentiment analysis, two main approaches are considered: machine-learning methods and lexicon-based approach. The survey written by Pang and Lee (Pang & Lee 2008) covers the most popular techniques and approaches.

On one hand, machine-learning methods are used to classify texts. An example of the use of machine-learning techniques in order to classify movie reviews is presented in (Pang, Lee & Vaithyanathan 2002). It compares different techniques to classify movie reviews, obtaining 82.9% of accuracy when applying Support Vector Machines (SVM). Generally, it is difficult to obtain better results, due to characteristics of natural language. However, in specific domains, the use of machine learning algorithms for classifying texts according to their sentiment orientation performs well.

On the other hand, the lexicon-based approach consists of analyzing the text grammar and executing a function to give a sentiment score to the text, considering a predefined sentiment lexicon (Turney 2002) (Taboada, Brooke, Tofiloski, Voll & Stede 2011). There exist some sentiment lexicon available, such as SentiWordNet (Esuli & Sebastiani 2006), but it has been noticed that most of the researches build their own lexicon ad hoc, managing the semantic relationships between words with tools such as WordNet (Miller 1995), already mentioned above. The great advantage of the lexicon-based approach is that it is not necessary to have a labeled training set to start classifying texts. This approach tends to get worse results than machine learning approaches in specific domains, but when the domain is less bounded the results are better. This is because the lexicon approach analyzes the text grammar, whereas the machine-learning methods fit the algorithms to the training dataset particularities. As an example, in (Taboada et al. 2011) the authors use a lexicon-based method with six different corpora from different domains and obtained 80% accuracy. However, when using machine learning (with a preprocessing phase to summarize movie reviews), 86.4% accuracy was achieved (Pang & Lee 2004). When comparing these two works, the machine-learning approach gets a better accuracy, but it may suffer overfitting to the training dataset, whereas the lexicon-based approach gets a lower accuracy, although is more robust when considering different domains.

2.2.3 Segmentation of Different Users

Educational data mining involves different groups of users or participants. Various groups look at educational information from different angles according to their mission, vision and objectives for using data mining (Hanna 2004). For example, knowledge discovered by Educational data mining algorithms can be used?to help teachers to manage their classes. To understand their students learning processes and reflect on their teaching methods. However, also to support a learners reflections on the situation and provide feedback to learners (Merceron & Yacef 2005). Although an initial consideration seems to involve only two main groups, the students and the instructors, there are more groups involved with many more objectives.

Learners, Students, Pupils

To personalize e-learning; to recommend activities to learners and resources and learning tasks that could further improve their learning; to suggest exciting learning experiences to the students; to suggest path pruning and shortening or only links to follow, to generate adaptive hints, to recommend courses, relevant discussions, books, etc.

Educators, Teachers, Instructors, Tutors

To get objective feedback about instruction; to analyze students learning and behavior; to detect which students require support; to predict student performance; to classify learners into groups; to find a learners regular as well as irregular patterns; to find the most frequently made mistakes; to determine more effective activities; to improve the adaptation and customization of courses, etc.

Course Developers, Educational Researchers

To evaluate and maintain courseware; to improve student learning; to evaluate the structure of course content and its effectiveness in the learning process; to automatically construct student models and tutor models; to compare data mining techniques in order to be able to recommend the most useful one for each task; to develop specific data mining tools for educational purposes; etc.

Organizations, Learning Providers, Universities, Private Training Companies

To enhance the decision processes in higher learning institutions; to streamline efficiency in the decision-making process; to achieve specific objectives; to suggest certain courses that might be valuable for each class of learners; to find the most cost-effective way of improving retention and grades; to select the most qualified applicants for graduation; to help to admit students who will do well in university, etc.

Administrators, School District Administrators, Network Administrators, System Administrators

To develop the best way to organize institutional resources (human and material) and their educational offer; to utilize available resources more effectively; to enhance educational program offers and determine the effectiveness of the distance learning approach; to evaluate teacher and curricula; to set parameters for improving web-site efficiency and adapting it to users (optimal server size, network traffic distribution, etc.).

2.2.4 Diversity of Scenarios

Nowadays, there is a great variety of educational systems environments such as: the traditional classroom, e-learning, LMS, AH educational systems, ITS, tests quizzes, texts contents, and others such as: learning object repositories, concept maps, social networks, forums, educational game environments, virtual environments, ubiquitous computing environments, etc. All data provided by each of the aforementioned educational environments are different, thus enabling different problems and tasks to be resolved using data mining techniques. The following content list of the most important studies on Educational data mining grouped according to the type of data environment involved.

Traditional Education

(Beck, Baker, Corbett, Kay, Litman, Mitrovic & Ritter 2004) analyzed student-tutor interaction logs to improve educational outcomes, (Hernándeza, Ochoab, Muñozd & Burlaka 2006) detected cheats in online student assessments using Data Mining, (Dekker, Pechenizkiy & Vleeshouwers 2009) predicted students drop out, (Beikzadeh, Phon-Amnuaisuk & Delavari 2008) developed a data mining application in higher learning institutions, (Fausett

& Elwasif 1994) predicted performance from test scores using backpropagation and counterpropagation, (Gedeon & Turner 1993) explained student grades predicted by a neural network,(Golding & Donaldson 2006) predicted academic performance, (Hien & Haddawy 2007) developed a decision support system for evaluating international student applications, (Hien & Haddawy 2007) developed a decision support system for evaluating international student applications, (Hsia, Shie & Chen 2008) developed a course planning system of extension education to meet market demand by using data mining techniques, (Huang, Lin, Wang & Wang 2009) tried on planning of educational training courses by data mining, (Ibrahim & Rusli 2007) compared the artificial neural network, decision tree and linear regression performance on predicting student performance, (Jin, Wu, Liu & Yan 2009) developed an application of visual data mining in higher-education evaluation system, (Khajuria 2007) designed a model to predict student matriculation from admissions data, (Kiang, Fisher, Chen, Fisher & Chi 2009) developed an application of SOM as a decision support tool tdo identify AACSB peer schools, (Kotsiantis, Pierrakeas & Pintelas 2003) worked on preventing student dropout in distance learning using machine learning techniques, (Kotsiantis & Pintelas 2005) has done the research on predicting students marks in hellenic open university, (Ma, Liu, Wong, Yu & Lee 2000) explained how to target the right students using data mining, (Martinez 2001) predicted student outcomes using discriminant function analysis, (Campbell & Dickson 1996) predicted student success by using integrative review and meta-analysis, (Ogor 2007) developed a system monitoring and evaluation on student academic performance using data mining techniques, (Oladokun et al. 2008) predicted students academic performance using artificial neural network, (Quevedo & Montañés 2009) obtained rubric weights for assessments by more than one lecturer using a pairwise learning model, (Ranjan & Khalil 2008) explored a conceptual framework of data mining process in management education in India, (Sanjeev & Zytkow 1995) discovered enrollment knowledge in university databases, (Schönbrunn & Hilbert 2007) investigate on data mining in higher education, (Selmoune & Alimazighi 2008) developed a decisional tool for quality improvement in higher education, (Superby, Vandamme & Meskens 2006), (Thomas & Galambos 2004), (Tsantis & Castellani 2001), (Vialardi, Bravo, Shafti & Ortigosa 2009), (Vranic, Pintar & Skocir 2007) worked on the use of data mining in education environment, (Wang, Cheng, Chang & Jen 2008) produced an application of data mining technique and genetic algorithm to an automatic course scheduling system, (Yu, Digangi, Jannasch-Pennell & Kaprolet 2008) profiled students who take online courses using data mining methods, (Zukhri & Omar 2008) solved the problem of new student allocation problem with genetic algorithm.

Web-based Education E-learning

(Avouris, Komis, Fiotakis, Margaritis & Voyiatzaki 2005) worked on logging of fingertip actions is not enough for analysis of learning activities, (Chan 2007) developed a framework for assessing usage of web-based E-Learning systems, (Chen, Liu, Ou, Liu et al. 2000) discovered the decision knowledge from web log portfolio for managing classroom processes by applying decision tree and data cube technology, (Chen, Duh & Liu 2004) developed a personalized course ware recommendation system based on fuzzy item response theory, (Cocea & Weibelzahl 2007) has done the research on the cross-system validation of engagement prediction from log files, (Fok, Wong & Chen 2005) developed a hidden markov model based characterization of content access patterns in an E-Learning environment, (García, Amandi, Schiaffino & Campo 2007) evaluated Bayesian networks precision for detecting students learning styles,(Grob, Bensberg & Kaderali 2004) controlled open source intermediaries-a web log mining approach, (Ha, Bae & Park 2000) investigated the web mining for distance education, (Hadwin, Nesbit, Jamieson-Noel, Code & Winne 2007) explored examining trace data to explore self-regulated learning, (Hershkovitz & Nachmias 2009) explained on the consistency of students' pace in online learning, (Huang, Zhu & Luo 2007) developed a personality mining method in web based education system using data mining, (Hwang, Tsai, Tsai & Tseng 2008) introduced a novel approach for assisting teachers in analyzing student web-searching behaviors, (Ingram 2000) used web server logs in evaluating instructional web sites, (Kosba, Dimitrova & Boyle 2005) used student and group models to support teachers in web-based distance education, (Khribi, Jemni & Nasraoui 2008) developed a system that automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval, (Lee, Chen & Liu 2007) worked on mining learners behavior in accessing web-based interface, (Lee, Chen, Chrysostomou & Liu 2009) researched on mining students behavior in web-based learning programs, (Lemire, Boley, McGrath & Ball 2005) produced a system by using collaborative filtering and inference rules for context-aware learning object recommendation, (Licchelli, Basile, Di Mauro, Esposito, Semeraro & Ferilli 2004) introduced a machine learning approaches for inducing student models, (Lykourentzou, Giannoukos, Nikolopoulos, Mpardis & Loumos 2009) worked on dropout prediction in e-learning courses through the combination of machine learning techniques, (Li, Luo & Yuan 2007) developed personalized recommendation service system in e-learning using web intelligence, (Merceron, Oliveira, Scholl & Ullrich 2004) used the data mining for content re-use and exchange-Solutions and problems, (Cristian & Dan 2006) tested attribute selection algorithms for classification performance on real data, (Minaei-Bidgoli et al. 2003) developed an application of data mining methods with an educational web-based system, (Minaei-Bidgoli, Tan & Punch 2004) introduced mining interesting contrast rules for a web-based educational system, (Myller et al. 2002) used data mining for improving web-based course design, (Nebot et al. 2006) identified of fuzzy models to predict students performance in an e-learning environment, (Orzechowski, Ernst & Dziech 2007) profiled Search Methods for e-Learning Systems, (Pahl & Donnellan 2002) explored the Data mining technology for the evaluation of web-based teaching and learning systems, (Rahkila & Karjalainen 1999) evaluated of learning in computer based education using log systems, (Ramli 2005) worked on a web usage mining using apriori algorithm, (Romero, Santos, Freire & Ventura 2008) introduced a mining and visualizing visited trails in Web-Based educational systems, (Shangping & Ping 2008) developed a data mining algorithm in distance learning, (Singley & Lam 2005) explained the supporting data-driven decision-making in the classroom, (Wen-Shung Tai, Wu & Li 2008) developed an effective e-learning recommendation system based on self-organizing maps and association mining, (Tian, Wang, Zheng & Zheng 2008) researched on e-learner personality grouping based on fuzzy clustering analysis, (Ueno & Nagaoka 2002) Learned Log Database and Data Mining system for e-Learning, (Wang 2002) used data-mining technology for browsing log file analysis in asynchronous learning environment, (Wang & Shao 2004) introduced an effective personalized recommendation based on time-framed navigation clustering and association mining, (Yang, Lin & Wu 2002) produced an agent-based recommender system for lesson plan sequencing, (Yang, Han, Shen & Hu 2005) developed a novel resource recommendation system based on connecting to similar e-learners, (Yoo, Yoo, Lance & Hankins 2006) proposed a student progress monitoring tool using tree view, [290], (Yu, Own & Lin 2001) explored the learning behavior analysis of web based interactive environment, (Zaïane & Luo 2001) introduced a web usage mining for a better web-based learning environment, (Zaíane 2002) has built a recommender agent for e-learning systems, (Zhang, Cui, Wang & Sui 2007) worked out an improvement of matrix-based clustering method for grouping learners in e-learning, (Zhang, Liu & Liu 2008) developed a personalized instructing recommendation system based on web mining, (Zhu, Ip, Fok & Cao 2008) introduced a personalized recommendation education system based on multi-agents.

Learning Management Systems

(Baruque, Amaral, Barcellos, da Silva Freitas & Longo 2007) analysed users' access logs in Moodle to improve e learning, (Chanchary, Haque & Khalid 2008) proposed a web usage mining to evaluate the transfer of learning in a web-based learning environment, (Chang, Kao, Chu & Chiu 2009) explored a learning style classification mechanism for e-learning, (Castro, Vellido, Nebot & Minguillon 2005) detected atypical student behaviour on an e-learning system, (Calvo-Flores et al. 2006) predicted students marks from Moodle logs using neural network models, (Etchells et al. 2006) develop a method of feature selection and rule extraction in a virtual course,(Guo & Zhang 2009) implemented web learning environment based on data mining, (Heathcote & Prakash 2007) introduced a monitoring system information to improve support for teachers using educational technologies, (Heathcote & Dawson 2005) discussed Data mining for evaluation, benchmarking and reflective practice,(Jovanović, Gašević, Brooks, Devedžić & Hatala 2007) developed a tool for raising teachers awareness in online learning environments, (Liu & Shih 2007) proposed a learning activity-based e-learning material recommendation system, (dos Santos Machado & Becker 2003) introduced a web usage mining case study for the evaluation of learning sites,(Mazza & Milani 2004) proposed a graphical interactive student monitoring tool for course management systems,(Merceron & Yacef 2008) discussed the interestingness measures for associations rules in educational data., (Myszkowski, Kwasnicka & Markowska-Kaczmar 2008) explained data mining techniques in e-learning CelGrid system, (Monk 2005) used data mining for e-learning decision making,(Psaromiligkos, Orfanidou, Kytagias & Zafiri 2011) researched mining log data for the analysis of learners behaviour in web-based learning management systems, (Perera, Kay, Koprinska, Yacef & Zaïane 2009) explored clustering and sequential pattern mining of online collaborative learning data, (Romero, Ventura & García 2008) discussed data mining in course management systems, (Retalis, Papasalouros, Psaromiligkos, Siscos & Kargidis 2006) introduced a concept and a tool towards networked learning analytics, (Shen, Yang & Han 2002) discussed data analysis center based on e-learning platform, (Talavera & Gaudioso 2004) researched on mining student data to characterize similar behavior groups in unstructured collaboration spaces, (Vellido, Castro, Etchells, Nebot & Mugica 2007) explored data mining of virtual campus data,(Ventura, Romero & Hervás 2008) analyzed rule evaluation measures with educational data sets, (Wang 2008) proposed a content recommendation based on education-contextualized browsing events for web-based personalized learning, (Zafra & Ventura 2009) predicted student grades in learning management systems with multiple instance genetic programming, (Zorrilla, Menasalvas, Marin, Mora & Segovia 2005) developed a web usage mining project for improving web-based learning sites.

Intelligent Tutoring Systems

(Antunes 2008) acquired background knowledge for intelligent tutoring systems, (Baker, Corbett & Koedinger 2004) detected student misuse of intelligent tutoring systems, (Baker 2007) modeled the understanding students' off-task behavior in intelligent tutoring systems, (Baker, Corbett & Aleven 2008) improved contextual models of guessing and slipping with a truncated training set, (Barnes & Stamper 2008) developed an automatic hint generation for logic proof tutoring using historical student data, (Beal & Cohen 2008) introduced a temporal data mining technique for educational applications, (Beck & Woolf 2000) discussed high-level student modeling with machine learning, (Chang, Beck, Mostow & Corbett 2006) proposed a bayesnet toolkit for student modeling in intelligent tutoring systems, (Cetintas et al. 2009) predicted correctness of problem solving from Low-Level log data in intelligent tutoring systems., (Crespo, Pardo, Pérez & Kloos 2005) developed an algorithm for peer review matching using student profiles based on fuzzy classification and genetic algorithms,(Feng & Beck 2009) discussed a non-automated method of constructing transfer models, (Gong, Rai, Beck & Heffernan 2009) discussed whether self-discipline impact students' knowledge and learning, (Hämäläinen & Vinni 2006) described the comparison of machine learning methods for intelligent tutoring systems, (Heraud, France & Mille 2004) proposed an ITS that guides students with the help of learners' interaction log, (Hurley & Weibelzahl 2007) used MotSaRT to support on-line teachers in student motivation, (Jonsson, Johns, Mehranian, Arroyo, Woolf, Barto, Fisher & Mahadevan 2005) discussed the evaluating the feasibility of learning student models from data, (Koedinger et al. 2008) introduced an open repository and analysis tools for fine-grained, longitudinal learner data, (McLaren, Koedinger, Schneider, Harrer & Bollen 2004) developed a semi-automated tutor authoring using student log files, (Merceron & Yacef 2005) discussed a case study in educational data mining,(Mostow, Beck, Cen, Cuneo, Gouvea & Heiner 2005) proposed an educational data mining tool to browse tutor-student interactions, (Pardos et al. 2007) discussed a effect of model granularity on student performance prediction using Bayesian networks, (Pavlik Jr, Cen & Koedinger 2009) used learning curve analysis to automatically generate domain models, (Rai, Gong & Beck 2009) used dirichlet priors to improve model parameter plausibility, (Ritter, Harris, Nixon, Dickison, Murray & Towle 2009) discussed how to reduce the knowledge tracing space, (Robinet, Bisson, Gordon & Lemaire 2007) developed an algorithm that searching for student intermediate mental steps, (Rus, Lintean & Azevedo 2009) introduced an automatic detection system for student mental models during prior knowledge activation in meta-tutor, (Stamper & Barnes 2009) proposed a unsupervised MDP value selection for automating ITS capabilities,(Vee, Meyer & Mannock 2006) discussed the understanding novice errors and error paths in object-oriented programming through log analysis, (Wang 2002) used data-mining technology for browsing log file analysis in asynchronous learning environment, (Yudelson, Medvedeva, Legowski, Castine, Jukic & Rebecca 2006) mined student learning data to develop high level pedagogic strategy in a medical ITS, (Zakrzewska 2008) has done the research no cluster analysis for users modeling in intelligent E-learning systems.

Adaptive Educational Systems

(Amershi & Conati 2009) combined unsupervised and supervised classification to build user models for exploratory, (Ba-Omar, Petrounias & Anwar 2007) proposed a framework for using web usage mining to personalise e-learning, (Bellaachia, Vommina & Berrada 2006) developed a framework for mining e-learning logs, (Ben-Naim, Marcus & Bain 2008) discussed the visualization and analysis of student interaction in an adaptive exploratory learning environment, (Desmarais & Gagnon 2006) developed bayesian student models based on item to item knowledge structures, (García, Romero, Ventura & De Castro 2009) proposed an architecture for making recommendations to courseware authors using association rule mining and collaborative filtering, (Romero, Ventura, García & de Castro 2009) developed a collaborative data mining tool for education, (Hämäläinen, Suhonen, Sutinen & Toivonen 2004) discussed data mining in personalizing distance education courses, (Hübscher, Puntambekar & Nye 2007) proposed a domain specific interactive data mining, (Hwang, Chang & Chen 2004) discussed the relationship of learning traits, motivation and performance-learning response dynamics,(Jong, Chan & Wu 2007) proposed a learning log explorer in e-learning diagnosis, (Karampiperis & Sampson 2005) developed an adaptive learning resources sequencing in educational hyper-media systems, (Kelly & Tangney 2005) used the machine learning to know the learning style, (Krištofič 2005) proposed a recommender system for adaptive hypermedia applications, (Lu 2004) developed a personalized e-learning material recommender system, (Lu, Li, Liu, Yang, Tan & He 2007) researched on personalized e-learning system using fuzzy set based clustering algorithm, (Muehlenbrock 2005) developed an automatic action analysis in an interactive learning environment, (Romero et al. 2004) discussed the knowledge discovery with genetic programming for providing feedback to courseware authors, (Romero, Ventura, Zafra & De Bra 2009) applied web usage mining for personalizing hyper-links in Web-based adaptive educational systems, (Simko & Bieliková 2009) discussed the automatic concept relationships discovery for an adaptive e-course, (Tang & McCalla 2003) proposed a smart recommendation for an evolving e-learning system, (Tsai, Tseng & Lin 2001) proposed a two-phase fuzzy mining and learning algorithm for adaptive learning environment, (Vialardi, Bravo & Ortigosa 2008) improved AEH courses through log analysis, (Wang, Weng, Su & Tseng 2004) discussed learning portfolio analysis and mining in SCORM compliant environment, (Wang, T-seng & Liao 2009) introduced the data mining for adaptive learning sequence in English language instruction, (Zinn & Scheuer 2006) got to know student in distance learning contexts.

Others

Tests Questionnaires

(Anozie & Junker 2006) predicted end-of-year accountability assessment scores from monthly student records in an online tutoring system, (Ayers & Junker 2006) discussed do skills combine additively to predict task difficulty in eighth grade mathematics, (Ayers, Nugent & Dean 2009) described a comparison of student skill knowledge estimates, (Barnes 2005) proposed a Q-matrix method:mining student response data for knowledge, (Bravo & Ortigosa 2009) detected symptoms of low performance using production rules, (Hernándeza et al. 2006) detecting cheats in online student assessments using Data Mining, (Lara et al. 2014) developed a system for knowledge discovery in e-learning environments within the European Higher Education Area–Application to student data, (Chen & Weng 2009) researched on mining fuzzy association rules from questionnaire data, (Chu, Hwang, Tseng & Hwang 2006) proposed a computerized approach to diagnosing student learning problems in health education, (Feng et al. 2005) looked for sources of error in predicting students knowledge,(Freyberger et al. 2004) used association rules to guide a search for best fitting transfer models of student learning, (Hwang 2005) developed a Data Mining approach to diagnosing student learning problems in sciences courses, (Madhyastha & Hunt 2009) researched on mining diagnostic assessment data for concept similarity, (Nugent, Ayers & Dean 2009) introduced the conditional subspace clustering of skill mastery, (Pardos et al. 2008) analysed of multi-skill math questions in ITS, (Pardos & Heffernan 2009) determined the significance of item order in randomized problem sets, (Pechenizkiy, Calders, Vasilyeva & De Bra 2008) researched on mining the student assessment data, (Pechenizkiy, Trcka, Vasilyeva, van der Aalst & De Bra 2009) processed the mining online assessment Data,(Spacco, Winters & Payne 2006) inferred use cases from unit testing, (Viola, Graf, Leo et al. 2006) analysed of learning styles by a data-driven statistical approach, (Winters, Shelton, Payne & Mei 2005) discussed topic extraction from item-level grades, (Li & Yamanishi 2001) researched on mining from open answers in questionnaire data, (Zoubek & Burda 2009) visualization of differences in data measuring mathematical skills.

Texts Contents

(Abbas & Sawamura 2008) explored the first step towards argument mining and its use in arguing agents, (Alfonseca, Rodríguez & Pérez 2007) introduced an approach for automatic generation of adaptive hyper-media in education with multilingual knowledge discovery techniques, (Burr & Spennemann 2004) discussed patterns of user behaviour in university online forums, (Dringus & Ellis 2005) used data mining as a strategy for assessing asynchronous discussion forums, (Kim, Chern, Feng, Shaw & Hovy 2006) discussed mining and assessing discussions on the web through speech act analysis, (Lee 2007) proposed predictive and compositional modeling with data mining in integrated learning environments, (Lin, Hsieh & Chuang 2009) discovered genres of online discussion threads via text mining, (Song, Lin & Yang 2007) developed an opinion mining in e-learning system, (Su, Song, Lin & Li 2008)proposed a web text clustering for personalized e-Learning based on maximal frequent item-sets, (Ueno 2004) discussed data mining and text mining technologies for collaborative learning, (Li & Yamanishi 2001) researched on mining from open answers in questionnaire data, (Zhang, Mostow, Duke, Trotochaud, Valeri & Corbett 2008) worked on mining free-form spoken responses to tutor prompts.

Others

(Abel, Bittencourt, Henze, Krause & Vassileva 2008) developed a rule-based recommender system for online discussion forums, (Bari & Lavoie 2007) predicted interactive properties by mining educational multimedia presentations, (Cakir, Xhafa, Zhou & Stahl 2005) proposed a thread-based analysis of patterns of collaborative interaction in chat, (Chen, Wei, Chen et al. 2008) researched mining e-Learning domain concept map from academic articles, (Chen & Chen 2009) proposed a mobile formative assessment tool based on data mining techniques for supporting web-based learning, (Drachsler, Hummel & Koper 2008) discussed personal recommender systems for learners in lifelong learning networks, (El-Kechaï & Després 2007) proposed the underlying causes that lead to the trainees erroneous actions to the trainer, (Farzan & Brusilovsky 2006) developed a social navigation support in a course recommendation system, (Hardof-Jaffe, Hershkovitz, Abu-Kishk, Bergman & Nachmias 2009) introduced how do students organize personal information spaces, (Huang, Cheng & Huang 2009) proposed a blog article recommendation generating mechanism using an SBACPSO algorithm,(Kay, Maisonneuve, Yacef & Zaïane 2006) worked on mining patterns of events in students teamwork data, (Kiu & Lee 2007) research on learning objects reusability and retrieval through ontological sharing, (Lee, Lee & Leu 2009) developed an application of automatically constructed concept map of learning to conceptual diagnosis of e-learning, (Nankani, Simoff, Denize & Young 2009) proposed a supporting strategic decision making in an enterprise university through detecting patterns of academic collaboration, (Ting, Ouyang & Zhu 2008) introduced a learning object relationship mining-based repository, (Prata, d Baker, Costa, Rosé, Cui & de Carvalho 2009) detected and understanding the impact of cognitive and interpersonal conflict in computer supported collaborative learning environments, (Reffay & Chanier 2003) discussed how social network analysis can help to measure cohesion in collaborative distance-learning,(Reyes & Tchounikine 2005) researched mining learning groups' activities in forum-type tools, (Stevens et al. 2005) developed a framework for integrating prior problem solving and knowledge sharing histories of a group to predict future group performance, (Tseng, Sue, Su, Weng & Tsai 2007) proposed a new approach for constructing the concept map, (Zheng, Xiong, Huang & Wu 2008) used methods of association rules mining optimization in in web-based mobile-learning system.

The number of publications about Educational data mining has grown exponentially in the last few years. A clear sign of this tendency is the appearance of the peer-reviewed journal JEducational data mining (Journal of Educational Data Mining) and two specific books on Educational data mining edited by Romero Ventura entitled: Data Mining in E-learning (Romero & Ventura 2006) and The Handbook of Educational Data Mining (Romero, Ventura, Pechenizkiy & Baker 2010) co-edited with Baker Pechenizkiy. There were also two surveys carried out previously about Educational data mining. The first one (Romero & Ventura 2007) is a former review of Romero Ventura with 81 references until 2005 in which papers were classified by the Data mining techniques used. In fact, the present survey is an improved, updated and much-extended version of this previous one with 306 references in which papers are classified by educational categories tasks, and the types of data used. It also shows some examples of new groups that have appeared since the 2005 survey such as social network analysis and constructing courseware. The other survey is a recent review by Ryan Yacef with 46 references encompassing up to 2009. This study uses mainly the top 8 most cited papers in the first 2005 consideration and the proceedings of Educational data mining08 and Educational data mining09 conferences; it also groups papers according to Educational data mining methods and applications, as describing in the next section.

2.2.5 Technologies and Applications

Analysis and Visualization of Data

The objective of the analysis and visualization of data is to highlight useful information and support decision-making. In the educational environment, for example, it can help educators and course administrators to analyze the students course activities and usage information to get a general view of a students learning. Statistics and visualization information are the two

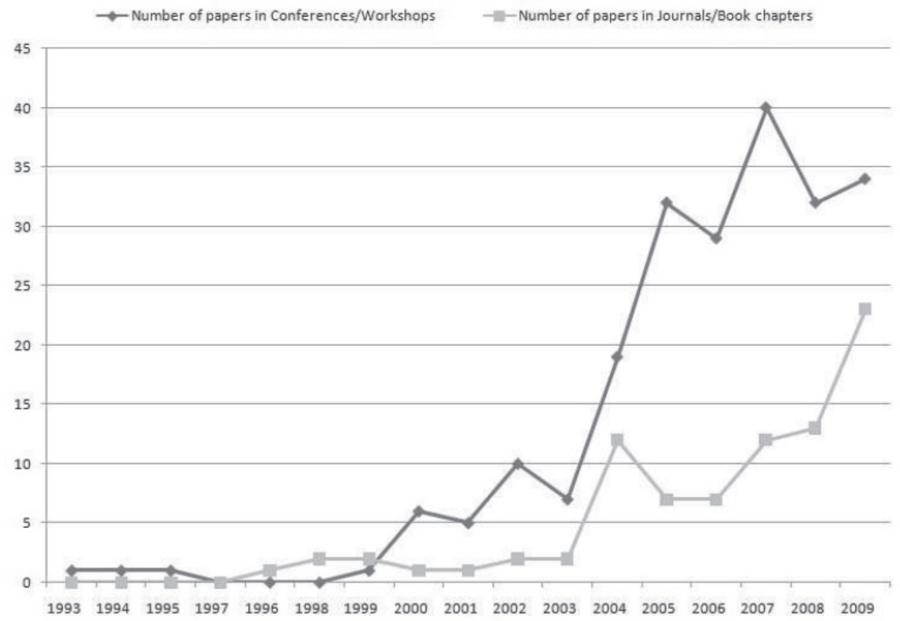


Figure 2.1: Publication Trend of the Educational Data Mining

primary techniques that have been most widely used for this task.

Statistics is a mathematical science concerning the collection, analysis, interpretation or explanation, and presentation of data. It is relatively easy to get basic descriptive statistics from statistical software such as SPSS. By?Using?educational data, this descriptive analysis can provide such global data characteristics as summaries and reports about learner behavior (Wu & Leung 2002). It is not surprising that teachers prefer pedagogically oriented statistics that are natural to interpret (Zinn & Scheuer 2006). On the other hand, teachers find the fine-grained statistics in log data too cumbersome to inspect or too time-consuming to interpret. Statistical analysis of educational data (logs files databases) can tell us such things as where students enter and exit, the most popular pages, the browsers students tend to use, patterns of use over time, (Ingram 2000); the number of visits, origin of visitors, number of hits, patterns of use throughout various time periods; number of visits and duration per quarter, top search terms, number of downloads of e-learning resources (Grob et al. 2004); number of different pages browsed, total time for browsing the different pages (Hwang et al. 2008); usage summaries and reports on weekly and monthly user trends and activities (Monk 2005); session statistics and meeting patterns (Pahl & Donnellan 2002); statistical indicators on the learners interactions in forums (Anaya & Boticario 2009); the amount of material students might go through, the order in which students study topics (Rahkila & Karjalainen 1999); resources used by students, resources valued by students (Sheard, Ceddia, Hurst & Tuovinen 2003); the overall averages of contributions to discussion forums, the amount of posting vs. replies, the amount of learner-to-learner interaction vs. learner-to-teacher interaction (Heathcote & Dawson 2005); the time a student dedicates to the course or a particular part of it (Pahl & Donnellan 2002); the learners behavior and time distribution , the distribution of network traffic over time (Zorrilla et al. 2005); the frequency of studying events, patterns of studying activity, timing and sequencing of events and the content analysis of students notes and summaries (Hadwin et al. 2007). Statistical analysis is also very useful to obtain reports assessing how many minutes the student has worked, how many minutes he has worked today, how many problems he has resolved and his correct percentage, our prediction of his score and his performance level.

Information visualization uses graphic techniques to help people understand and analyze data (Mazza 2009). Visual representations and interaction techniques take advantage of the human eyes broad bandwidth pathway into the mind to allow users to see, explore, and understand large amounts of information at once. There are several studies oriented toward visualizing different educational data such as patterns of annual, seasonal, daily and hourly user behavior on online forums (Burr & Spennemann 2004); the complete educational (assessment) process (Pechenizkiy et al. 2009); mean values of attributes analyzed in data to measure mathematical skills (Zoubek & Burda 2009); tutor-student interaction data from an automated reading tutor (Mostow et al. 2005); statistical graphs about assignments complement, questions admitted, exam score and so on (Shen et al. 2002); student tracking data regarding social, cognitive and behavioral aspects of students; student attendance, access to resources, overview of discussions and results on assignments and quizzes (Mazza & Milani 2004); weekly information regarding students' and groups' activity (Juan, Daradoumis, Faulin & Xhafa 2009); s-tudent progression per question as a transition between the types of questions (Ben-Naim et al. 2008); fingertip actions in collaborative learning activities (Avouris et al. 2005); deficiencies in a students primary understanding of individual concepts (Yoo et al. 2006) and higher-education student evaluation data (Jin et al. 2009); students interactions with online learning environments (Jovanović et al. 2007); the students on-line exercise work including students' interactions and answers, mistakes, teachers' comments and so on (Merceron et al. 2004); questions and suggestions in an adaptive tutorial (Ben-Naim, Bain & Marcus 2009); navigational behavior and the performance of the learner (Bellaachia et al. 2006); educational trails of Web-pages visited and activities done (Romero, Santos, Freire & Ventura 2008) and the sequence of learning objects and educational trails.

Recommendations for Students

The objective is to be able to make recommendations directly to the students with respect to their personalized activities, links to visits, the next task or problem to be done, etc. and also to be able to adapt learning contents, interfaces and sequences to each particular student. Several DM techniques have been used for this task, but the most common are association rule mining, clustering and sequential pattern mining. Sequence Sequential pattern mining aims to discover the relationships between occurrences of subsequent events, to find if there exists any particular order in the occurrences (Dong & Pei 2007).

Sequential pattern mining has been developed to personalize recommendations on learning content based on learning style and web usage habits (Zhang, Liu & Liu 2008); to study eye movements (of students reading concept maps) in order to detect when focal actions overlap unrelated actions

(Nesbit, Xu, Winne & Zhou 2008); for developing personalized learning scenarios in which the learners are assisted by the system based on patterns and preferred learning styles (Ba-Omar et al. 2007); to identify significant sequences of activity indicative of problems success in order to support student teams by early recognition of problems (Kay et al. 2006); to generate personalized activities for learners (Wang et al. 2004); for personalizing based on itineraries and long-term navigational behavior (Mor & Minguillón 2004); to recommend the most appropriate future links for a student to visit in a web-based adaptive educational system (Romero, Ventura, Zafra & De Bra 2009); to include the concept of recommended itinerary in SCORM standard by combining teachers expertise with learned experience (Mor & Minguillón 2004); to select different learning objects for different learners based on learner profiles and the internal relation of concepts (Shen & Shen 2004); for personalizing activity trees according to learning portfolios in a SCORM compliant environment (Wang et al. 2004); for recommending lessons that a student should study next while using an adaptive hypermedia system (Krištofič 2005); to discover LO relationship patterns to suggest related learning objects to learners (Ting et al. 2008); for adapting learning resource sequencing (Karampiperis & Sampson 2005).

Association rule mining has been used to recommend online learning activities or shortcuts on a course website (Zaíane 2002); to produce recommendations for learning material in e-learning systems (Markellou, Mousourouli, Spiros & Tsakalidis 2005); for content recommendation based on educationally-contextualized browsing events for web-based personalized learning (Wang 2008); for recommending relevant discussions to the students (Abel et al. 2008); to provide students with personalized learning suggestions by analyzing their test results and test related concepts (Chu et al. 2006); for making recommendations to courseware authors about how to improve adaptive courses (García et al. 2009); for building a personalized e-learning material-recommender system to help students find learning materials (Lu 2004); for course recommendation with respect to optimal elective courses (Wen-

Shung Tai et al. 2008); for designing a material recommendation system based on the learning actions of previous learners (Liu & Shih 2007).

Clustering has been developed to establish a recommendation model for students in similar situations in the future (Wang & Shao 2004); for grouping web documents using clustering methods in order to personalize e-learning based on maximal frequent itemsets (Su et al. 2008); for providing personalized course material recommendations based on learner ability (Lu et al. 2007) and to recommend to students those resources they have not yet visited but would find most helpful.

Other Data mining techniques used are: neural networks and decision trees to provide adaptive and personalized learning support (Guo & Zhang 2009); production rules to help students to make decisions about their academic itineraries (Vialardi et al. 2009); decision tree analysis to recommend optimal learning sequences to facilitate the students learning process and maximize their learning outcome (Wang et al. 2009); learning factor transfers and Q-matrixes to generate domain models that will sequence item-types to maximize learning (Pavlik Jr et al. 2009); an item order effect model to suggest the most efficient item sequences to facilitate learning (Pardos & Heffernan 2009); a fuzzy item-response theory to recommend appropriate courseware for learners (Chen et al. 2004); intelligent agent technology and SCORM based course objects to build an agent-based recommender system for lesson plan sequencing in web-based learning (Yang et al. 2002); data mining and text mining to recommend books related to the books that the target pupil has consulted (Nagata, Takeda, Suda, Kakegawa & Morihiro 2009); case-based reasoning to offer contextual help to learners, providing them with an adapted link structure for the course (Heraud et al. 2004); Markov decision process to automatically generate adaptive hints in ITS (to identify the action that will lead to the next state with the highest value) (Stamper & Barnes 2009) and an extended Serial Blog Article Composition Particle Swarm Optimization (SBACPSO) algorithm to provide optimal recommended materials to users in blog-assisted learning (Huang, Cheng & Huang 2009).

Student Modeling

The objective of student modeling is to develop cognitive models of human users students, including a modeling of their skills and declarative knowledge. Data mining has been applied to automatically consider user characteristics (motivation, satisfaction, learning styles, effective status, etc.) And learning behavior in order to automate the construction of student models (Frias-Martinez, Chen & Liu 2006). Different Data mining techniques and algorithms have been used for this task (mainly, Bayesian networks).

Several data mining algorithms (Naive Bayes, Bayes net, support vector machines, logistic regression and decision trees) have been compared to detect student mental models in intelligent tutoring systems (Rus et al. 2009). Unsupervised (clustering) and supervised (classification) machine learning have been proposed to reduce development costs by building user models and to facilitate transferability in intelligent learning environments (Amershi & Conati 2009). Clustering and classification of learning variables have been used to measure the online learner's motivation (Hershkovitz & Nachmias 2008).

Bayesian networks have been used to make predictions about student knowledge, i.e. the probability that student has of knowing a skill at a given time through cognitive tutors (Baker et al. 2008); to detect students learning styles in a web-based education system (García et al. 2007); to predict whether a student will answer a problem correctly (Jonsson et al. 2005); to model a students changing state of knowledge during skill acquisition in ITS (Chang et al. 2006); to infer unobservable learning variables from students help-seeking behavior in a web-based tutoring system (Arroyo, Murray, Woolf & Beal 2004) and for knowledge tracing in order to verify the impact of self-discipline on pupils knowledge and learning (Gong et al. 2009).

Sequential pattern mining has been used automatically to acquire the knowledge to construct student models (Antunes 2008); to identify meaningful user characteristics and to update the user model to reflect newly gained knowledge (Andrejko, Barla, Bieliková & Tvarozek 2007) and for predicting students intermediate mental steps in sequences of actions stored by -learning environments based on problem solving (Robinet et al. 2007).

Association rule algorithms have been applied to personality mining based on web-based education models in order to reduce learners personality characteristics (Huang et al. 2007) and for student modeling in intelligent tutoring systems.

Other Data mining techniques and models have also been used for student modeling. A logistic regression model has been used to construct transfer models (to accurately predict the level at which a student represents knowledge) (Feng & Beck 2009). A learning agent that models student behaviors using linear regression has been constructed in order to predict the probability that the students next response has of being correct (Beck & Woolf 2000). Inductive logic programming and a profile extractor system (using numeric algorithms) have been developed to induce student profiles in e-learning systems (Licchelli et al. 2004). The Markov decision process has?been proposed to create student models automatically by generating hints for an intelligent tutoring that learns (Barnes & Stamper 2008). Fuzzy techniques have used student models in web-based learning environments in order to produce advice for the teachers (Kosba et al. 2005). A dynamic learning response model has been developed for inferring, testing and verifying student learning models on an adaptive learning website (Hwang et al. 2004). Bootstrapping novice data can create an initial skeletal model of a tutor from log data collected from actual use of the tool by students (McLaren et al. 2004). A collaborative-based data mining approach has been developed for diagnostic and predictive student modeling purposes in integrated learning environments (Lee et al. 2007). Multiple correspondence analysis and cross-validation by correlation?analysis have been applied to identify learning styles in ILS (Index of Learning Styles) questionnaires (Viola et al. 2006). The Q-matrix method has been used to create concept models that represent relationships between concepts and questions, and to group student test question responses according to concepts (Barnes 2005). An algorithm to estimate Dirichlet priors has been developed to produce model parameters that provide a more plausible picture of student knowledge (Rai et al. 2009). Self-organizing maps and principal component analysis have been applied to predictive and compositional modeling of the student profile (Lee 2007). A clustering algorithm (K-means) has been developed to model student behavior with a very small set of parameters without compromising the behavior of the system (Ritter et al. 2009).

The objective is to create groups of students according to their customized features, personal characteristics, etc. Then, the clusters groups of students obtained can be used by the instructor developer to build a personalized learning system, to promote effective group learning, to provide adaptive contents, etc. The Data mining techniques employed in this task are classification (supervised learning) and clustering (unsupervised learning). Cluster analysis or clustering is the assignment of a set of observations into subsets (called clusters) so that observations in the same cluster have some points in common (Romesburg 2004).

Different clustering algorithms have been used to group students, such as: hierarchical agglomerative clustering, Kmeans and model-based clustering to identify groups of students with similar skill profiles (Ayers et al. 2009); a clustering algorithm based on large generalized sequences to find groups of s-tudents with similar learning characteristics based on their traversal path patterns and the content of each page they have visited (Tang & McCalla 2002); model-based clustering to automatically discover useful groups from LMS data to obtain profiles of student behavior (Talavera & Gaudioso 2004); a hierarchical clustering algorithm for user modeling (learning styles) in intelligent e-learning systems in order to group students according to their individual learning style preferences (Zakrzewska 2008); discriminating features and external profiling features (pass-fail) to support teachers in collaborative student modeling (Gaudioso, Montero, Talavera & Hernandez-del Olmo 2009); an improvement in the matrix-based clustering method for grouping learners by characteristics in e-e-learning (Zhang et al. 2007); a fuzzy clustering algorithm to find interested groups of learners according to their personality and learning strategy data collected from an online course (Tian et al. 2008); a hybrid method of clustering and Bayesian networks to group students according to their skills (Hämäläinen et al. 2004); a K-means clustering algorithm for effectively grouping students who demonstrate similar learning portfolios (students assignment scores, exam scores and online learning records) (Lara et al. 2014); an Expectation-Maximization algorithm to form heterogeneous groups according to student skills (Myller et al. 2002); a Kmeans clustering algorithm to discover interesting patterns that characterize the work of stronger and weaker students (Perera et al. 2009); a conditional subspace clustering algorithm to identify skills which differentiate students (Nugent et al. 2009); a two-step cluster analysis to classify how students organize personal information spaces (piling, one-folder, small-folders and big-folder filing) (Hardof-Jaffe et al. 2009); hierarchical cluster analysis to establish the proportion of students who get an exercise wrong or right (Barker-Plummer, Cox & Dale 2009); a genetic clustering algorithm to solve the problem of allocating new students (which places new students into classes so that the gaps between learning levels in each class is minimum and the number of students in each class does not exceed the limit) (Zukhri & Omar 2008).

Several classification algorithms have been applied in order to group students, such as: discriminant analysis, neural networks, random forests and decision trees for classifying university students into three groups (low-risk, medium-risk and high-risk of failing) (Superby et al. 2006); classification and regression tree, chi-squared automatic interaction detection and C4.5 algorithm for the automatic identification of the students cognitive styles (Lee, Chen, Chrysostomou & Liu 2009); a classification and regression tree to create a decision tree model to illustrate a users learning behavior in order to analyze it according to different cognitive style groups (Lee et al. 2007); a hidden Markov-model-based classification approach to characterize various types of users through their navigation or content access patterns (Fok et al. 2005); decision trees for classifying students according to their accumulated knowledge in e-learning systems (Cristian & Dan 2006); C4.5 decision tree algorithm for discovering potential student groups with similar characteristics who will react to a particular strategy (Chen et al. 2000); Naive Bayes classifier to classify learning styles that describe learning behavior and educational content (Kelly & Tangney 2005); genetic algorithms for grouping students according to their profiles in a peer review content (Crespo et al. 2005); classification trees and multivariate adaptive regression to identify those students who tend to take online courses and those who do not (Yu et al. 2008); decision tree and support vector machine for assessing an activity by more than one lecturer using a pair-wise learning model (Quevedo & Montañés 2009); a classification algorithm for speech act patterns to determine participants roles and identify discussion threads (Kim et al. 2006) and K-nearest neighbor (K-NN) classification combined with genetic algorithms to detect and classify student learning styles (Chang, Kao, Chu & Chiu 2009).

2.3 Summary

This chapter began with the preliminaries of this thesis. The first section discussed the definition of the coupling concept and given a practical solution to deal with the categorical date that reveals the implicit coupling relation between the data objects. Furthermore, the foundation of the classification method is introduced for the student performance prediction task and the text mining basics for the student social media sentiment analysis work. In the following chapters will illustrate the detail of the proposed coupling analysis of the Educational Data, and the experiment will demonstrate the performance of the proposed method. The second section is a review of the state of the art with respect to educational data analysis and digests the most relevant work in this area to date. Each study has been classified, not only by the type of data and data analysis techniques used, but also and more importantly, by the kind of the educational task that they resolve. Educational data mining has been introduced as an up and coming research area related to several well-established areas of research including e-learning, adaptive hyper-media, intelligent tutoring systems, web mining, data mining, etc. It has been seen how fast educational data analysis is growing as reflected in the increasing number of contributions published every year in International Conferences and Journals, and the number of distinct tools specially developed for applying data mining algorithms in educational data environments. So, it could be said that Educational data mining is now approaching its adolescence, that is, it is no longer in its early days but is not yet a mature area. Some interesting future lines has been found but for it to become a more mature area it is also necessary for researchers to develop more unified and collaborative studies instead of the current plethora of multiple individual proposals and lines. Thus, the full integration of coupling analysis in the educational environment will become a reality, and fully operative implementations could be made available not only to researchers and developers but also for relevant users.

Chapter 3

A Coupled Similarity based K-Nearest Centroid Classifier for Student Performance Prediction

3.1 Introduction

This thesis begins with a student performance prediction task and focuses on an overall performance prediction for the new incoming university students each year. There are more than 100,000 historical instances of training, and every year there is need to predict the performance risk for about 10,000 new enrolled students. Hence, not only is the prediction performance necessary, but also achieving it with maximum efficiency is critical. The idea integrates coupling similarity with customary classification methods. Accordingly, a novel effective classification method that applies a weighted K-Nearest Centroid to obtain the coupled similarity for student data with none-iid features has been designed. From value and attribute perspectives, coupled similarity serves as a metric for nominal objects, which considers not only intra-coupled similarity within an attribute but also an inter-coupled similarity between attributes. The student performance related data source is significant, however the student performance prediction task is time effective during the prediction phase, because it is required to know their performance prediction result as early as possible. Hence, this thesis proposes an efficient method that measures the nearest centroid instead of comparing all of the objects for the prediction. Extensive experiments on UCI and student data sets reveal that the proposed method outperforms classical methods for higher accuracy and the proposed method completes the predictions much more quickly. Because the student data source for the students that there is mainly the categorical data, the student performance prediction task can be generalized into a classification task for categorical data.

Most of the existing classification methods make an assumption about the independence among values, attributes and objects(Cao 2013). For example, SVM(Joachims 1999) tries to classify objects by converting all the nominal features into binary numerical features(Hsu, Chang, Lin et al. 2003). More precisely, an attribute with ν distinct categorical values can be converted into ν separate new features. Consequently, this kind of methods believes different values of each feature share equal discriminative power in classification. However, in real world data, such as multi-agent interactions and agent behaviours, the coupling relationships and heterogeneity among data features and values are ubiquitous. This is a significant challenge to existing theories and systems.

Some similarity metrics try to measure distance by geometric analogies which represent the relationship of data values. For example, the similarity between 10 and 12 is greater than that of 10 and 2. There are many similarity metrics which have been explored for numerical data, such as Euclidean and Minkowski distances (Gan et al. 2007). By contrast, similarity measurement over nominal variables has received much less attention. In a supervised learning process, heterogeneous distances (Wilson & Martinez 1997) and modified value distance matrix (*MVDM*) (Cost & Salzberg 1993) describes the similarity between categorical values. In unlabeled data, only limited research (Gan et al. 2007), such as simple matching similarity (*SMS*, which only uses 0s and 1s to distinguish similarities between distinct and identical categorical values) and occurrence frequency (Boriah, Chandola & Kumar 2008), discusses the similarity between nominal values. (Gan et al. 2007) defined a specific similarity measure between attribute values, by extracting the intensity of the relationship between two data objects, which frequently resemble each other, to obtain a larger similarity between them.

(Boriah et al. 2008, Gan et al. 2007) discussed the similarity between categorical attributes. Cost and Salzberg (Cost & Salzberg 1993) proposed *MVDM* based on labels, while Wilson and Martinez (Wilson & Martinez 1997) studied heterogeneous distances for instance based learning. Some data mining techniques for nominal data (Ahmad & Dey 2007*a*, Boriah et al. 2008) existed. The most famous is the distance measure and its diverse variants such as Jaccard coefficients (Gan et al. 2007), which are intuitively based on the principle that the similarity measure is 1 with identical values and is otherwise 0. More recently, attribute value frequency distribution has been considered for similarity measures (Boriah et al. 2008); neighbourhood-based similarities (Houle, Oria & Qasim 2010) are explored to describe the object neighbourhood by using an overlap measure. These are different from the method that is being proposed here, which directly reveals the similarity between a pair of objects.

Recently, a number of researchers have pointed out that the attribute value similarities are also dependent on their coupling relations (Boriah et al. 2008, Cao & Philip 2012). Das and Mannila presented the Iterated Contextual Distances algorithm, convinced that the similarities among features and objects are inter-dependent (Das & Mannila 2000). Ahmad and Dey (Ahmad & Dey 2007*a*) proposed a computing dissimilarity matric by considering the value's co-occurrence. While the dissimilarity criterion of the latter leads to high accuracy, the computation is usually very costly, which limits its application in large-scale problems.

3.2 Problem Statement

This section firstly reviews the definition of the coupled similarity proposed by (Wang et al. 2011).

Intra-Coupling Interaction

Definition 3.1 *The* ***Intra-coupling Attribute Value Similarity (IaAVS)*** *between attribute values x and y of feature a_j is:*

$$\delta_j^{Ia}(x, y) = \frac{|g_j(x)| \cdot |g_j(y)|}{|g_j(x)| + |g_j(y)| + |g_j(x)| \cdot |g_j(y)|}. \tag{3.1}$$

Inter-Coupling Interaction

Definition 3.2 *The* ***Inter-coupling Relative Similarity based on Power Set (IRSP)*** *between attribute values x and y of feature a_j based on another feature a_k is:*

$$\delta_{j|k}^{P}(x, y) = \min_{W \subseteq V_k} \{2 - P_{k|j}(W|x) - P_{k|j}(\overline{W}|y)\}, \tag{3.2}$$

where $\overline{W} = V_k \backslash W$ is the complementary set of a set W under the complete set V_k.

Definition 3.3 *Given an information table S, the* ***Inter-coupling Attribute Value Similarity (IeAVS)*** *between attribute values x and y of feature a_j is:*

$$\delta_j^{Ie}(x, y) = \sum_{k=1, k \neq j}^{n} \alpha_k \delta_{j|k}(x, y), \tag{3.3}$$

where α_k is the weight parameter for feature a_k, $\sum_{k=1}^{n} \alpha_k = 1$, $\alpha_k \in [0, 1]$, and $\delta_{j|k}(x, y)$ is one of the inter-coupling relative similarity candidates.

Coupled Similarity

Definition 3.4 *The* ***Coupling Attribute Value Similarity (CAVS)*** *between attribute values x and y of feature a_j is:*

$$\delta_j^{A}(x, y) = \delta_j^{Ia}(x, y) \cdot \delta_j^{Ie}(x, y) \tag{3.4}$$

where δ_j^{Ia} and δ_j^{Ie} are IaAVS and IeAVS, respectively.

Definition 3.5 *Given an information table S, the* ***Coupling Object Similarity (COS)*** *between objects u_{i_1} and u_{i_2}:*

$$COS(u_{i_1}, u_{i_2}) = \sum_{j=1}^{n} \delta_j^A(x_{i_1 j}, x_{i_2 j}), \tag{3.5}$$

where δ_j^A is the CAVS *measure defined in (3.4), $x_{i_1 j}$ and $x_{i_2 j}$ are the attribute values of feature a_j for objects u_{i_1} and u_{i_2} respectively, and $1 \leq i_1, i_2 \leq m$, $1 \leq j \leq n$.*

3.3 Coupled Similarities Based Classification

The original coupled similarity metric is used in the clustering task but it is very straightforward to modify it into classification task. In this section, a novel classification method based on the coupled similarity metric is proposed; given a data set $\mathcal{D}$ =$\{d_1, d_2, \ldots, d_n\}$, and Π=$\{\pi_1, \pi_2, \ldots, \pi_n\}$ are the classes of the data set. The proposed method aims to extract more information from feature to feature and feature to class by applying coupling similarities, which can also been referred to as coupling distance. In terms of the distance based classification task, the K-Nearest-Neighbors(KNN) is the most popular method; however, it lacks efficiency when it does the classification. In detail, when the KNN algorithm performs the classification task, it needs to compute the distance between the given object d_n to each of the other objects in $\mathcal{D}$ to find the K-nearest objects, and then ascertain to which cluster this object belongs. In contrast, the proposed method only calculates the distance between the object to the cluster's centroid, which dramatically reduces comparison time Figure3.1, and as the more training sets there are, the more time will be saved. This is a generalized process to find the most representative object to stand for the similar objects within one cluster. Moreover, the proposed method also establishes a novel method to improve the weaknesses of KNN classification. As KNN is based on equivalent significance to neighbours, the proposed adds weight to every object to enhance the discriminative power. The experiments show that the proposed

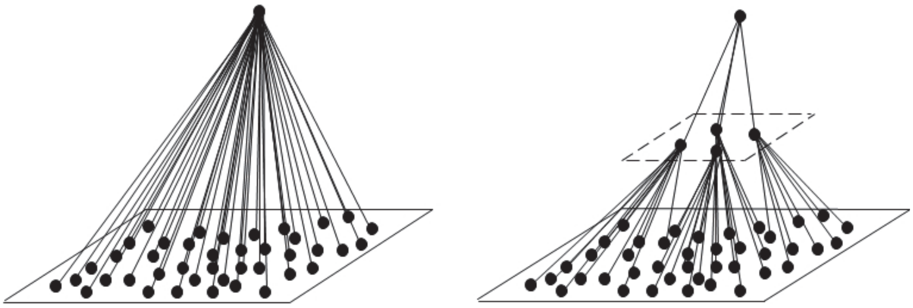


Figure 3.1: Comparison Times with and without Clustering

method reduces classification time substantially, without a loss of classification accuracy.

3.3.1 Clustering Within the Class with Coupled Similarities

In this section, a coupling similarities based clustering method is illustrated. Firstly, by $definition$3.5, a coupled similarity between two objects $COS(d_i, d_j)$ can be calculated. After this, for the classification task, it computes the coupling similarities within one class first because this thesis assumes there might be more coupling relations within one class than between two classes. In order to enhance the speed of the clustering process, all the object s within the data set $\mathcal{D} = \{d_1, d_2, \ldots, d_n\}$ has been enumerated into a comparison table $Table$3.1 and then the coupling similarities were calculated between each of them. Since this definition of similarity is a relative value, it only can be applied when given two objects, which means it cannot create a middle point between two objects. Furthermore, the mean of two categorical attributes cannot be calculated either, for instance, it is hard to say what gender is between male and female. As a result, a traditional clustering method like K-Means(Teknomo 2006) cannot be applied directly, because it cannot find the mean point within a group of objects. To solve this problem, the spherical K-Means(Zhong 2005) clustering method is used to instead of K-Means as this clustering method.

Table 3.1: Coupled Similarity Between Objects

Object Pairs	Similarity
d_1, d_2	0.23
d_1, d_3	0.31
. . .	. . .
d_n, d_m	s

3.3.2 Spherical K-Means Clustering with Coupled Similarities

Let d_1, d_2 be two categorical objects from the data set $\mathcal{D} = \{d_1, d_2, \ldots, d_n\}$, the similarities among the objects is based on definitioninition 3.5. The clustering process partitions data set $\mathcal{D}$ into T clusters, and each of the clusters can be named as $\mathcal{C} = \{c_1, c_2, \ldots, c_t\}$ respectively. The perfect solution can be formally described as the following maximization problem:

$$\{c_t\}_{j=1}^{k} = \underset{\{c_t\}_{t=1}^{k}}{\arg\max} \sum_{t=1}^{k} \sum_{d_i \in c_t} Cos(m_t, d_i) \tag{3.6}$$

$\{c_t\} = \{d_{t1}, d_{t2}, \ldots, d_{tn}\}$ is a cluster with certain objects, A centroid point m_t of cluster c_t is an object within the c_t which has minimal similarity to all other objects within the cluster, for any object d' in c_t, the centroid point m_t that

$$\sum_{d_i \in c_t} Cos(m_t, d_i) \leq \sum_{d_i \in c_t} Cos(d', d_i) \tag{3.7}$$

The clustering method is straightforward, and is very similar to K-Means. Firstly, it randomly chooses K object from data set $\mathcal{D}$ as the centroid object m_k, m stands for the temp mode of the cluster and k is the cluster id for each cluster. Secondly, it allocates each object d_n to theirs nearest centroid object m_k as a intermediate cluster c_k, where c_k contains a set of objects $\{d_{k1}, d_{k2}, \ldots, d_{kn}\}$ which are the nearest objects to this centroid object m_k.

Thirdly, it searches for a new centroid object within each cluster c_k, the new centroid object being the object which has minimal similarity to all other objects with in the cluster.When the new centroid object has been confirmed, the process is repeated to assign each object to the new centroid object to reform the cluster. Finally,it iterates the process until the centroid object is fixed for any cluster c_t.

$$m_t^n = m_t^{n+1} \tag{3.8}$$

n stands for the iteration times. Meanwhile, in some extreme case, the centroid object cannot be fixed at all,so there is an alternative criterion that

$$|(\sum_{t=1}^{k} \sum_{d_i \in c_t} Cos(m_t, d_i))^n - (\sum_{t=1}^{k} \sum_{d_i \in c_t} Cos(m_t, d_i))^{n+1}| \leq \varepsilon \tag{3.9}$$

Similar to the above, n is the iteration time, ε is the certain threshold, if the "change" of the cluster after iteration is not significant, the searching algorithm will stop. The Spherical K-Means clustering method prevents the problems which K-Means leads to, thus it suitable for our coupled similarity based clustering.

3.3.3 Classification with Coupled Similarities Weighted Cluster Centroid

To simplify the problem, the binary classification task is used as an example. Once the clustering process has been finished, there are several clusters within both positive class π_A and negative class π_B. Moreover, each centroid of the cluster has its unique value for classification, due to the fact that difference of the coupled similarity between them is substantial. The coupled similarity cannot be expressed in 2D space, since all the similarities are relative and cannot be drawn on one flat picture. However, for simplicity, the following figure illustrates the different similarities between clusters.

As Figure 3.2 shows, each circular represents a cluster, and the caption of the circular C_{π_A} and C_{π_B} stands for the centroid object in both the positive

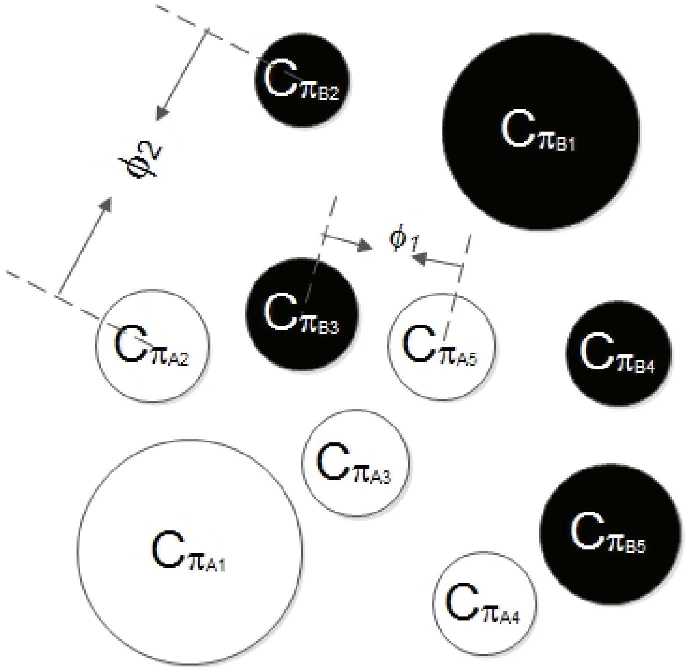


Figure 3.2: The Training Data Set After Clustering

class and negative class respectively. Moreover, it also uses a different color to represent the clusters which belong to different classes. The ϕ_1 and ϕ_2 is the coupled similarity between two centroids $C_{\pi_{A3}}$ and $C_{\pi_{B5}}$ and $C_{\pi_{A2}}$ and $C_{\pi_{B2}}$. It is clear to see that, the coupled similarity ϕ_2 is significantly larger than ϕ_1. When undertaking a classification task, differences in the distance will affect the result significantly. Unfortunately, the classic KNN algorithm neglects this difference and judge every point as equivalently significant. More precisely, when it undertakes a classification task with an incoming object d_n, it only counts the numer of the k nearest neighbors $Count(\chi^k(d_n))$ which belong to each class, where the $\chi^k(d_n)$ denote the set of the nearest neighbors of object d_n, and if the number of neighbours belonging to class A is larger than the number of neighbours belonging to class B, then it classifies the object to class A, without considering the unique value of each of its neighbours. If F is the classification function, the classification process of traditional KNN can be describe as:

$$F(d_n) = \begin{cases} d_n \in \pi_A & Count(\chi^k(d_n) \in \pi_A) > \\ & \qquad Count(\chi^k(d_n) \in \pi_A) \\ d_n \in \pi_B & Count(\chi^k(d_n) \in \pi_B) > \\ & \qquad Count(\chi^k(d_n) \in \pi_A) \end{cases} \tag{3.10}$$

This thesis proposes a novel classifier with a weighted cluster centroid, which comprehensively gathers the information of the coupled similarity from every centroid object C_{π_A} in one class π_A to all other objects $d^j_{\pi_{\bar{A}}}$ belonging to the opposite class $\pi_{\bar{A}}$. The reason for this is that, by the concept of coupling, every object can be described by other objects, which have a relationship with it. The proposed method utilizes this information to give centroid object, the computation of coupling similarities weight is quite straightforward:

$$W(C_{\pi_{An}}) = \sum_{i=1}^{m} \sum_{j=1}^{n} Cos(d^i_{\pi_A}, d^j_{\pi_{\bar{A}}}) \tag{3.11}$$

Finally, to classify an incoming object by accumulating the coupling similarities to every centroid object and adding these weights, the classification function becomes:

$$F(d_n) = \begin{cases} d_n \in \pi_A & \sum_{i=1}^{k} W(\chi^k(d_n) \in \pi_A) > \\ & \qquad \sum_{i=1}^{k} W(\chi^k(d_n) \in \pi_B) \\ d_n \in \pi_B & \sum_{i=1}^{k} W(\chi^k(d_n) \in \pi_B) > \\ & \qquad \sum_{i=1}^{k} W(\chi^k(d_n) \in \pi_A) \end{cases} \tag{3.12}$$

3.4 Experiment and Evaluation

In this section, empirical experiences of some UCI data will be explored. Without losing generality, four UCI data sets for this experiment have been chosen. Subsequently, it has been performed on the student performance prediction task. More precisely, the sets referred to are sonar,hepatitis,horse-colic and SPECTF. In this experiment, the proposed method is compared to the two most widely used classification method, the C4.5 decision tree and support vector machines(SVMs) and also the original k-NN method based on the overlap similarity for the categorical data. The methods' relative precision in the classification tasks but also the recall and f1 measurement of the classification result. The C4.5 decision tree set the confidence C to 0.5 and minimal count of leaf m to 2, the SVM set the cost c to 1 and eps e to 0.001, the k-NN algorithm set the k to 3, and this method sets the cluster number K to one-tenth of the attribute's numbers. Since not all the features are categorical, if it is numerical it can be discreted into five equal frequency categorical values. The SVMs transformed that the categorical feature into one hot vector numerical features. The k-NN used overlap similarity as the distance metric. The C4.5 algorithm and k-NN based on weka(Hall, Frank, Holmes, Pfahringer, Reutemann & Witten 2009), the SVMs algorithm based on libsvm(Chang & Lin 2011), the proposed method developed in JAVA is also based on WEKA core functions. Because these methods are not implemented in one programming language, it is hard to compare the efficiency properly; the classification performance is compared only in this experiment. The environment of the experiment is on a desktop PC with 4G ram and Intel i5 CPU, all the algorithm runs on single thread, and no parallel computation has been added.

Classification Performance Comparison

To evaluate the classification performance, four basic measures should be introduced. The true positive (tp) is the classifier classified as positive and the ground truth is also positive. The true negative (tn) is the classifier classified as negative and the ground truth is also negative. The false negative (fn) is the classifier classified as negative and the ground truth is positive. The false positive (fp) is the classifier classified as positive and the ground truth is negative. The classification performance matrices are all based on

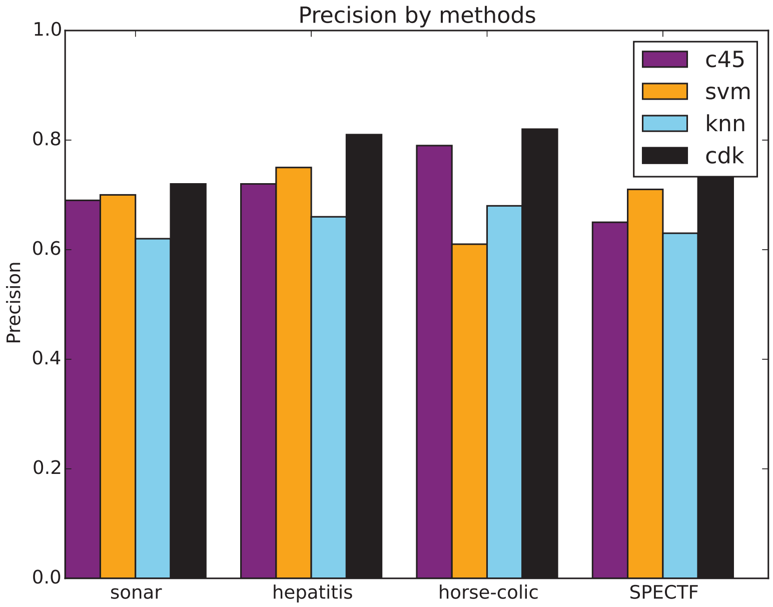


Figure 3.3: Classification Precision Comparison

the three measurements. They can be defined as follows:

$$precision = \frac{tp}{tp+fp}$$

$$recall = \frac{tp}{tp+fn}$$

$$f1 = 2 * \frac{precision*recall}{precision+recall}$$

The experiment results (Figure 3.3) show that the proposed method ”Coupled Similarity based K-Nearest Centroid Classification” (CDK) outperforms the existing method in all four data sets with different extensions according to the data distribution, which means that the performance of the proposed method is related to the data itself; that is, the more coupling relationships among the data, the more precise the classification will be.

Since not all the data sets have balanced class distribution, precision is not the vital metric of evaluation for the performance of the classifier. For instance, if a data set has 99 percent of data belonging to one class and the

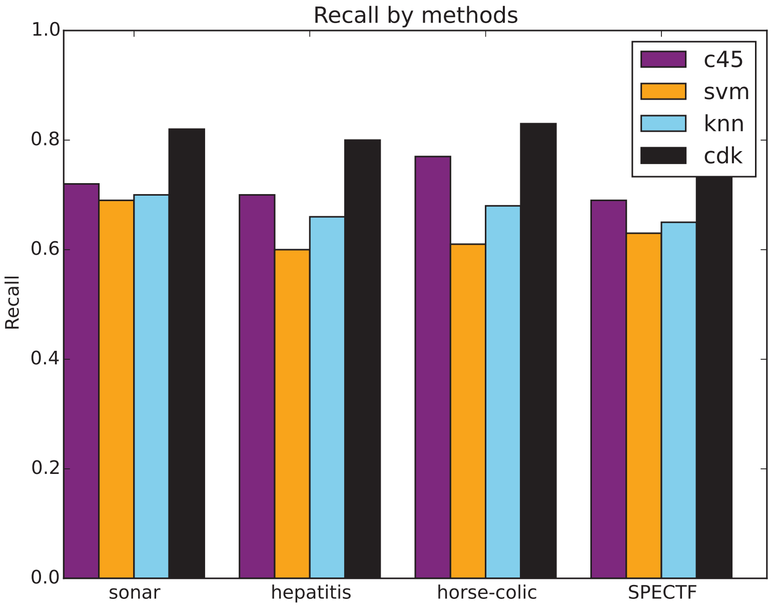


Figure 3.4: Classification Recall Comparison

remainder belongs to another class, the classifier simply classifies all the data into the larger class, so it can reach 99 percent of precision for the classification task. This problem is serious in the SVM because it tries to minimize the misclassified number. The experiment of the recall (Figure 3.4) aims to reveal this problem. The results show that the proposed method makes a significant improvement on the recall criteria, as the proposed method fully considers the class variation of the data in the weighted centroid classification process.

Similar to the recall metric, it consolidates the advantage of the proposed method by comparing the f1 measure (Figure 3.5). The experiment result also indicates that the proposed method outperforms the other methods. The above three experiments verified the significance of the proposed method. The following section will discuss the application of the proposed method on educational data, and also an efficiency experiment will show how the proposed method reduces time cost dramatically.

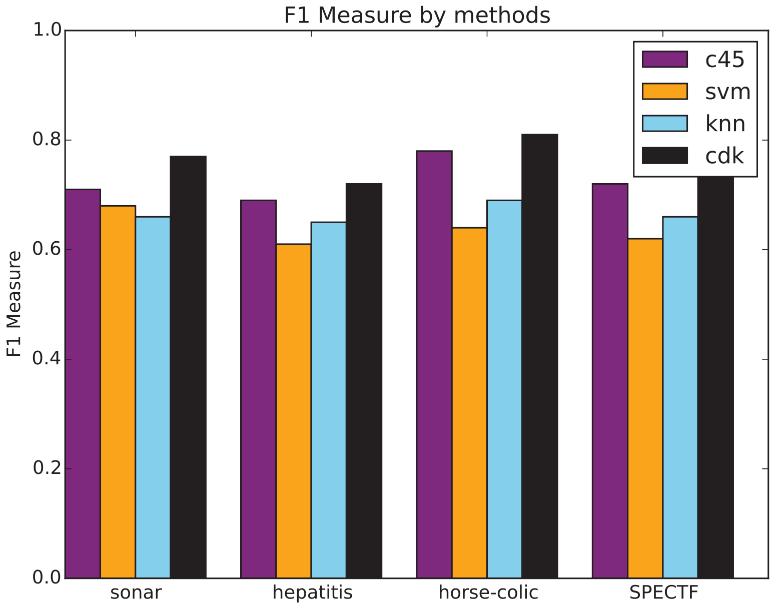


Figure 3.5: Classification F1 Measure Comparison

3.5 Application

In this section, the performance of this novel method by applying it to real world educational data is illustrated. Educational data analysis is a growing research topic, and a significant number of studies mention predicting the probability of students failing their subjects. Unfortunately, most of the previous work has only been done for research purposes, very few of them have been implemented in real world scenarios.

This thesis is different as the proposed methods all have already been used in real world systems and have a continuous impact on the teaching and learning processes of the university. This program has been implemented by the student services unit for several years to provide early intervention services for at-risk students. They contact all first year students to identify students who are at risk of having academic difficulties. Appropriate early interventions are recommended to them to enhance their experience of the university and to reduce the attrition rate. Currently, the challenge is to improve the efficiency of the original process. Considering that all first year students are contacted, it is essential to accurately identify and target at risk students by analyzing the intensive data within the university which relates to students demographic and performance. In this way, more time and effort can be given to high risk students who really need help from the service available and more appropriate assistances can be referred to students according to their real risk scores.

3.5.1 Data Source and Pre-process

There is more than ten years' of student historical data available, and every year the university enrolls roughly 10,000 new students, hence, there are 100,000 instances of student data available for analysis. There are various data sources that related to the student, such as:

- The Curriculum and Student System (CASS). The CASS data warehouse managed by Student Systems is a main data resource for this project. The data stored in CASS includes (and is not limited to) the students demographic details, the course/subjects the students selected and the final scores of the subjects.

- Allocate Plus. The Allocate+ system records different kinds of activities (e.g., lectures or workshops) which students took part in. All activities are related to certain subjects. Data in Allocate+ can be integrated into CASS, the interfaces run hourly and others daily.

- Assessment Data. The assessment data refers to the spreadsheets that contain students' assessment scores. Usually, the final score of a subject consists of different assessment scores, midterm test scores, and final exam results. The assessment score reflects if a student is going well in a subject in an ongoing way.

- Sanction Data. The student sanction data is submitted to CASS every semester. However, EIS (Early Intervention Scheme) records are not submitted to CASS since it is not regarded as a kind of sanction. The EIS records are only applied to international students to help them stay away from academic cautions.

- Online Data. Online data is an online system for students to enrol in subjects and to access subject information, or submit assessments online. Students are requested to enroll in a certain subject on Online service before they can actually study it. The coordinator of the subject can put up through Online service notices along with related materials/assessments. Students can read or download materials from the system, as well as submit assessments online. The system captures the students every move in this system.

- Business Intelligence (BI) Portal. The BI portal is designed and maintained by the Planning and Quality Unit (PQU). The BI portal integrates several data warehouses, and provides reporting and CUBE-based analysis (e.g. failure rates of subjects from different perspectives, student feedback surveys, etc) on them.

- Library. Library stores the records of students logging into the library system, downloading references related to certain subjects and borrowing books. The library records of students indicate if they are hard working or have good study habit.

- Housing data. The housing department stores housing data which reflects students accommodation status, e.g., accommodation types and debt status. This kind of data can then be used for social network analysis.

Figure 3.6 illustrates the data source and the structure of the system. After extract, transform and load (ETL) the data into a fact table, all the

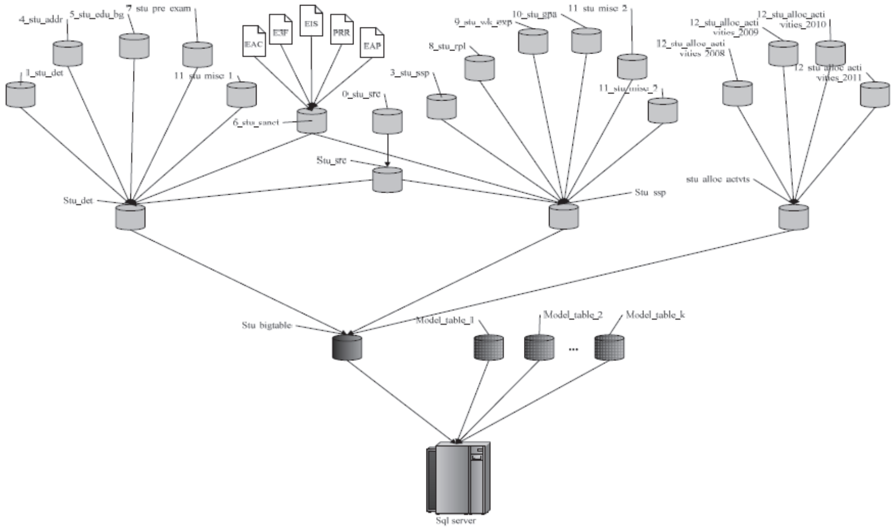


Figure 3.6: Data Source for Student Performance Prediction

features related to the student are accessible, and also have the label of historical data. The processed data is discretized to the features that fit the proposed method, and after data cleaning and feature selection 89 features were chosen to build the final model. The features not only include the demographic information about the student, but also cover the student behaviours and status. The demographic features include their nationality, previous educational background, previous academic grades, previous scholarship records and more. Moreover, the data was categorized by students' previous examination results, that is, if the student was in the top 30 percent of their peers, it was labeled as class A while others were labeled as B. The configuration of the experiment is the same as the aforementioned experiment. This experiment compared this novel method with other classic classification methods by contrasting the standard classification task metric result, and Table 3.2 shows that the proposed method has an advantage over most of the metrics. More importantly, the result supports our main statement about real world data, in that there are plenty of coupling relationships not only between the value of features, but also among the objects.

Table 3.2: Comparison on Student Data

Method	Precision	Recall	F1
C4.5	0.751	0.745	0.734
SVM	0.770	0.773	0.76
This Work	**0.801**	**0.785**	**0.776**

3.5.2 Classification Efficiency Comparison

Despite the high time complexity of calculating coupling similarity (n^2R^3, n is the number of instance in the data set, and R is the distinct value of the data set, respectively), and also the complexity of original k-NN is n^2, when dealing with the student performance prediction task at the size of 100,000 instance and 89 features, the processing time is extremely high. Though it has a superior classification performance, it still hard to apply to the real world system. However, with the proposed "Coupled Similarity based K-Nearest Centroid Classification" (CDK) the classification time is reduced significantly, and does not lose too much prediction performance. Figure 3.7 verified when the number of clusters in the proposed method drops down, the prediction performance does not fall too much. Even if only a few centroids have been selected, the prediction precision still outperforms the traditional methods.

In contrast, Figure 3.8 shows that the prediction time cost drops down dramatically when the number of clusters is reduced. Intuitively, without this strategy, the prediction time is unacceptable in this task. However, the time cost the of proposed method outperform the traditional method significantly; this disadvantage will be addressed in the later chapter. The two figures prove the aforementioned statement concerning the proposed method, performing well in the student performance prediction task.

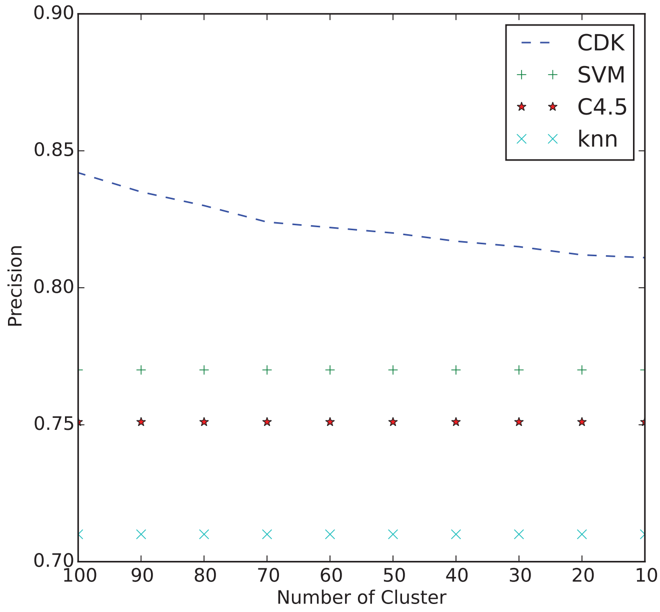


Figure 3.7: Precision on Different Number of Clusters

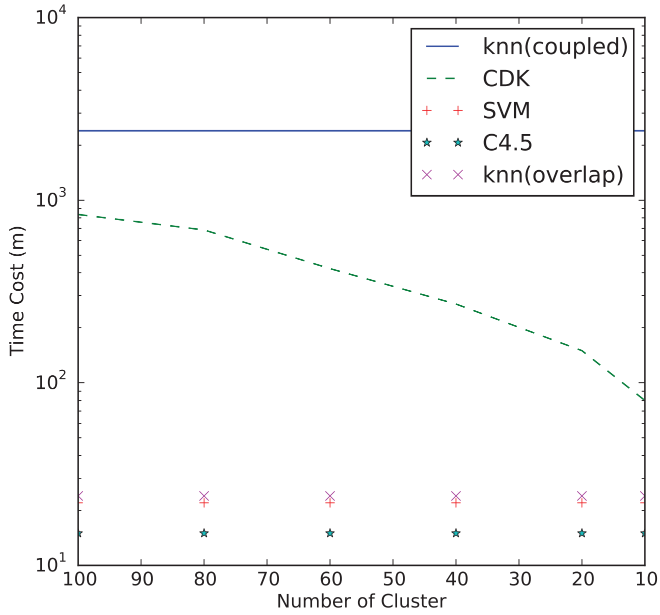


Figure 3.8: Time Cost on Different Number of Clusters

3.6 Summary

In this chapter, a coupling distance based K-nearest weighted centroid classification method has been proposed, and a coupling object similarity metric which involves both attribute value frequency distribution (intra-coupling) and feature dependency aggregation (inter-coupling) in measuring attribute value similarity for the classification of nominal data has been applied. Substantial experiments have shown that applied inter-coupling with relative similarity measures significantly outperform the other method without considering the coupling relations. In terms of efficiency, in particular with large-scale data, our method does not show its advantages, however, this thesis will introduce a large scale solution in one of later chapters.

Chapter 4

A Coupled Similarity Kernel based Pairwise Support Vector Machines for Student Performance Prediction

4.1 Introduction

After the first step of applying of the coupled similarity, a more sophisticated method must be integrated, the support vector machines with coupled similarity. The application purpose in this chapter is different from the previous chapter. This chapter serves as a foundation algorithm of "killer subject analysis" which finds the highest risk students in the killer subjects, those with highest fail rate. There are only five killer subjects that have been selected, and only a few hundred training samples. However, this issue requires a more accurate solution and the time costs are not critical for this task. The support vector machines is a supervised learning model with associated learning algorithms that analyze data and recognize patterns. In various applications, the SVMs shows its advantages in terms of classification performance. However, the original SVMs is designed for numerical data. As the SVMs runs on the nominal data, most previous research has used a certain number instead of each nominal value or it has transformed the nominal value into one hot vector. The one hot vector is a vector that has only one value of 1 while all others' values are 0. The position of the value 1 depends on the nominal value of the original data. Both methods cannot present the original nominal data's structure and the similarities between them, which leads to a loss of information from the data and reduces the classification performance. Moreover, the classic SVMs needs to find the hyperplane to do the classification, however, there is no such hyperplane that can be calculated in the categorical data sets. In this chapter, a novel coupled similarity metric between nominal attributed data is presented. Furthermore, because the metric of optimization is pairwise, the mathematical hyperplane is not necessary in the proposed pairwise SVMs. The experiment result shows the proposed method outperforms the traditional SVMs and other popular classification methods on various public data sets and also the student performance prediction task.

To measure similarity or distance between two data objects is an essential step for several data mining and knowledge discovery tasks. The direction of similarity for continuous data is relatively well explored, but for categorical data, the computation of similarity between the two data objects is not s-traightforward. Several data-driven similarity measures have been proposed in the research literature to compute the similarity between two categorical data instances, for example, clustering (K- means), distance-based outlier detection, classification (KNN), and several other data mining approaches. These algorithms typically treat the similarity computation as an unimportant step and can make use of the traditional measure directly.

For the continuous data, the *Minkowski Distance* is a widely accepted general method which can be applied to compute distance between two data objects with continuous attributes. There are two popular distance metrics, the *Manhattan Distance* and the *Euclidean Distance*. These two distance metrics are the particular case of the *Minkowski Distance* in order 1 and order 2 respectively.

On the other hand, the computation of the similarity of the categorical data is not as straightforward as when applied to continuous data. The key difference of the categorical data is that the values which an attribute takes are not ordered, hence, it is not easy to compare two categorical values directly. In this chapter, a novel coupling data object similarity metric based on the categorical attributes value frequency distribution and the attribute's value co-occurrence is proposed, and the proposed similarity metric is then used as the kernel for the pairwise SVM classification task. The most characteristic property of the proposed method is that the distance between two data objects is a related measurement; as the *Euclidean Distance* does not have a universal consistency, therefore, the pairwise SVM is the perfect solution to tackle this problem.

Some surveys (Boriah et al. 2008) (Gan et al. 2007) have discussed the similarities between categorical attributes. Cost and Salzberg (Cost & Salzberg 1993) proposed MVDM based on labels, while Wilson and Mar-tinez (Wilson & Martinez 1997) studied heterogeneous distances for instance based learning. These measures in their study are only designed for supervised approaches. The SVM is successfully used in a large number of applications, extending *Euclidean Distance* based SVM classifiers to wider range *Relative Distance* based classification. Several modifications have been suggested, for example the one against all technique, the one against one technique, or directed acyclic graphs, see Duan and Keerthi (Duan & Keerthi 2005), Hill and Doucet (Hill & Doucet 2007), Hsu and Lin (Hsu & Lin 2002), and Rifkin and Klautau(Rifkin & Klautau 2004) for further information, discussions, and comparisons.

For more machine learning task, there are some existing data mining techniques for nominal data (Ahmad & Dey 2007*a*, Boriah et al. 2008). The most famous are the SMS measure and its diverse variants such as Jaccard coefficients (Gan et al. 2007), which are all intuitively based on the principle that the similarity measure is 1 with identical values and is otherwise 0. More recently, attribute value frequency distribution has been considered for similarity measures (Boriah et al. 2008); neighborhood-based similarities (Ahmad & Dey 2007*a*) are explored to describe the object neighborhood by using an overlap measure. They are different from the proposed method, which directly reveals the similarity between a pair of objects.

This Chapter is organized as follows. In Section 2, there is a rough review of the related work. Preliminary definitions of the proposed coupled similarity metric for the categorical data are illustrated in the Section 3. In the next section, the formal definition of pairwise SVMs with the coupled similarity kernel will be given. In Section 5, the proposed algorithm is performed on several sets of public data and also the student analysis application data. Lastly, conclusions are drawn about the proposed method and indicate future research directions.

4.2 Problem Statement

In this section, unlike the previous chapter that uses the coupled similarity (Wang et al. 2011) directly, a *coupled similarity* for data objects is revised with categorical attributes which consider more relations between the attributes' values. This chapter aims to find a proper method which reveals the information from the categorical values in different attributes to identify the coupling relation between two data objects. The basic assumption is that the coupled similarity between two categorical values can be firstly illustrated by the frequency of the value, secondly that it can be aggregated by the co-occurrence with other values, and then the coupled similarity between two data object can be computed by engaging with each attribute's value. The formal definition is as follows:

Definition 4.1 *Given an set of categorical values $V = \{v_1, v_2, \ldots, v_n\}$, the* ***Categorical Value*** *(v_θ) is the θth value of the Attribute.*

Definition 4.2 *Given an set of Attributes $A = \{a_1, a_2, \ldots, a_n\}$, the* ***Attribute with Categorical Values*** *(a_γ) is the γth attribute of the data* *object.*

Definition 4.3 *Given an set of Data Objects $D = \{d_1, d_2, \ldots, d_n\}$, the **Data Objects with Categorical Attributes** (o_φ) is the φth object of the data set.*

In fact, the discrepancy of attribute value occurrence times reflects the value similarity in terms of frequency distribution. Thus, when calculating attribute value similarity, it considers the relationship between attribute value frequencies on one feature, proposed as coupling attribute similarity in the following.

Definition 4.4 *Given the previous definition, the **coupling intro similarity** between two values $v_{\theta 1}$ and $v_{\theta 2}$ within on attribute a_γ is:*

$$Sv^a_{Ia}(v_{\theta 1}, v_{\theta 2}) = \frac{|v_{\theta 1}| \cdot |v_{\theta 2}|}{|v_{\theta 1}| + |v_{\theta 2}| + |v_{\theta 1}| \cdot |v_{\theta 2}|} \tag{4.1}$$

where $|v_\theta|$ stands for the appeared times of the value v_θ within the attribute a_γ.

Meanwhile, not only the value within one attribute can be computed as a sophisticated similarity but also the values among different attributes have an impact on the similarity computation. The assumption is the same as the aforementioned theory, the co-occurrence of one pair of values among the attribute demonstrates exhibit a relationship between them.

Definition 4.5 *Given the frequency of two values $|v^{a_{\gamma 1}}_{\theta 1}|$ and $|v^{a_{\gamma 2}}_{\theta 2}|$, the **coupling inter similarity** can be defined as:*

$$Sv^a_{Ie}(v^{a_{\gamma 1}}_{\theta 1}, v^{a_{\gamma 2}}_{\theta 2}) = \sum_{x^{a1,a2}} \min\left(\frac{|\{x^{a1}\} \bigcap v^{a_{\gamma 1}}_{\theta 1}|}{|v^{a_{\gamma 1}}_{\theta 1}|}, \frac{|\{x^{a2}\} \bigcap v^{a_{\gamma 2}}_{\theta 2}|}{|v^{a_{\gamma 2}}_{\theta 2}|}\right) \tag{4.2}$$

where x^{a_γ} is one value of each attribute a_γ, and the equation iterate all the possible value of the two attributes.

With the definition of the **coupling intra similarity** and **coupling inter similarity**, a comprehensive **coupled similarity** between two data objects ban be summarized.

Definition 4.6 *Given the previous definition, the* ***coupled similarity*** *between two data objects $d_{\theta 1}$ and $d_{\theta 2}$ is:*

$$S(d_i, d_j) = \sum_{a_\gamma}^{A} Sv_{Ie}^{a_\gamma} \times Sv_{Ia}^{a_\gamma} \tag{4.3}$$

The computation iterate all the attribute of the data object, and conclude an empirical ***coupled similarity*** *measurement.*

Finally, the proposed **coupled similarity** metric will be used in the next section as the kernel for the pairwise support vector machines.

4.3 Pairwise SVM with Coupled Similarity

4.3.1 Coupled Similarity as a Kernel

The classic SVM uses a fixed format kernel function which maps the original data into a higher dimensional space, by doing this, the SVM tries to find an optimized solution to split the data into two classes. The most popular kernel function of the SVM is following:

$$Linearkernel: \qquad K(x, y) = (x, y)^d \tag{4.4}$$

$$Polynomialkernel: \qquad K(x, y) = ((x, y) + 1)^d \tag{4.5}$$

$$RBFkernel: \qquad K(x, y) = e^{\frac{-|x-y|^2}{2\sigma^2}} \tag{4.6}$$

$$Sigmoidkernel: \qquad K(x, y) = tanh(\rho(x, y) + c) \tag{4.7}$$

These kernel functions have been applied very successfully in plenty of applications; however, there still some issues in using these kernel functions. Firstly, these kernel functions are designed to deal with the numerical data, for categorical data they do not have adaptive strategies. Secondly, all functions have the parameters that are not included in the optimization process. The classic way is just setting a value for the parameters by the experts' experience or by using the grid parameter search to get the optimized solution by chance.

To prevent these problems, the proposed method has its advantages. The proposed **coupled similarity** is designed to reveal more information from the categorical values, as though the coupled similarity's computation process still has assumptions, it is better than the classic method that never concerned itself with the relations between the categorical values; furthermore, the measurement is based on the distribution of the categorical value and its co-occurrence, the metric is more objective than the grid searched result. In conclusion, this chapter used the proposed **coupled similarity** as the kernel function for the further SVM optimization process.

Definition 4.7 *Given the previous definition, the* ***coupled similarity kernel*** *between two data objects x and y is:*

$$k(x, y) = \sum_{a_\gamma}^{A} Sv_{Ie}^{a_\gamma} \times Sv_{Ia}^{a_\gamma} \tag{4.8}$$

4.3.2 Pairwise SVM

By the definition of the coupled similarity, the distance between two data objects is a relative value, which means it is hard to define a universal space for measuring the distance but fortunately, the concept of the pairwise classification (Brunner et al. 2012) is perfectly adapted to solve this problem. Pairwise classification determines if the two input examples are in the same class, instead of only considering one data object as belonging to which class. It has a particular advantage if only a subset of classes is known for training. For later use, a support vector machine(SVM) that is able to handle pairwise classification tasks is called pairwise SVM.

Pairwise classification is the task of predicting whether the examples a, b

of a pair (a, b) belong to the same class or to different classes. In particular, interclass generalization problems can be treated in this way. In pairwise classification, the order of the two input examples should not affect the classification result. To achieve this, particular kernels as well as the use of symmetric training sets in the framework of support vector machines are suggested.

Pairwise classification determines that the two input examples are in the same class, instead of only considering one data object as belonging to one class. It has a particular advantage if only a subset of classes is known for training. For later use, a support vector machines(SVM) that is able to handle pairwise classification tasks is called pairwise SVM.

Given two data objects, the x_i and x_j, in the training set, the pairs should be pre-defined (x_i, x_j) as belonging to the same class or not. Next, this thesis defines a function $F(x_i, x_j)$:

$$F(x_i, x_j) = \begin{cases} +1 & oneclass \\ -1 & notone \end{cases} \tag{4.9}$$

If the pair (x_i, x_j) belongs to the same class, it can be called a positive pair, on the other hand, if the pair (x_i, x_j) does not belong to the same class, it can be called a negative pair.

The pairwise classification aims to decide whether the examples of the pair (x_i, x_j) belong to the same class or not. This chapter uses the decision function F and the pair (x_i, x_j) must be defined in the training set as being **in** or **not in** the same class.

The common kernel function has been explored in the previous section, this section discusses the pairwise kernel $K : (X \times X) \times (X \times X) \to \Re$. In this chapter, the assumption is made that the pairwise kernel is symmetric:

$$K((a, b), (c, d)) = K((c, d), (a, b)) \tag{4.10}$$

Similarly to the coupled similarity definition, a pairwise kernel is defined as follows:

$$K((a,b),(c,d)) = k(a,c) \times k(b,d) \tag{4.11}$$

It is assumed that pairwise decision function is is also symmetric:

$$F(x_i, x_j) = F(x_j, x_i) \tag{4.12}$$

And by evolving all other pairs with the pairwise kernel, the decision function can be defined as:

$$F(x_i, x_j) = \sum_{(k,l)} \alpha_{kl} y_{kl} K((x_i, x_j), (x_k, x_l)) + \gamma \tag{4.13}$$

The classical SVM solves the problem as follows:

$$\begin{aligned} &\max(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} y_i y_j \alpha_i \alpha_j K(x_i, x_j) \\ &s.t. \sum_{i=1}^{n} \alpha_i y_i = 0 \\ &\alpha_i \geq 0 (i = 1 \ldots n) \end{aligned} \tag{4.14}$$

With the pairwise kernel (Brunner et al. 2012) defined a dual pairwise SVM:

$$\begin{aligned} &\min_{\alpha} G(\alpha) \\ &s.t. \qquad\qquad 0 \leq \alpha_{ij} \leq C \\ &\sum_{(i,j)} y_{ij} \alpha_{ij} = 0. \end{aligned} \tag{4.15}$$

where the

$$G(\alpha) = \tfrac{1}{2} \textstyle\sum_{(i,j),(k,l)} \alpha_{ij} \alpha_{kl} y_{ij} y_{kl} K((x_i, x_j), (x_k, x_l)) - \sum_{i,j} \alpha_{ij}$$

By using the pairwise support vector machines, the coupled similarity can be easily integrated into the classifier. In the later section, the performance of the proposed method in the student performance prediction task is illustrated. This section discussed how to define the **coupled similarity** as the kernel to be used by the next stage SVM, followed by applying the **coupled similarity kernel** into the pairwise SVM dual optimization problem. The next section will demonstrate the advantages of the proposed method by implementation and experimentation on various public data sets.

4.4 Experiment and Evaluation

The proposed method has been implemented based on the widely used SVM library LIBSVM. Because the LIBSVM is an open source package, it is easy to modify the code to add new functions. Firstly, there is the implementation of the **coupled similarity** on the categorical data, secondly, the optimization function is overridden into pairwise case. The experiment runs on the PC with Intel i5 CPU and 4GB memory.

In this section, not only will some UCI data be used to test the performance of the algorithm, but also the data sets from student data will be used to predict the students' exam performance. There are four UCI data sets for this experiment, more precisely, they are the sonar,hepatitis,horse-colic and SPECTF data sets. After executing the algorithm on these data sets, the proposed method(svm-p) was compared to the classic SVM with one hot vector transformation(svm-o) and frequency transformation(svm-f) of the categorical data, and the "Coupled Similarity based K-Nearest Centroid Classification" (CDK) was which proposed in the previous chapter also has been compared. The experiment setting is the same as the previous chapter and the evaluation criteria is also the same.

Precision

The experiment results (Figure 4.1) show that the proposed method outperforms both the traditional SVM with both one hot vector transform and the frequency transformation. Moreover, the proposed method also significantly boosts the classification precision in comparison to the "Coupled Similarity based K-Nearest Centroid Classification" that was outlined in the pre-

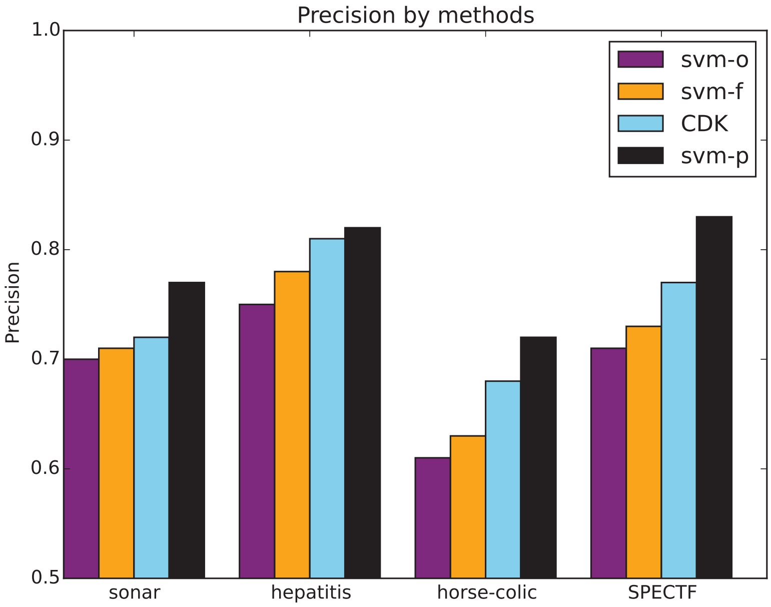


Figure 4.1: Classification Precision Comparison

vious chapter. This experiment shows the potential of the proposed method which can be applied to the student performance prediction task.

Recall

As described in the previous chapter, with the benefit of the coupling similarity, the recall(Figure 4.2) measurement increased significantly from the traditional SVM to the proposed pairwise SVM with coupled similarity as kernel. There is also a clear gap between the pairwise SVM and the CDK method that makes a further contribution to the student performance prediction task.

F1 measure

Again, as expected, the proposed method also show its advantage in the F1 measure of the classification comparison(Figure 4.3). In conclusion, the proposed method outperform the others methods in many ways. It is effective enough to be applied to the ”killer subject analysis” system that has an impact on the real world environment.

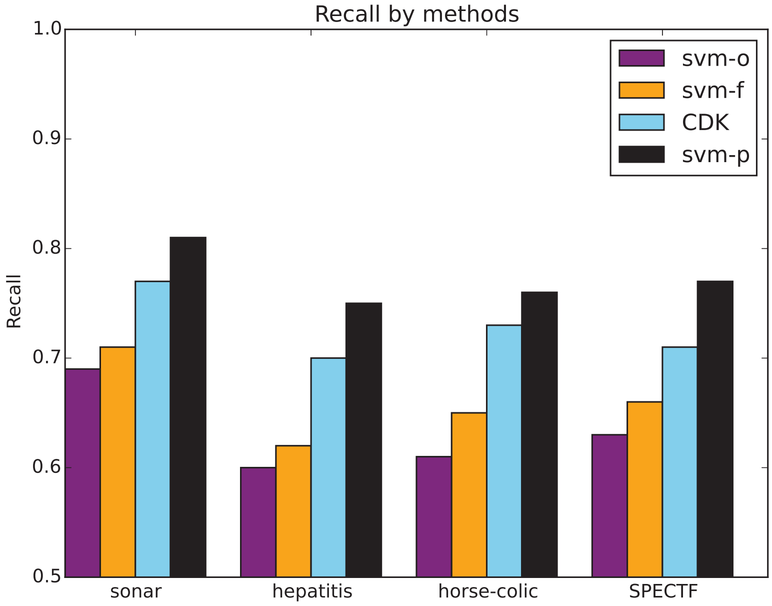


Figure 4.2: Classification Recall Comparison

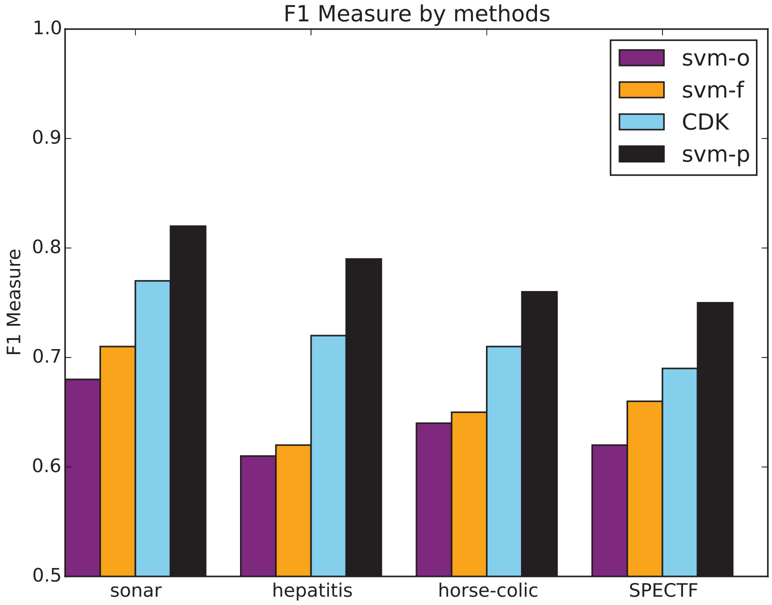


Figure 4.3: Classification F1 measure Comparison

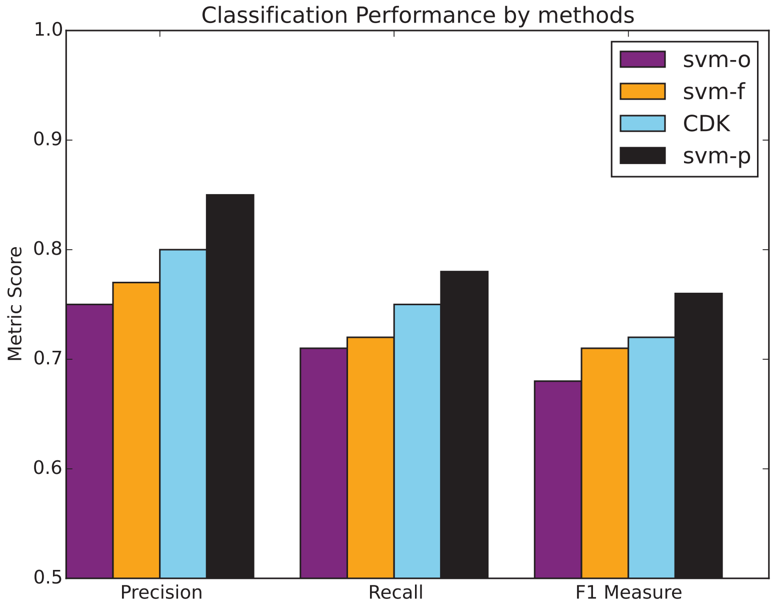


Figure 4.4: Subject Introduction to Electrical Engineering Classification Comparison

4.5 Application

After the experiment on the public data, this section evaluates the proposed method for using the student data from the five ”killer subjects”. The ”killer subjects” are in Electrical Engineering field: ”Introduction to Electrical Engineering”, ”Electronics and Circuits”, ”Fundamentals of Electrical Engineering”, ”Circuit Analysis” and ”Signals and Systems” respectively. They are called killer subjects because they have particularly high failure rates. For example, subject Circuit Analysis has a particularly high failure rate of around 70 percent. The other four subjects pass rates are also at a very low level, around 50 percent. Therefore, it is critical to find out who is most likely to fail in these subjects. This project has been conducted in the following steps to achieve the goals of:

- Data collection and integration; collecting all possible data resources

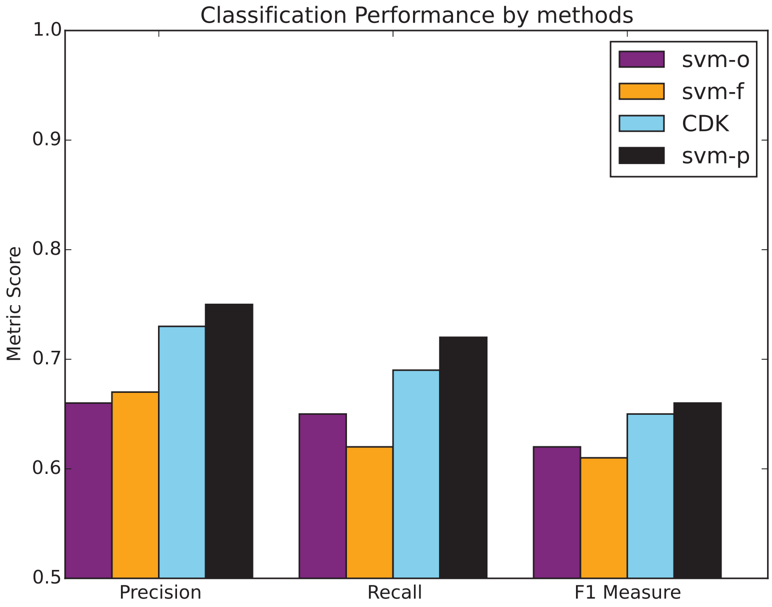


Figure 4.5: Subject Electronics and Circuits Classification Comparison

that relate to studentslearning performance and behaviours, and integrating them for further analysis.

- Basic analysis; conducting basic analysis on major data features to discover which factors affect students performance significantly and which factors are trivial. This analysis will provide a basic knowledge of all of the factors and will help in the design of more complicated analysis.

- Combined analysis; since it is impractical to find a single model that determines students performance, the application of combined analysis will produce more discriminative samples by a combination of separate models. By comparing the model of different subjects, key samples in common and distinct key samples for different killer subjects can be ascertained.

- System building; an application system is built to integrate the selected models. Based on the designed models, risk scores are calculated for

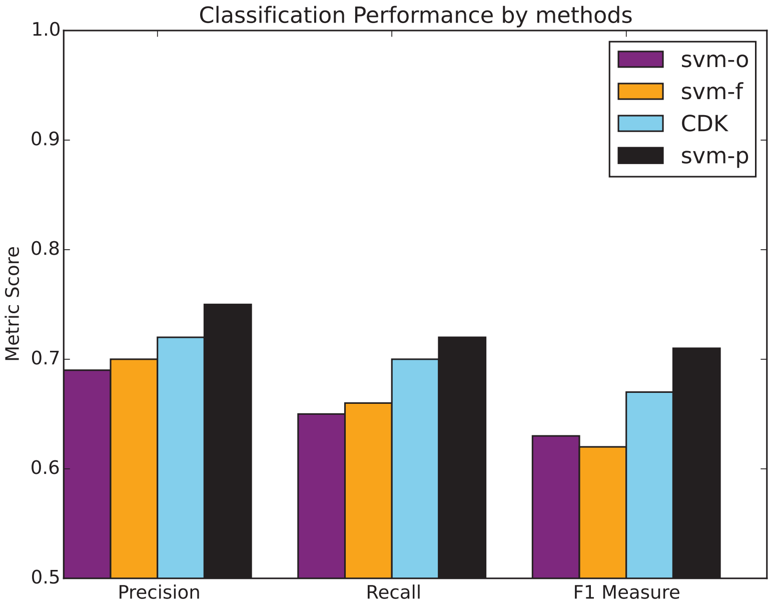


Figure 4.6: Subject Fundamentals of Electrical Engineering Classification Comparison

all students.

In this experiment, a students' demographic data set combined with the pre-academic assessment data is used; 1348 samples with 80 attributes who enrolled in the five killer subjects in the last three years for training, and most recently 142 students who enrolled in the five killer subjects for testing. The demographic attributes include their nationality, previous education background, previous academic grades, previous scholarship records and much more. The data was labeled by students last semester exam result, as to whether the student failed this subject. Information has been evaluated by two strategies. Firstly, the model was trained on each killer subject separately. In the second strategy, the was trained model by select the most representative samples from each subject and adding all of them to each training set as the new training set, and testing on the new coming student.

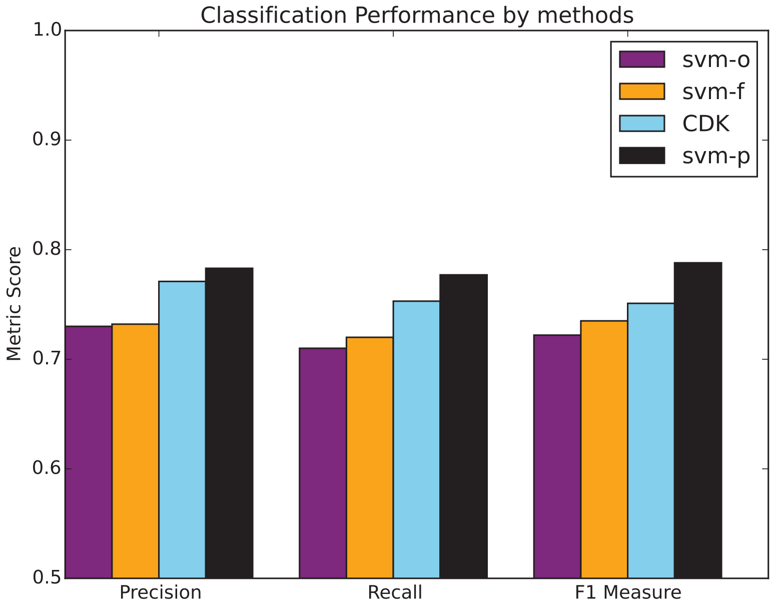


Figure 4.7: Subject Circuit Analysis Classification Comparison

The experiment wa performed on strategy one first, Figure 4.4 shows the proposed method for the subject ”Introduction to Electrical Engineering” outperforms the classic classification algorithms in all metrics. Figure 4.5,Figure 4.6,Figure 4.7,and Figure 4.8 indicate that the adaptation of the proposed method means that it performs at a superior rate for variations of the data sets.

The experiment was also performed under the second strategy. Figure 4.9 illustrates the advantage of the second strategy, as it boosts the classification performance for all the data sets. The ”svm-p1” denotes strategy one and the ”svm-p2” denotes strategy two. ”Subject 1”,”Subject 1”,”Subject 1”,”Subject 1”,”Subject 1” denotes the subject name of ”Introduction to Electrical Engineering”, ”Electronics and Circuits”, ”Fundamentals of Electrical Engineering”, ”Circuit Analysis” and ”Signals and Systems” respectively.

More importantly, the result supports the main assumption, that there are plenty of coupling relations among the categorical data, and that the

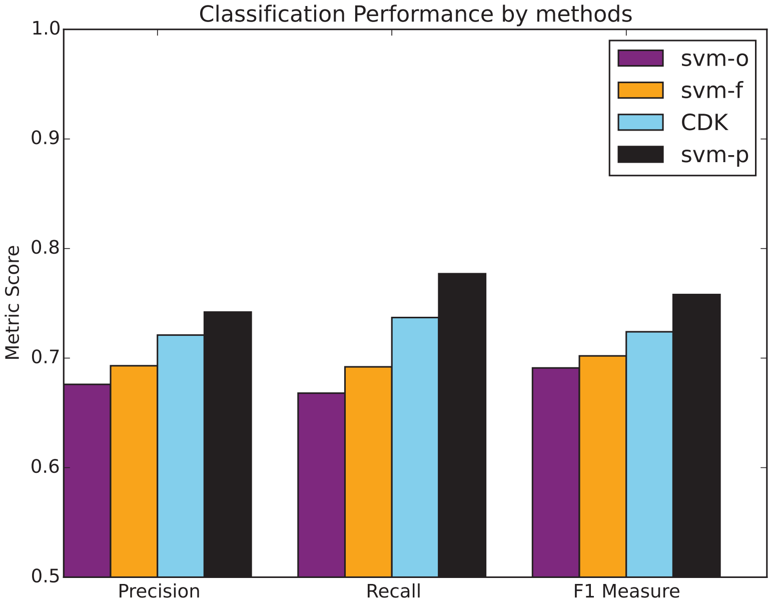


Figure 4.8: Subject Signals and Systems Classification Comparison

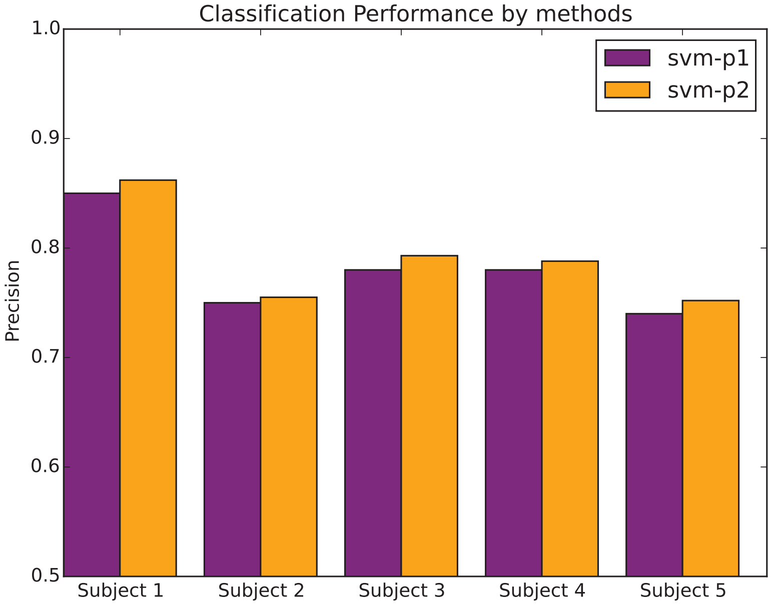


Figure 4.9: Two Strategies on All Subjects Classification Comparison

proposed method significantly improves the student performance prediction accuracy.

4.6 Summary

In this chapter, a novel pairwise classification method for the student performance prediction task was proposed. It involved the coupling relation between the categorical data and adaptively engaged with the pairwise SVM. Substantial experiment result have demonstrated the advantage of the proposed method. However, the efficiency of the proposed method is not good, and even though classic SVM is already time consuming, this method would be even more time consuming than that. Unfortunately, it cannot be extrapolated to a larger data set.

In the future, firstly, the formal time complexity could be estimated; after further analysis of the time complexity, a more efficient method could be developed, some caching or approximate method could also be applied.

Chapter 5

Extracting Coupling Relations from Term to Term for Student Social Media Sentiment Analysis

5.1 Introduction

The second object of this thesis is the student social media sentiment analysis. This chapter proposes a novel method that utilizes coupling relations from term to term for student social media sentiment analysis. With the rapid proliferation of social media and the growing online community, a significant amount of social media content data has been generated. Discovering the insightful value of the text data from the social media content, has increased its importance. Accordingly, a variety of text mining and process algorithms have been created in recent years such as classification, clustering, and similarity comparison. Most previous research uses the vector-space model for text representation and analysis, however, the vector-space model does not utilize the information about the relationships between terms. Moreover, the typical classification methods also ignores the relationships between text documents. In other words, the traditional text mining techniques assume the relation between the term to term and from the document to document are independent and identically distributed(iid). This chapter introduces introduce a novel term representation by involving the coupling relations from term to term. This coupling representation provides much richer information that enables the creation of a coupled similarity metric from document to document, as a coupling document similarity based K-Nearest centroid classifier has been applied to the classification task. Experiments verify that the proposed approach outperforms the classic vector-space based classifier and shows the potential advantages and richness in the student social media sentiment analysis tasks.

Text processing and mining methods have received increasing interest in recent years, because of a variety of applications such as social media, blogs, and online communities. The most general form of text data is strings, and the most common representation for text is the vector-space representation. The vector-space model represents the text in each document as a "bag-of-words". Though the vector-space representation is very efficient, it loses information about the structural information of the words in the document, especially when it is used purely in the form of individual word presentations.

In many applications, the "unordered bag of words" representation is insufficient for finding the analytical insights, especially in the case of fine-grained applications, in which the structure of the documents affect the underlying semantics. Intuitively, the advantage of the vector-space representation is that the simplicity lends itself to straightforward processing, however, the vector-space representation is very inaccurate because it does not include any information about the ordering of the terms in the document. Additionally, as it is shown in Figure 5.1 the vector-space model implements the term frequency inverse document frequency(TFIDF) of each term as the feature of one document and the discriminative power of this approach is not strong on the low frequency term because many low frequency terms share a relative same TFIDF value. In fact, most of the text documents are made by these

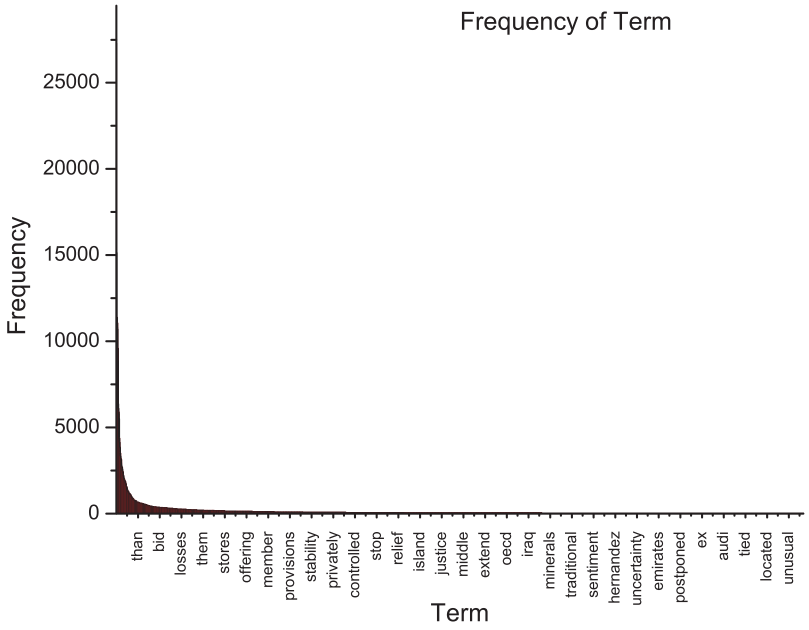


Figure 5.1: Term Frequency in Reuters-21578 R8

low frequency terms. Therefore, it is very hard to distinguish the similarity among those documents, regardless of the method used, either in using Euclidean distance or Cosine distance. For instance, here are some examples from the Reuters-21578 R8 data set:

noranda to spin off forest interests into separate company

burlington northern inc st qtr shr profit cts vs loss dlrs

api says distillate stocks off mln bbls gasoline off crude up

Clearly, the three documents have a different topic. However, the vector-space model representation lacks of the ability to distinguish them from each other, because many terms's TFIDF value in these documents is relative low and similar.

To prevent the aforementioned weakness of the previous research, the key contributions of this paper are as follows:

- Firstly the proposal of a novel method that capture the order information of the terms by aggregate information from the term's most co-occurring neighbourhood.

- Secondly the building of a vector presentation for each term instead of the scalar value which used in the traditional vector space model. By doing this, a much richer contest information for document similarity comparison is involved. It is defined as the coupled similarity between document.

- Thirdly the implementation of a novel classification method by involving the coupled similarity among each document.

The research to explore term's representations has a long history; early proposals can be found in (Hinton & Sejnowski 1986)(Pollack 1990)(Elman 1991)(Deerwester, Dumais, Landauer, Furnas & Harshman 1990). Recently, many models have been proposed for involving the information given by the "next" word to given words. For instance, it has been explored in approaches that are based on learning a clustering of the words: each word is associated deterministically or probabilistically with a discrete class, and terms in the same categories are in the same respect.

The concept of using vector-space representation for terms in information retrieval also has been researched, where feature vectors for words are learned on the basis of their probability of co-occurring in the same documents(Hofmann 2001). An important difference is that in this chapter looks for a representation of terms that is helpful in representing Non-iid relations between each other separate to their context. This is similar to what has been done with documents for information retrieval with LSI (Papadimitriou, Tamaki, Raghavan & Vempala 1998). The idea of using a continuous representation for words has however been exploited successfully by in the context of an n-gram based statistical language model(Bellegarda 2004), using LSI to dynamically identify the topic of discourse.

Another main research area is using the neural networks to create representation of the terms. These models were first studied in the context of feed-forward networks(Bengio, Schwenk, Senécal, Morin & Gauvain 2006), and later in the context of recurrent models(Mikolov, Karafiát, Burget, Cernockỳ & Khudanpur 2010)(Mikolov, Kombrink, Burget, Cernocky & Khudanpur 2011). The use of distributed topic representations has also been studied in(Hinton & Salakhutdinov 2006)(Hinton & Salakhutdinov 2011), and (Bordes, Glorot, Weston & Bengio 2012) proposed a semantically driven method for obtaining word representations, (Cheng, Miao, Wang & Cao 2013) underling an implicitly semantic relation into the traditional measure based on the co-occurrence frequency.

Meanwhile, there are many other works showing that the word vectors can be used to improve many NLP applications(Collobert, Weston, Bottou, Karlen, Kavukcuoglu & Kuksa 2011)(Turian, Ratinov & Bengio 2010)(Collobert & Weston 2008). Estimation of the word vectors itself using different models on various corpora(Mnih & Hinton 2008)(Huang, Socher, Manning & Ng 2012). The word vectors are very useful for future research and comparison, but as most of the previous research use independent and identically distributed assumption(Cao & Philip 2012), this thesis proposes a novel method, aiming to capture the coupling relationships not only on a term to term level but also on a document to document level. The coupling relation has been explored in structured numerical data(Wang, She & Cao 2013*a*)(Wang, She & Cao 2013*b*), but only a few researchers talk about the coupling relation in the text mining task(Cheng et al. 2013). This work attempts to create a comprehensive framework of involving the coupling relations in the document classification task.

This chapter is organized as follows: the next section, explores the recent research on text representation and mining method. Section 3 illustrates the details of neighbourhood co-occurring based text term representation approach is . Section 4 discusses the coupled similarity metric and the novel classification method. Section 5 explains the experiment result. The conclusions and summary are presented in section 6.

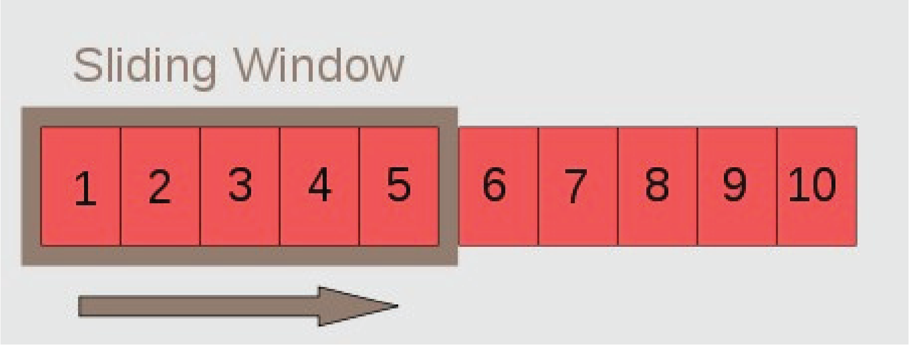


Figure 5.2: Sliding Window

5.2 Vector Presentation of Terms

This thesis has developed a novel text representation method describing the term's feature in a vector. The vector contains information about the term's IDF attributes as other methods produce, but also considers the impact of the other terms. The coupling relationships between term to term have been defined from two perspectives: one is the intra-coupling relationships which directly describes the term's feature by its most co-occurred neighbour's IDF value; while another is inter-coupling relationships, which involve the indirect information, when the two terms share some neighbourhood. Moreover, the coupling relation can be described in many ways; in this thesis one possible way has been explored. The next two subsections illustrate the definition of intra-coupling and inter-coupling relationships respectively.

5.2.1 Intra-Coupling Relation

Direct relations between term to term means the relations can be captured from the physical appearance of the terms in the documents. It is reasonable that there is some kind of relationship between two terms when they appear as neighbours. The proposed method is straightforward, in Figure 5.2 demonstrates a sliding window with width w, the middle block is m, and it also contains left blocks L and right blocks R. The term in the middle block is the key that is focused on, aiming to capture the impact from its neighbours. After initialization, the sliding window traversed all the text content, as a consequence, every term's neighbour can be discovered and the frequency can be calculated as well.

Definition 5.1 *A set of terms $T_\phi = \{t_1, t_2, \ldots, t_n\}$ is a related terms to certain term t_c if the terms in T co-occurred within the width w of the sliding window. There is a mapping function $F_\phi(t_c) = T_\phi$ to find the correlated terms of t_c.*

The ϕ is defined by the different relation criterion, so the mapping function F_ϕ could be any format. The simplest way is to rank the frequency of the neighbours for a certain term t_c. This thesis uses the neighbourhood based relation criterion. In detail, the top n ranked high frequency neighbor of t_c is selected, and their relationship is built by choosing the neighbours on the left and right side separately, and joining them all together. The mapping function F_ϕ becomes F_l,F_r,$F_{l\&r}$, and the related terms set T_ϕ becomes T_l,T_r,$T_{l\&r}$ respectively.

Assuming t_c as the center term, F_l overrides F_ϕ which finds that the most frequent terms appear as the prefix of the t_c within w, while the $F_r(t_c)$ stands for terms appearing in as the suffix of t_c within w. The traverse procedure with the purpose of finding the highest frequency neighbour terms T_ϕ of the center term t_c

The terms sharing the same meaning would have the relative same neighbours based on this selection because the same meaning term should have the relative same neighbours. At this stage, each term can be described by itself and its top ranked highly related neighbours, more precisely, the IDF is used as the base numerical measure of each term, each high related neighbor's IDF values combined as a vector to describe a term t_c. For instance, set $n = 2$ chooses the 2 most revelent terms and the IDF value is defined as $IDF(t_{\phi_n})$ where ϕ is the relating method. After that, t_c can be represented as a vector $V_{t_c} = [IDF(t_{l_1}), IDF(t_{l_2}), IDF(t_x), IDF(t_{r_1}), IDF_{t(r_2)}]$ instead of just a single scalar representation $IDF(t_c)$.

Table 5.1: Most Related Terms of the Key-Term

left 2	left 1	key-term	right 1	right 2
credit	bank	guarantee	lead	intern
bid	offer	comparison	Illinois	year
affecting	of	liquidity	the	in
with	the	proxy	materials	statement
in	raw	material	sciences	on
subject	to	regulatory	approval	approvals

After determining the most relevant terms to every term t_c, the vector representation V_{t_c} of the term t_c are a available simultaneously. For simplicity, the definition of the vector presentation $V_{t_c} = \{v_1, v_2, \ldots, v_n\}$,$n$ is the number that stands for top n high frequency neighbor, and $v_x = IDF_x$. As is shown in Table 5.1, the key-terms in the table are all in a same range of IDF value, which means that they cannot be distinguished from each other by the traditional method. However, with the proposed method, each term has it's unique neighbourhood and, by transforming the IDF into the vector presentation V_{t_x}, it is easy to distinguish every term from one to another. Also, if two terms share the similar neighbour, they have similar contexts and they can be judged as the synonyms. With this advantage, many applications can capture more semantic meaning of each term instead of using a scalar IDF value as the only feature for each team.

The V_t is a vector which includes the IDF values of t_c most related terms, and v_c is the IDF value of t_c. $\gamma(t_c)$ is used to involve the information from related terms of t_c and $\gamma(t_c)$ also is a vector.

$$\gamma(t_c) = \frac{1}{nw} V_c / v_c \tag{5.1}$$

where n is the number of top ranked terms, and w is the sliding window size, dividing by them aims at normalization.

Finally, with the normalization function $\gamma(t_c)$, the direct coupling relation called intra-coupled similarity between two terms can be defined as follows:

$$\delta^{Ia}(t_i, t_j) = \frac{\gamma(t_i) \cdot \gamma(t_j)}{\sqrt{\gamma(t_i) \cdot \gamma(t_i)} \cdot \sqrt{\gamma(t_j) \cdot \gamma(t_j)}} \tag{5.2}$$

5.2.2 Inter-Coupling Relation

The intra-coupling relation is getting the information from the most co-occurring neighbourhood directly. However, there are many terms, which have relations, but do not occur in the sliding window. In this section, a novel method is proposed to detect the relations that are not directly related.

The inter-coupling relation analysis is inspired by the fact that terms with the similar sense must appear in a similar context. Therefore the relation between a pair of terms by their most related terms has been explored, for simplicity using the most co-occurring neighbourhoods as examples.

As is shown in Figure 5.3, the $Term1$ and $Term2$ do not have the direct connection which means they never co-occurred within the sliding window, however, it may find some indirect relations by its direct related terms. $Term1$ has some direct related terms while $Term2$ also has some, actually they share some direct related terms T_s, therefore, the indirect relation can be captured by accumulating all their shared direct related items. A threshold of ϵ was set to restrict the minimal number s of share terms to ignore the too small impacts of its directly related terms. The inter-coupled similarity between two terms is defined as follows:

$$\delta^{Ie}(t_i, t_j) = \begin{cases} \hfill 0 & s < \epsilon \\ \displaystyle\sum_{t_s \in T_s} \delta^{Ia}(t_i, t_s) \cdot \delta^{Ia}(t_s, t_j) & s > \epsilon \end{cases} \tag{5.3}$$

The inter coupling relations involved richer context information for a term to term similarity than the traditional method. This is a fundamental criterion and can be applied to many applications as classification and clustering. The next section will discuss the document classification task based on the intra-coupling and inter-coupling relations.

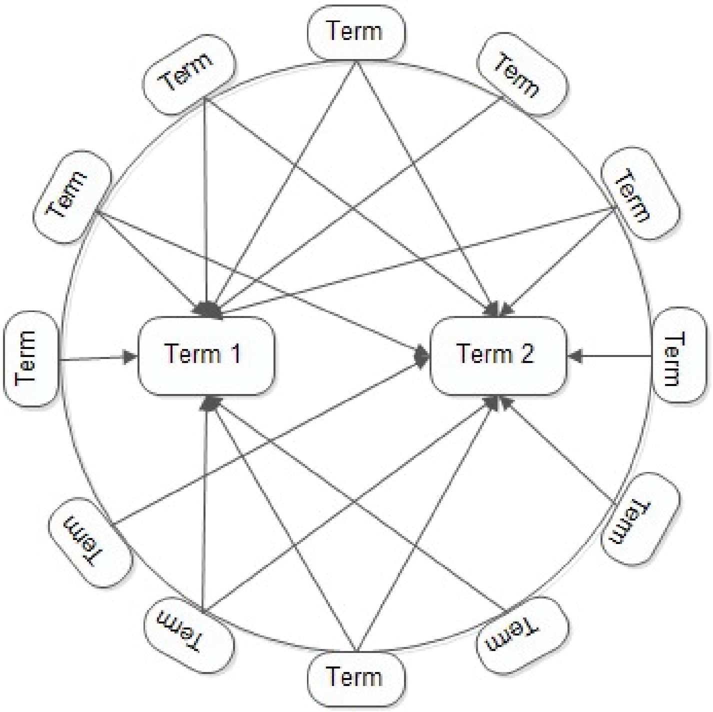


Figure 5.3: Inter Relation from Term to Term

5.3 Coupled Similarity based Document Classification

With the aforementioned definition of the coupling relations between term to term, a comprehensive coupled similarity term to term paradigm Cst is defined as follows:

$$Cst(t_i, t_j) = \alpha \delta^{Ie}(t_i, t_j) + \beta \delta^{Ia}(t_i, t_j) \tag{5.4}$$

α and β are the parameters for adjusting the weight for the importance of intra and inter coupling similarities. After defining the coupled similarity between term to term, a new criterion for the coupled similarity measurement for the document to document should also be defined. The relation between terms is captured by their coupling term to term similarity across the two documents $D = \{d_i, d_j\}$ which need to be compared. As a consequence the document-term matrix W is utilized which contains the coupled similarity Cst for each pair of terms shown in the documents. The coupled similarity between two documents by using the corresponding kernel is expressed as:

$$Csd(d_i, d_j) = \overrightarrow{d_i} W \overrightarrow{d_j}^T \tag{5.5}$$

where $\overrightarrow{d_i} = tfidf(t_1, d), tfidf(t_2, d), \ldots, tfidf(t_n, d)$ is the vector space model presentation of the document.

Upon formulation of the coupled similarity of the documents, the classification task is straightforward. If the coupled similarity is used of the document as the distance between two documents, the KNN algorithm as the classification method can be applied. Moreover, the traditional KNN is adapted to the planed data set. Firstly, the classification task is multi-label classification. Secondly, the computational cost of comparing two documents is high, therefore the comparison time should be made as low as possible. This can be done by finding the most λ representative document from each class in the training set. A centroid document m_t of one class c_t is a document within the c_t which has maximal similarity to all other documents within the class, for any document d' in c_t, the centroid document m_t satisfy:

$$\sum_{d_i \in c_t} Csd(m_t, d_i) \leq \sum_{d_i \in c_t} Csd(d', d_i) \tag{5.6}$$

where $\{c_t\} = \{d_{t1}, d_{t2}, \ldots, d_{tn}\}$ is the class which contains some documents. After choosing the most λ representative documents into T_λ for each class, the classic KNN algorithm is run on the set T_λ to perform the classification task.

5.4 Experiment and Evaluation

The purpose of this section is to illustrate the advantage of utilizing vector representation of terms. This will be achieved by extracting the coupling relation among terms. Both intra-coupling and inter-coupling relations will enhance the performance of the classification task, and the future optimization of many algorithms is possible; this issue is considered in the next section. The application of classification does provide qualitatively effective results. Furthermore, the use of different kinds of applications show that these results are not restricted to a specific data source, but can achieve effective results over a wide variety of applications.

All experiments were performed on a quad core Intel I5 CPU, 4G memory, and running windows 7 enterprise. All algorithms were implemented in $C\#$ and running on the .Net Framework.

5.4.1 Public Data Sets

The experiments as performed on some public data sets show the advantage of the proposed method. Three popular data sets used in traditional text mining and information retrieval applications were used in the experimental studies: (1) 20 newsgroups (2) Reuters-21578 and (3) WebKB. Furthermore, the Reuters-21578 data set is of two kinds: Reuters-21578 R8 and Reuters-21578 R52 so there are four different data sets in total. The 20 newsgroups

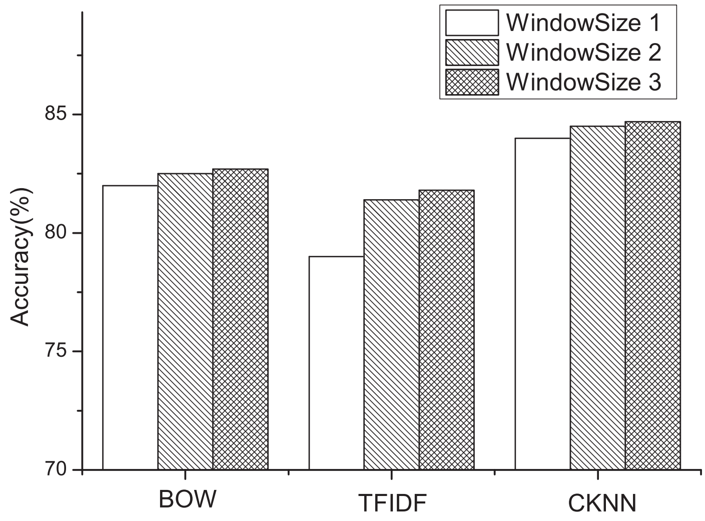


Figure 5.4: Classification (20 Newsgroup)

data set includes 20,000 messages from 20 Usenet newsgroups, each of which have 1,000 Usenet articles. Each newsgroup is stored in a directory, which can be regarded as a class label, and each news article is stored as a separate file. The Reuters-21578 corpus5 is a widely used test collection for text mining research. The data was originally collected and labeled by Carnegie Group, Inc. and Reuters, Ltd. in the course of developing the CONSTRUE text categorization system. Due to the fact that the class distribution for the corpus is very skewed, two sub-collections: Reuters-21578 R52 and Reuters-21578 R8, are usually considered for text mining tasks. These experimental studies make use of both of these two data sets to evaluate a series of different data mining algorithms. Every document in the aforementioned data sets is in a raw format, not having been preprocessed by eliminating non-alphanumeric symbols, specialized headers or tags and stop-words,and also not having been stemmed. The proposed methods are defined with respect to this never processed representation.

5.4.2 Experiment

This section first tests the effectiveness of the coupled similarity on a variety of classification algorithms. Our algorithm reads and indexes text documents and builds the statistical model. Then, different text classification algorithms are performed upon the statistical model. Three different algorithms were used for text classification. These algorithms are the Naive Bayes based on bag of words model, SVM classifier based on TFIDF vector representation of documents and the proposed coupled similarity based K-Nearest centroid classifier respectively. For each classification method of interest, the vector-space models were employed including unigram, bigram and trigram models, and the sliding window size ranging from 1 to 3, respectively, as the underlying representational models for text classification. In order to simulate the behaviors of the bigram model and the trigram model, the most frequent 100 doublets and triplets from the corpora were extracted and augmented with such doublets and triplets, respectively. The vector-space models are therefore further categorized as unigram with no extra words augmentation, bigram with doublets augmentation and trigram with triplets augmentation. 5-fold cross validation for each algorithm was conducted in order to compare the classification accuracies derived from different representation strategies. All the reported classification accuracies are statistically significant with a 95% significance level.

Figure 5.4 illustrates the classification accuracy results in the 20 newsgroups data set for the three different classifiers. In addition to the vector space representations for unigram, bigram and trigram models(for simple display,these models are also named as window size 1 to 3 respectively), the classification results for the coupling distance representations with different sliding window size ranging from 1 to 3 have also been illustrated. It is clear that the addition of neighbourhood information in the coupled similarity models improves the quality of the underlying result in most cases. Specifically, the best classification results are obtained for sliding window size 3 in K-Nearest centroid classifier.

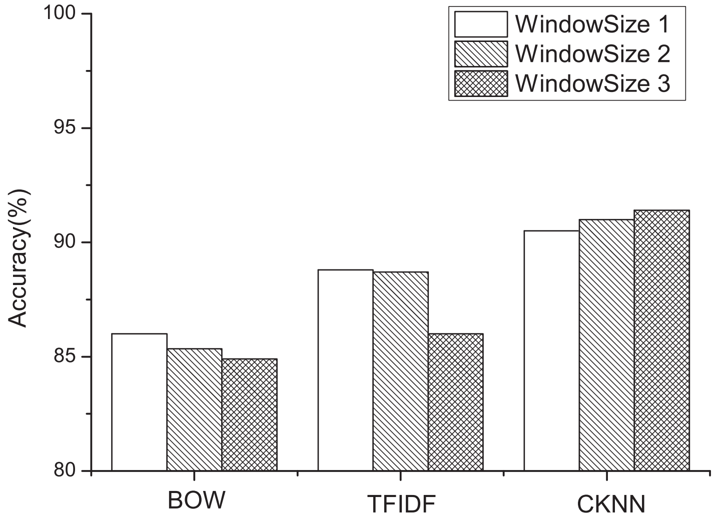


Figure 5.5: Classification (Reuters-21578 R52)

Meanwhile, in all of the cases, the coupled similarity models consistently obtained better classification results than all the traditional vector-space models, including the unigram, the bigram and the trigram models. Even though the optimal classification accuracy is achieved for coupled similarity model with sliding windows size of 1 and 2 in some experimental scenarios, it is noteworthy that the vector-space representations did not even perform better than the bigger sliding window size of coupled similarity model in all cases.

The classification results for the Reuters-21578 (R8 and R52) data sets were also tested. The classification accuracy results are illustrated in Figure 5.6 and Figure 5.5, respectively. It is evident that the coupled similarity based Nearest K-centroid classifier is able to provide a higher classification accuracy over the different kinds of classifiers as compared to the vector-space representations. The reason for this is that the both intra and inter coupled similarities can capture structural information about the documents which is used in neighbourhoods to help improve the classification accuracy. As a

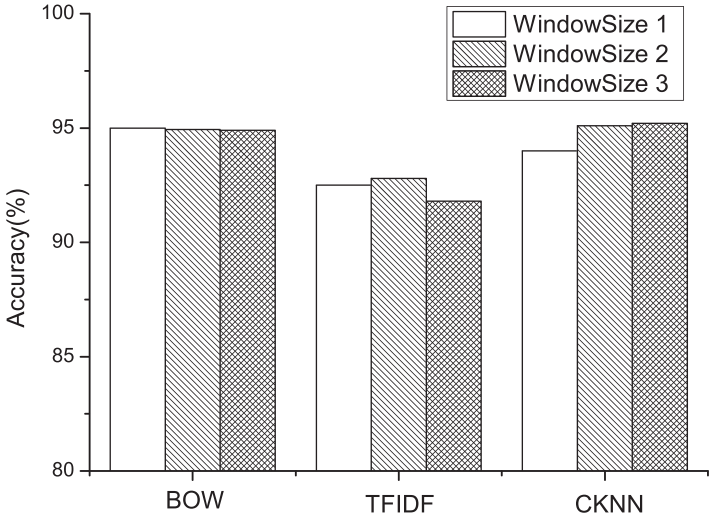


Figure 5.6: Classification (Reuters-21578 R8)

result, the classification results obtained with the use of the coupled similarity models are superior to those obtained using the vector-space representations.

The previous part of this section discussed the effective of accuracy between the traditional method and proposed method. Moreover, there are many parameters that can be adjusted to fine tune the performance. As Figure 5.7 shows, different sliding window types were tried on different window size. The optimal window type is left(L) and the right(R) neighbourhood of the center term, as discussed in Section 3. Moreover, the union set of L and R ($L + R$) and the joined set ($L\&R$) has been experimented with. For window size, the peak point for different window type is uncertain, but most of the best performance is around 3 to 4, which is why 3 was tried as the maximum window size in the previous experiment. The number of nearest neighbourhood in the KNN algorithm also has an impact on the the final performance, but as it is most related to the KNN algorithm itself, it is discussed it in this chapter.

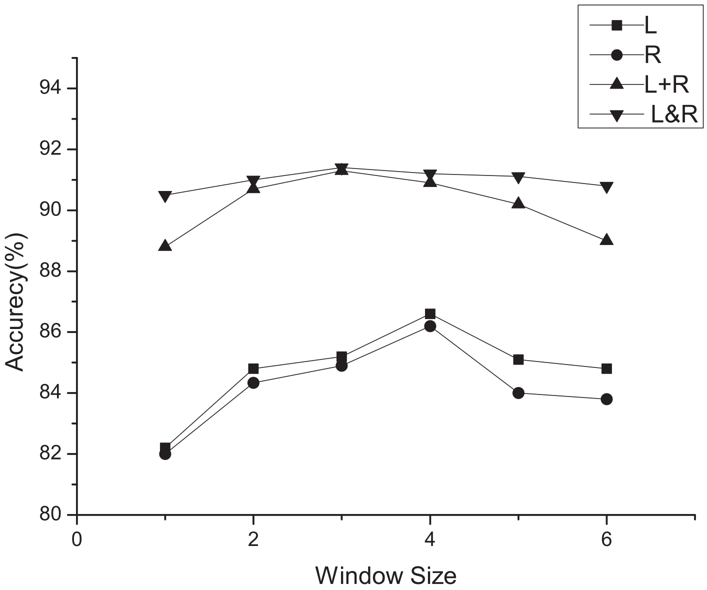


Figure 5.7: Different Window Type

5.5 Application

After evaluating the proposed method on the public data set, the experiment's positive result gave cause to run the proposed method on the student sentiment analysis task. The scenario of the student sentiment analysis in this thesis is very simple and the detail is described in Figure 5.8. Text content from the social media with sentiment label ":)",":(" for positive and negative sentiment respectively was collected. The social media data from the public content is very easy to get by crawler so 3 million posts with sentiment label from the social media like Twitter, Face Book, and Instagram was collected. The model was then trained by the proposed method. The evaluation for the social media data set can only be done in the public data set, because the incoming student posts following any university facilities are without label. After evaluating the proposed method as applied to the public data set with labels, and this experiment implicitly illustrated the sentiment analysis performance on the label free data sets. The experiment has been performed

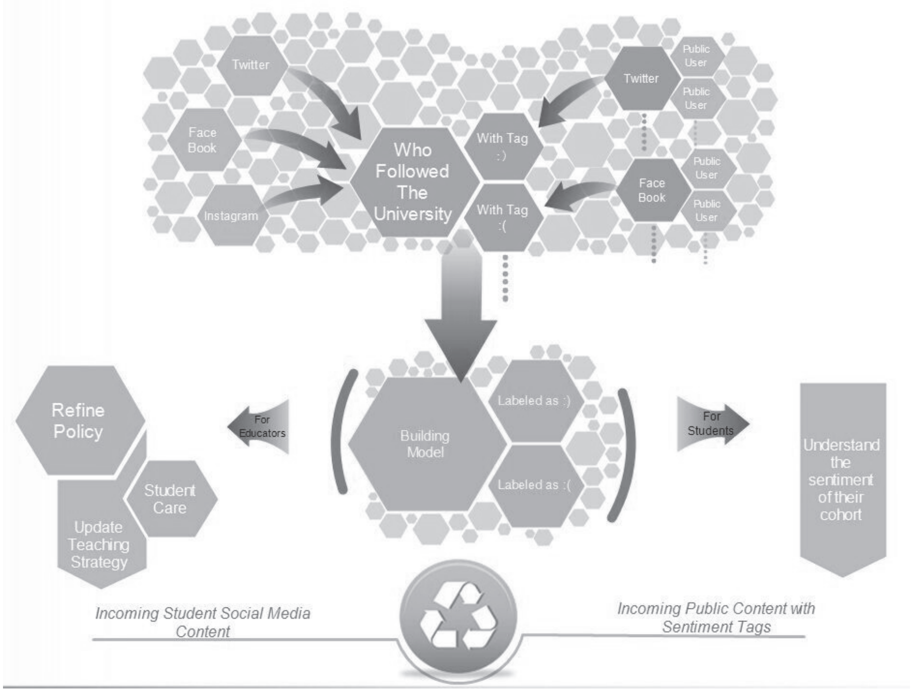


Figure 5.8: The Data Flow of Student Sentiment Analysis

under the same configurations of the experiment on the public data set and Figure 5.9 expresses the performance on the labeled public data. The result also verified that the proposed method outperforms the traditional methods.

5.6 Summary

This Chapter introduced the concept of coupled similarity from term to term based on its co-occurrences neighbourhoods in a sliding window, a new paradigm for text representation and processing. The coupling relation maintains information about the relative placement of words in regard to each other, and this provides a richer representation for document classification purposes. This similarity can be used in order to exploit the recent advancements in text mining algorithms. Furthermore, the coupled similarity criterion can be used with minimal changes to existing data mining algorithms if desired. Thus, the new coupled similarity framework does not require addi-

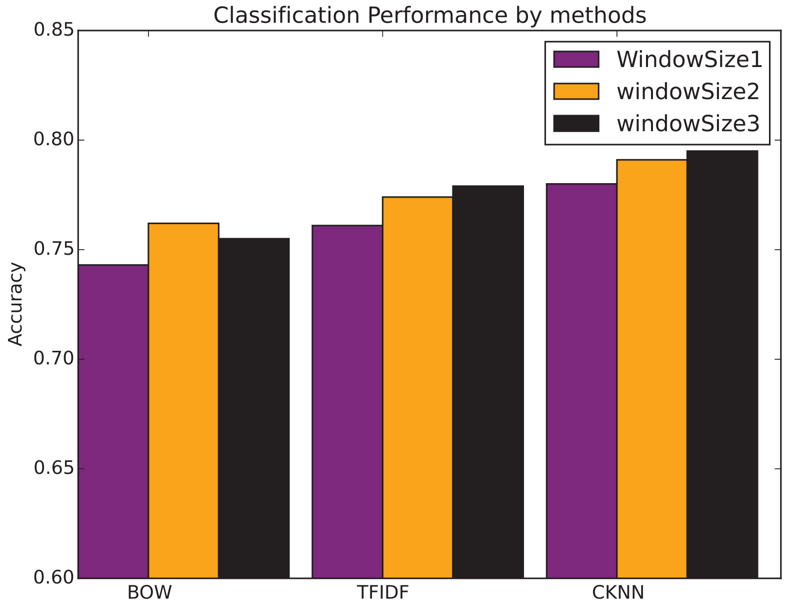


Figure 5.9: Classification Performance of Student Sentiment Analysis

tional development of new data mining algorithms. This is a huge advantage since existing text processing and mining infrastructure can be used directly with the coupled similarity based model. This chapter tested the proposed approach with a large number of different classification applications. The results suggest that the use of the coupled similarity provides significant advantages from an effectiveness perspective.

Chapter 6

Large-Scale Coupling Analysis For Educational Data

6.1 Introduction

Finally, due to the complexity of the proposed algorithms and the enormous size of the educational related data source, a scalable educational data analysis platform with the ability of coupling analysis is vital. During the past decades, the data sizes have grown faster than the speed of processors. In the same time, the capabilities of statistical machine learning methods are limited by the computing time rather than the sample size, as used in classification or clustering. Qualitatively different performances are revealed by the same strategies in the case of small-scale and large-scale learning problems. The large-scale case needs an optimization for the traditional algorithm to satisfy the huge computational complexity or else it can not be processed. Another issue with the state-of-the-art machine learning algorithms used on big data is that they rarely consider the relationship between the attributes and simply assume that they are independent of each other (IID), namely that data is independent. In the real-world the attributes interact via explicit or implicit relationships, which respect the data structures and relationships embedded in data objects and features. This chapter propoes with the help of the Spark cluster computing framework, a novel parallelized coupling similarity-based learning approach to cater for the big data learning problem by parallelizing the similarity calculation process, and also considering the coupling relationship between attributes and parallelizing the process. The approach has been tested on several data sets, using both on classification and clustering problems. Compared to the original algorithms and the variants, the experimental results show that our supposed method not only outperforms the clustering and classification performance of the baselines, but gets a huge improvement on the data scale and the time cost. Finally, when the educational data applications are run in this new platform, the time consumption is reduced significantly.

The computational complexity of the learning algorithm becomes the critical limiting factor when the very large data sets are released. Decades ago, a single gigabyte of data meant a vast amount of information, but now, data stored in terabyte or petabyte, or even the exabyte or the yottabyte is common, which is a trillion terabytes are used. The explosion of social networks, cloud computing and new technologies has given rise to incomprehensibly large worlds of information, often described as ”Big Data”. This poses new questions and challenges, especially for computational abilities.

The growth of cluster computing systems and cloud computing facilities provide one solution. Cluster computing systems provide storage capacity, computing power and high-speed local area networks to handle large data sets. In conjunction with new forms of computation combining statistical analysis, optimization and artificial intelligence, researchers are able to construct statistical models from large collections of data to infer how the system should respond to new data.

As the deluge of data grows, a key question is how to make sense of the raw information. How can researchers use statistical tools and computer technologies to identify meaningful patterns of information? How shall significant correlations between data be interpreted? What is the role of traditional forms of scientific theorizing and analytic models in assessing data?

Along with the growth of data, a problem called ”the Curse of Dimensionality” arises. It is a term coined by Bellman (Bellman, Bellman, Bellman & Bellman 1961) to describe the problem caused by the exponential increase in volume associated with adding extra dimensions to a mathematical space. One implication of the curse of dimensionality is that some methods for the numerical solution of the Bellman equation require vastly more computer time when there are more state variables in the value function. For example, 100 evenly-spaced sample points suffice to sample a unit interval with no more than 0.01 distance between points; an equivalent sampling of a 10-dimensional unit hypercube with a lattice with a spacing of 0.01 between adjacent points would require 1020 sample points: thus, in some sense, the 10-dimensional hypercube can be said to be a factor of 1018 “larger” than the unit interval.

When the dimensionality d increases, the volume of the space increases so fast that the available data becomes sparse. In order to obtain a statistically sound and reliable result, e.g., to estimate multivariate functions with the same accuracy as functions in low dimensions, it requires the sample size n to grow exponentially with d. This will cause a huge computational burden for the traditional learning methods and undoubtedly will decrease the efficiency of the performance.

It is much more complex to conduct learning analysis on categorical featured big data. The computational burden and inability to capture the genuine relationships between categorical features are the two main points which limit the traditional algorithms' application. With the increase of class-imbalanced data such as that derived from social networks and internet transactions, it is important to develop effective strategy for categorical big data.

In this thesis, a novel parallelized coupling framework to solve the problems mentioned above is proposed. The key contributions are as follows:

- Exploring the coupling interactions within categorical values to produce a more accurate similarity measurement by the intra-coupling relationship and inter-relationship.

- Parallelizing the coupling process using spark, which significantly increased the traditional algorithms' big data handling ability and efficiency.

- As Evaluated by both the classification algorithm and clustering algorithm, the parallelized coupling framework's strong ability has been proven, and also that the coupling relationship as essential for the calculation of categorical similarity has been indicated.

The rise of Big Data is driven by the rapid increase of complex data (Birney 2012). Documents posted on WWW servers, Web Sites, social networks, communication networks, and transportation networks etc. are all featured within complex data. Inspired by the Big Data challenges, many data mining methods have been developed to find interesting knowledge from Big Data with complex relationships. For example, finding the outliers in a social network (Borgatti, Mehra, Brass & Labianca 2009) can effectively identify spammers and provide safe networking environments for the society.

In the context of Big Data, there exists relationships between individuals. There also exist social relationships between individuals forming complex social networks, such as the relationship data extracted from Facebook, Twitter, LinkedIn and other social media (Birney 2012, Chen, Peng & Lee 2012), devices and sensors information (Ahmed & Karypis 2012), GPS map data, massive image files transferred in the internet, Web text and click-stream data (Alam, Ha & Lee 2012), e-mail (Liu & Wang 2012), etc. To deal with complex relationship networks, emerging research efforts have begun to address the issues of structure and evolution, crowds and interaction, and information and communication.

In order to adapt to the multi-source, massive, dynamic Big Data, researchers have expanded existing data mining methods in many ways, including the efficiency improvement of single-source knowledge discovery methods (Chang, Bai & Zhu 2009), the study of dynamic data mining methods and the analysis of convection data (Domingos & Hulten 2000). The purpose is to improve the efficiency of single-source mining methods. On the basis of gradual improvement of computer hardware functions, researchers continue to explore ways to improve the efficiency of knowledge discovery algorithms to make them better for massive data.

Wu et al. (Wu & Zhang 2003, Wu, Zhang & Zhang 2005, Su, Huang, Wu & Zhang 2006) proposed and established the theory of local pattern analysis, which has laid a foundation for global knowledge discovery in multi-source data mining. This theory provides a solution not only for the problem of the full search, but also for finding global models that traditional mining methods cannot find. Local pattern analysis of data processing can avoid putting different data sources together to carry out centralized computing.

The overlap similarity or cosine similarity(Yang, Cao & Zhang 2010) for categorical data is too vague to clearly describe how close two categorical instances are. Those similarity measures assume that the categorical features are independent to each other. However, more researchers argue that the similarity between categorical feature values is also dependent on the couplings of other features (Boriah et al. 2008). Wang et al. (Wang, Cao, Li, Wei & Ou 2012) presents a coupling nominal similarity to examine both the intra-coupling and inter-coupling of categorical features. Their approaches focuses on the clustering learning of the small scale data; whereas our proposed framework considers both the classification learning and the clustering learning on categorical big data, which cannot be handled by their methods.

6.2 Problem Statement

The usual way to deal with the similarity between two categorical instances is the cosine similarity on frequency and overlap similarity on feature category. However, they are too inexact to measure the similarity and they do not consider the coupling relationships among features. Wang et al. (Wang et al. 2012) introduce a coupling nominal similarity (COS) for categorical data, which addresses both the intra-coupled similarity within a feature and the inter-coupled similarity among different features. The proposed similarity measure has been shown to outperform the SMS and the ADD(Ahmad & Dey 2007*b*) in clustering learning. In this thesis, the definition of the coupled similarity is continuously refined, in this chapter, the COS in our classification algorithm is adapted and extended to big data scenery.

Definition 6.1 *Given a training data set D, a pair of values $v_j^x, v_j^y (v_j^x \neq v_j^y)$ of feature a_j. v_j^x and v_j^y are defined to be intra-related in feature a_j. The* ***Intra coupled similarity*** *(IaCS) between categorical feature values v_j^x and v_j^y of feature a_j is formalized as:*

$$S^{Intra}(v_j^x, v_j^y) = \frac{F(v_j^x) \cdot F(v_j^y)}{F(v_j^x) + F(v_j^y) + F(v_j^x) \cdot F(v_j^y)}, \tag{6.1}$$

where $F(v_j^x)$ and $F(v_j^y)$ are the occurrence frequency of values v_j^x and v_j^y in feature a_j, respectively.

The Intra coupled similarity reflects the interaction of two values in the same feature. The higher these frequencies are, the closer such two values are. Thus, Equation (6.1) is designed to capture the value similarity in terms of occurrence times by taking into account the frequencies of categories.

In contrast, the Inter coupled similarity below is defined to capture the interaction of two values in the same feature according to a value that comes from another feature.

Definition 6.2 *Given a training dataset D and two different features a_i and a_j $(i \neq j)$, two feature values $v_i^x, v_i^y (v_i^x \neq v_i^y)$ from feature a_i and a feature value v_j^z from feature a_j. v_i^x and v_i^y are inter-related if there exists at least one pair value (v_p^{xz}) or (v_p^{yz}) that co-occurs in features a_i and a_j of instance U_p. The* ***Inter coupled similarity*** *(IeCS) between feature values v_i^x and v_i^y of features a_i according to feature value v_j^z of a_j is formalized as:*

$$S_{i|j}^{Inter}(v_i^x, v_i^y | v_j^z) = 2 \cdot \frac{\min(CF(v_p^{xz}), CF(v_p^{yz}))}{F(v_i^x) + F(v_i^y)}, \tag{6.2}$$

where $CF(v_p^{xz})$ and $CF(v_p^{yz})$ are the co-occurrence frequency count function for value pair v_p^{xz} or v_p^{yz}, and $F(v_i^x)$ and $F(v_i^y)$ are the occurrence frequency of values v_i^x and v_i^y in feature a_i, respectively.

Accordingly, there is $S_{i|j}^{Inter} \in [0,1]$. The Inter-coupled similarity reflects the interaction or relationship of two categorical values from one feature but based on the connection to the other feature.

By taking into account both the intra coupled similarity and the inter coupled similarity, the *Integrated coupled similarity* (ICS) between instances u_{i_1} and u_{i_2} is formalized as:

$$\begin{aligned} &ICS(u_{i_1}, u_{i_2}) \\ &= \sum_{j=1}^{n} [\beta \cdot S^{Intra}(v_j^{i_1}, v_j^{i_2}) + (1-\beta) \cdot \sum_{k=1, k\neq j}^{n} S_{j|k}^{Inter}(v_j^{i_1}, v_k^{i_2})], \end{aligned} \tag{6.3}$$

where $\beta \in [0,1]$ is the parameter that decides the degree of influence that comes from the other attributes; $v_j^{i_1}$ and $v_j^{i_2}$ are the values of features j for instances u_{i_1} and u_{i_2}, respectively. S^{Intra} and $S_{j|k}^{Inter}$ are the intra-coupling feature value similarity and inter-coupling feature value similarity, respectively.

The metric is used in the experiments is called **ICS**, and the when calculating the similarity between every pair of attribute values for all attributes, the computational complexity linearly depends on the time cost of Information Conditional Probability (IeCS). Let $S_{j|k}$ represent the time cost of IeCS for $\delta_{j|k}(v_j^x, V_j^y)$ and suppose the maximal number of values for each attribute is $R(= max_{j=1}^{n}|V_j|)$. In total, the number of value pairs for all the attributes is at most

$$CSteps = nR(R-1)/2, \tag{6.4}$$

which is also the number of calculation steps. For inter-coupling relative similarity, *IeCS* for $|S_{j|k}|$ times is calculated. As there are n attributes, the total *IeCS* time cost is

$$FpS = 2(n-1)Q, \tag{6.5}$$

where Q is the cardinality of the inter-section sets of the Inter-information Function. This evidences that the computational complexity essentially depends linearly on the time cost of $IeCS$ with given data. Accordingly, the Complexity of **ICS** is:

$$Complexity = O(n^2R^2a). \tag{6.6}$$

where n, R, a is the total number of instances, the number of distinct values, and the number of attributes respectively. Due to this complexity, the previous researchers failed to apply it into large scale data sets.

In the following sections, how to accelerate the coupled similarity computation is illustrated, and integrating the coupled similarity into the revised kMeans and kNN is explained.

6.3 Parallelization

6.3.1 Spark Parallel Platform

The Spark platform allows developers to write a driver program that implements the high-level control flow of their application and launches various operations in parallel. Spark provides two main abstractions for parallel programming which are resilient distributed data sets and parallel operations to these data sets (invoked by passing a function to apply on a data set). In addition, Spark supports two restricted types of shared variables that can be used in functions running on the cluster, which will explain later.

A resilient distributed data set (RDD) is a read-only collection of objects partitioned across a set of machines that can be rebuilt if a partition is lost. The elements of an RDD do not need to exist in physical storage; instead, a handle to an RDD contains enough information to compute the RDD starting from data in reliable storage. This means that RDDs can always be reconstructed if nodes fail.

Several parallel operations can be performed on RDDs. For instance, *map* passes each element in parallel through a user provided function, and gets the result in parallel through the user provided function; *reduce* combines data set elements using an associative function to produce a result at the driver program; *collect* sends all elements of the data set to the driver program. For example, an easy way to update an array in parallel is to parallelize, map and collect the array.

Programmers invoke operations like map, filter and reduce by passing closures (functions) to Spark. As is typical in functional programming, these closures can refer to variables in the scope where they are created. Normally, when Spark runs a closure on a worker node, these variables are copied to the worker. However, Spark also lets programmers create two restricted types of shared variables to support two simple but common usage patterns. In the next section, these operations are uesed to revise the coupled similarity computation.

In the system perspective, the spark has several useful things to note in regard to this architecture.Each application gets its own executor processes, which stays up for the duration of the whole application and runs tasks in multiple threads. This has the benefit of isolating applications from each other, on both the scheduling side (each driver schedules its own tasks) and executor side (tasks from different applications run in different JVMs). However, it also means that data cannot be shared across different Spark applications (instances of SparkContext) without writing it to an external storage system.Spark is agnostic to the underlying cluster manager. As long as it can acquire executor processes, and these communicate with each other, it is relatively easy to run it even on a cluster manager that also supports other applications. Because the driver schedules tasks on the cluster, it should be run close to the worker nodes, preferably on the same local area network. If users want to send requests to the cluster remotely, its better to open an RPC to the driver and have it submit operations from nearby than to run a driver far away from the worker nodes. A simple graphical introduction can be found in Figure 6.1.

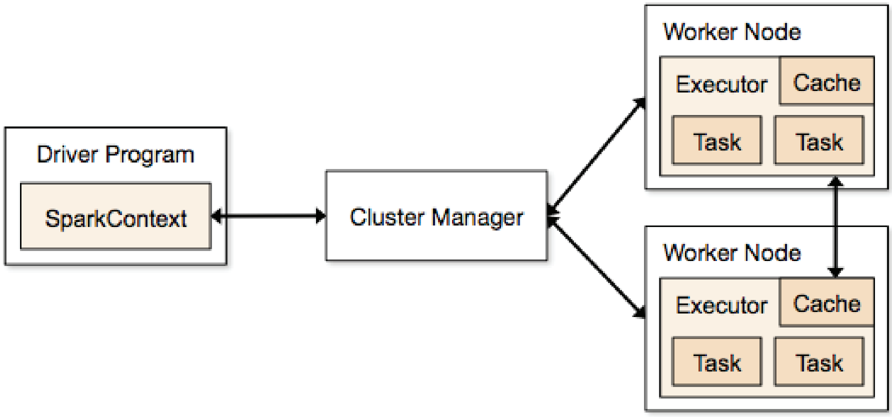


Figure 6.1: The Overview of the Spark Cluster.

6.3.2 Parallel Coupled Similarity

This section describes how to convert the coupled similarity to the Spark platform by a parallel way.

The first stage is the calculation of the **Intra coupled similarity**. By the definition 6.1, the main input into the formula is the frequency of each values. The source data set comes with a list of vectors v, each instance is a string vector and the whole data set as a list of vectors $\{v1, v2..v_n\}$. Firstly, the *map* function maps each vector the the attribute index $\{1, 2, ..a\}$ number into the string values $\{"A", "B", .."N"\}$, that make each value become a values (v_j, x) pair p_s and the vector is converted into $\{("A", 1), ("B", 2), ..("N", 3)\}$. Then $flatmap$ all the value pairs into a set of RDDs. Because each of the value index pair p_s is independent, the map-reduce scheme to count the occurrence of each p_s can be used in parallel and the value of the $F(v_j^x)$ is stored into a hash map to accelerate the later calculations . The work flow is described in Figure 6.2. Once the value of each distinct values is ascertained, the calculation of the **Intra coupled similarity** is straightforward. Again, the **Intra coupled similarity** S^{Intra} for each value pair is stored (p_s^i, p_s^j) in a hash map for the later computation of the *Integrated coupled similarity*.

The second stage is the calculation of the **Inter coupled similarity**. The same strategy is used to make the algorithm parallelize. Initially, same

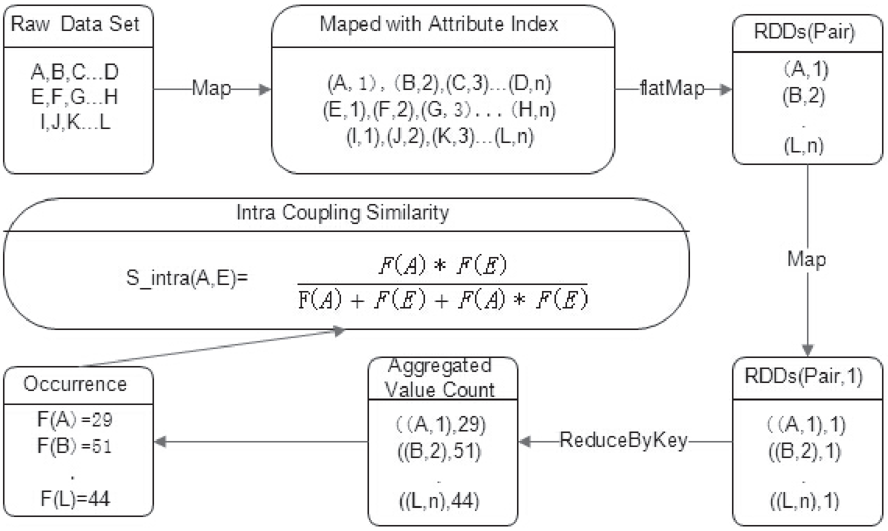


Figure 6.2: The Work Flow of Parallel Intra Coupled Similarity

data set with a list of vectors $\{v1, v2..v_n\}$ is identified, then the vector is converted into the index value pairs $\{(''A'', 1), (''B'', 2), ..(''N'', 3)\}$. Because the **Inter coupled similarity** requires an aggregate of the co-occurrence between different attributes, each two attributes are joined into a co-occurrence attribute pair $((''A'', 1), (''B'', 2))$ named p_d. Since the attribute pair p_d are also independent on each other, all the attribute pairs are flat mapped into a set of RDDs, then the map-reduce is used to group the list co-occurrence value pairs $\{p_s^1, p_s^2, ..p_s^n\}$ with the each distinct attribute pair in parallel. To make this easier to understand, the process has been illustrated in Figure 6.3. If there is the list of co-occurrence pairs for each value p_s, then when the **Inter coupled similarity** for two distinct values is calculated within one attribute p_s^i,p_s^j by joining their co-occurrence values pairs together and grouping by the attribute index and then finding the minimal co-occurrence count. By the definition of the **Inter coupled similarity**, the $S_{i|j}^{Inter}(v_i^x, v_i^y|v_j^z)$ for all the attributes can be calculated in one process, the details can be found in Figure 6.4. The calculation of the **Inter coupled similarity** is the most time consuming process of the previous work (Wang et al. 2012), however, the proposed process is fully parallel which dramatically improves the com-

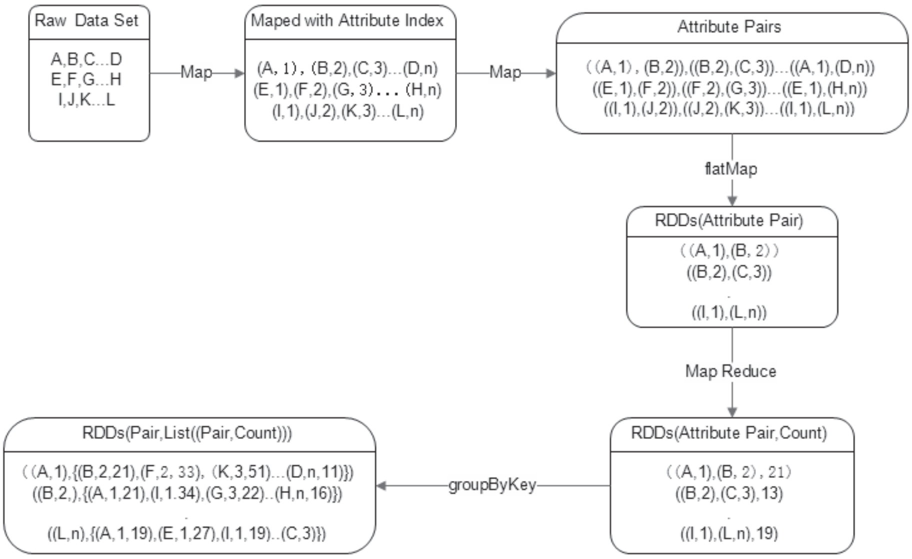

Raw Data Set
A,B,C...D
E,F,G...H
I,J,K...L
Map
Maped with Attribute Index
(A, 1), (B,2),(C,3)...(D,n)
(E,1),(F,2),(G, 3)... (H,n)
(I,1),(J,2),(K,3)...(L,n)
Map
Attribute Pairs
((A,1), (B,2)),((B,2),(C,3))...((A,1),(D,n))
((E,1),(F,2)),((F,2),(G,3))...((E,1),(H,n))
((I,1),(J,2)),((J,2),(K,3))...((I,1),(L,n))
flatMap
RDDs(Attribute Pair)
((A,1),(B, 2))
((B,2),(C,3))
.
((I,1),(L,n))
Map Reduce
RDDs(Attribute Pair,Count)
((A,1),(B, 2), 21)
((B,2),(C,3),13)
.
((I,1),(L,n),19)
groupByKey
RDDs(Pair,List((Pair,Count)))
((A,1),{(B,2,21),(F,2, 33), (K,3,51)...(D,n,11)})
((B,2,),{(A,1,21),(I,1.34),(G,3,22)..(H,n,16)})
.
((L,n),{(A,1,19),(E,1,27),(I,1,19)..(C,3)})

Figure 6.3: The Work Flow of Parallel Pre-Inter Coupled Similarity

putation efficiency compare to the loop based single thread algorithms. This is the main contribution of this chapter and the experiment will show that the proposed method reduce computational time by 99.9%. Again the **Inter coupled similarity** S^{Inter} was stored for each value pair (p_s^i, p_s^j) into a hash map for the later computation of the *Integrated coupled similarity*.

The last stage is the calculation of the *Integrated coupled similarity*. By the definition 6.3, by using the pre-stored **Intra coupled similarity** hash map and **Inter coupled similarity** hash map, the result of the *Integrated coupled similarity* between each instances can be quickly calculated. For the small data set this *Integrated coupled similarity* can be stored between each instance into a hash map for further classification and clustering tasks. Unfortunately, for the large data set due to the inefficient storage resource, this optimization method cannot be used.

Finally, this section introduces a novel parallel coupled similarity computation scenario, and the applications of the proposed parallel coupled similarity for both clustering and classification tasks will be discussed in the following sections.

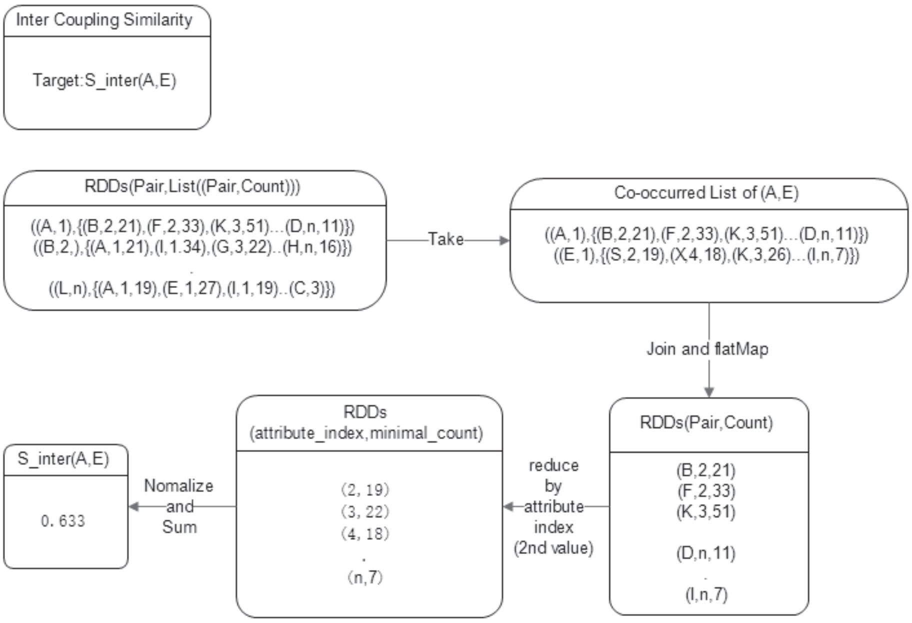

Inter Coupling Similarity
Target:S_inter(A,E)
RDDs(Pair,List((Pair,Count)))
((A,1),{(B,2,21),(F,2,33),(K,3,51)...(D,n,11)})
((B,2,),{(A,1,21),(I,1.34),(G,3,22)..(H,n,16)})
((L,n),{(A,1,19),(E,1,27),(I,1,19)..(C,3)})
Take
Co-occurred List of (A,E)
((A,1),{(B,2,21),(F,2,33),(K,3,51)...(D,n,11)})
((E,1),{(S,2,19),(X,4,18),(K,3,26)...(I,n,7)})
Join and flatMap
RDDs(Pair,Count)
(B,2,21)
(F,2,33)
(K,3,51)
(D,n,11)
(I,n,7)
reduce by attribute index (2nd value)
RDDs (attribute_index,minimal_count)
(2, 19)
(3, 22)
(4, 18)
(n,7)
Nomalize and Sum
S_inter(A,E)
0.633

Figure 6.4: The Work Flow of Parallel Inter Coupled Similarity

6.4 Integration

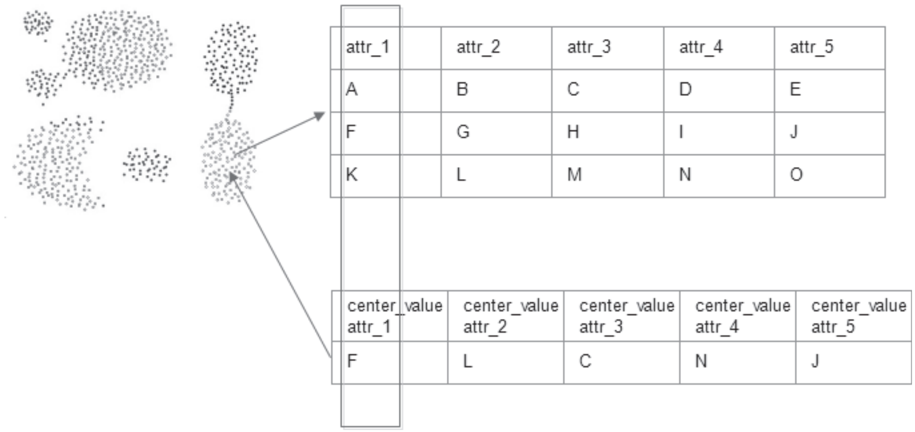
6.4.1 Parallel Coupled Similarity Based K-Means Algorithm

The most popular clustering algorithm is the K-Means algorithm. It is simple and scalable to the large data set, however, the K-Means algorithm is based on calculating the distance between two numerical instances. The vital problem for K-Means algorithm for the categorical data is that the mean of a set of categorical values cannot be computed. There are three popular strategies to handle the problem. The first one converts all the categorical value into numerical value by its frequency, and uses the traditional K-Means to do the clustering. The second method named K-Modes, calculates the center instance for each clusters instead of computing the mean of each attribute. Unfortunately, finding the center instance needs to compare the distance between all the instances within one cluster; the cost of computation exponentially increases with the number of instances within the cluster. While the classical K-Means only needs $O(n)$ to get similar periodical outcomes, it is scalable to the large scale data. The third solution is for every attribute within the cluster, to select the most frequent values as the center value of the attribute. The computational cost is smaller than the previous strategy, but there still some considerable disadvantages. For instance, for some very high frequent values, this will probably be selected as the center value for all the clusters, however it may not the most representable feature for that cluster.

This thesis proposed a novel parallel coupled similarity base K-Means Algorithm. The first step is to compute the coupling similarities between each value within one attribute. The K step is the same as the classical K-Means, randomly generating K center points, and assigning the instances to the nearest center points by the coupled similarity metric. In the $Means$ step, there is some modification of the proposed method compared to the classical K-Means algorithm. Combining the advantage of both K-Modes and frequency based method, for each attribute, the center value of every values are selected within this attribute. With the help of the coupled similarity, the distance between each values can be measured. The detail is described in Figure 6.5. There is proof that the number of distinct values is smaller than the number of instances, however, there is no space to discuss the detail. Hence, the proposed method could be faster than the K-Mode clustering algorithm. The experiment results shows that the proposed method has the ability to scale to very large data set, and the clustering performance outperforms the traditional methods.

6.4.2 Parallel Coupled Similarity Based KNN Algorithm

Classification learning from the categorical big data can be formally described as follows: $U = \{u_1, \cdots, u_m\}$ is a set of m instances, in which m is a dramatically large numbers; $A = \{a_1, \cdots, a_n\}$ is a set of n categorical attributes; $C = \{c_1, \cdots, c_L\}$ is a set of L classes. The goal is to classify an unlabelled



attr_1	attr_2	attr_3	attr_4	attr_5
A	B	C	D	E
F	G	H	I	J
K	L	M	N	O

center_value attr_1	center_value attr_2	center_value attr_3	center_value attr_4	center_value attr_5
F	L	C	N	J

Figure 6.5: Coupled Similarity Based Mean Step for Clustering

testing instance u_t based on the instances in the training set $\{u_i\}$ with known classes.

In the Spark platform, the implementation of the parallel coupled similarity based KNN algorithm is straightforward. Because the coupled similarity between two instances is independent from all other instances, calculate them one by one in parrallels.

6.5 Experiments and Evaluation

6.5.1 Experiment Setting

The experiment is on a heterogeneous cluster with following two configurations; 4 nodes from the Phoenix cluster with 3.4GHz Intel Xeon E5-2687W v2 (8 Cores) 25MB L3 Cache (Max turbo 4.0GHz, Min. 3.6GHz), and 32GB 1866MHz ECC DDR3-RAM (Quad Channel); 4 nodes from the Orion cluster with 2.9GHz Intel Xeon E5-2690 (8 Cores) 20MB L3 QPI (Max Turbo Freq. 3.8GHz, Min 3.3GHz), and 32GB 1600MHz ECC DDR3-RAM (Quad Channel). In total, there are 8 nodes with 64 cpu cores and 242GB RAM and they are connected by a 1Gbit ethernet. The Spark version 1.2.0 as the cluster computation framework software is used. One node from the Phoenix

Table 6.1: Experimental Data Sets

Data Set	**#Ins**	**#Att**	**Attribute Types**
Shuttle	58,000	9	Categorical
Poker Hand	1,025,010	11	Categorical, Integer
US Census	2,458,285	68	Categorical
KDD Cup 1999	4,000,000	42	Categorical, Integer

cluster is set as both the master and slave node, with 12G RAM for the cluster management use and 12G RAM for the computation use. Meanwhile, other 7 nodes are all set as the slave node with 12G RAM for computation use. There is a separate storage server with extensive storage space and high speed connection to all the nodes, hence there is no Hadoop HDFS system in use.

The data sets used are all from the UCI machine learning repository, they are the $Shuttle$, $PokerHand$, $USCensus$, $KDDCup1999$. The description of the data set is in the Table 6.1, the $Shuttle$ data set is the largest data set Can etc have explored. The $Shuttle$ data has 58000 instances with 9 attributes. To prove the scalability of the proposed method, three much bigger data sets were chosen. The $PokerHand$ data set has 1025010 instances with 11 attributes, the $USCensus$ data set has 2458285 instances with 68 attributes, and the $KDDCup1999$ data set has 4000000 instances with 42 attributes. These three data sets were chosen because they are the only three data sets has more than 1 million instances and have categorical attributes from the UCI machine learning repository.

As discussed in the previous chapter, the theoretical complexity of the coupled similarity is $O(n^2R^2a)$. This section, evaluates the scalability of the Spark coupled similarity computation in total performance, number of instances n, number of attributes I, and the core of the cluster, respectively. Though the number of distinct values R is a vital parameter of the total computation cost, it is hard to select from or restrict different data sets. Therefore, in this thesis, this parameter has not been evaluated, but to reduce

Table 6.2: Efficiency on Full Data Sets

Data Set	**#IRSI**	**#PICS**
Shuttle	6h	7s
Poker Hand	>48h(Fail)	11s
US Census	>48h(Fail)	2.1m
KDD Cup 1999	>48h(Fail)	4m

the computation cost, those numerical features have been discreted into ten equal frequency bins. Firstly, there is an assessment to the availability to all the data sets. In the following table 6.2, the total time consumption of whole data in both attributes and instances is demonstrated by using the maximal computational power of the cluster. Compared to the previous work, the proposed method shows that it can handle the large data set to get the proper outcome. If compare to the same data set(*shuttle*), the proposed method is more than one thousand times faster than the previous work.

6.5.2 The Scalability of the Spark Coupled Similarity

The second experiment is on the scalability of the number of instances and the result is shown in Figure 6.6. In theory, the complexity of the number of instances is n^2. Due to the system initialization cost and the network cost, the experiment result does not strictly follow the theoretical complexity. The time-consumption not increase rapidly when the number of instances increase. The computation time cost for each instance becomes less and less when the number of instance grows. This means for larger data sets the proposed method can be more efficient in terms of the computation time cost of the single instance. It is clear that the $KDDCpu1999$ data set and $USCensus$ date set share the same pattern when the number of instances increases while the $PorkerHand$ data set always costs much less computational time due to the number of attributes and especially as the distinct values are small. This indicates that the computational time cost is related

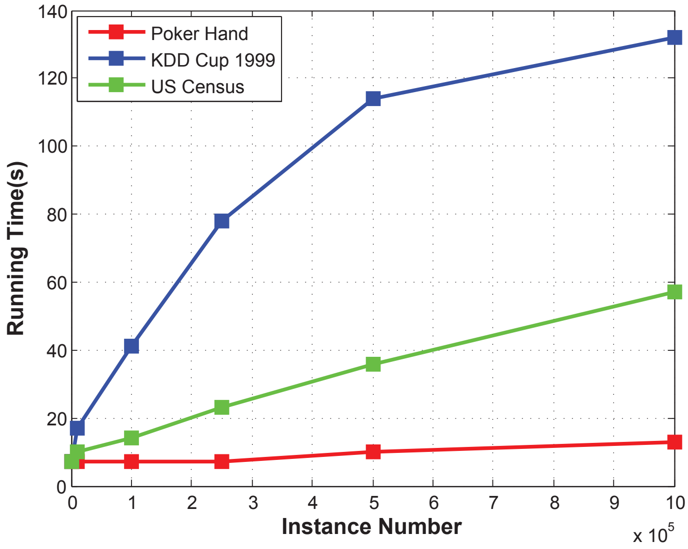


Figure 6.6: The Running time with different data sets.

to the character of the data set.

The third experiment is on the scalability of the number of attributes. Based on the theoretical analysis, the computational time cost should rise linearly with the increase in the number of attributes. The number of attributes dose not actively effect the computational complexity, however, the number of distinct value pairs is highly related to the number of attributes. As discussed in the previous section, the distinct value pairs are the genuine character of data set, and it is hard to select or restrict. Hence, this experiment is just partially shows the scalability of the proposed method on the expansion of the attributes. As it shows in Figure 6.7, the assumed computational time cost following the linear increase, is not correct, as that the number of distinct value pairs also increases when the number of attributes increase. However, the experiment still shows an acceptable scalability of the proposed method based on the number of attributes.

The last experiment is on the scalability of the number of cores that are used in the cluster. This experiment aims to illustrate the potential of the proposed method that can be performed when the data sets become even

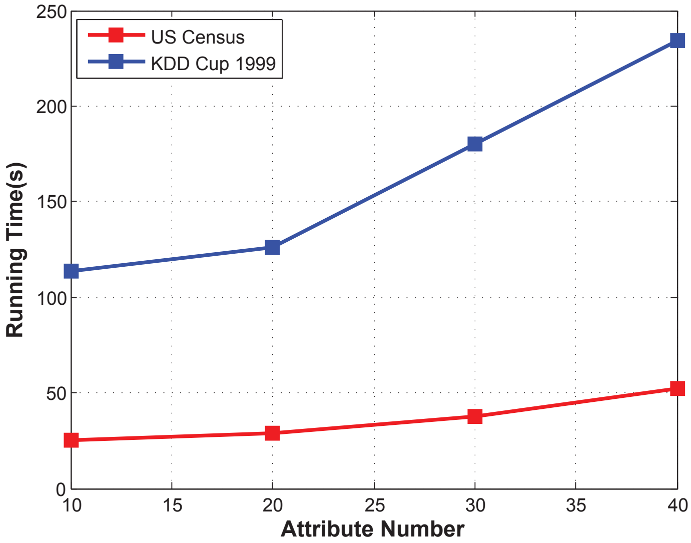


Figure 6.7: The Running time with different feature numbers.

larger. The Figure 6.8 demonstrates that when a sufficient computation resource has been added, the computational time cost has been dropped down rapidly. However, it does not decrease linearly due to the expense of the network and resource management. Nevertheless, when it has more cores, the computational time cost is much shorter than a single core. This shows the scalability of the proposed method of the increase of cores.

In summary, the scalability of the proposed parallelized coupled similarity computation scenario in many perspectives has been evaluated. The experiment shows that it can scale to large data sets though the computation cost of the coupling is significant. The previous researchers failed to explore the clustering and classification performance on the large scale data because they could not get the coupled similarity result in an acceptable time. This work has made it possible, and the next two sections will show how the coupled similarity can significantly increase the clustering and classification performance.

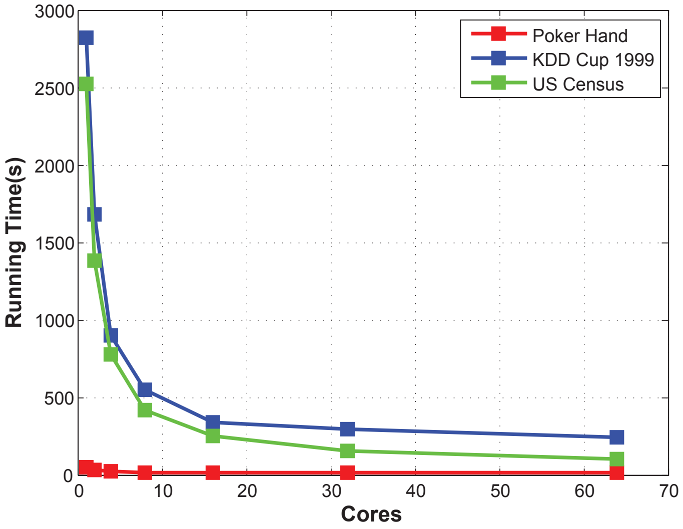


Figure 6.8: The Running time with different cores on 3 data sets.

6.5.3 The Performance of SK-Means

This section evaluates the clustering performance and the scalability of the parallel coupled similarity based K-Means via Spark cluster. (Wang et al. 2012) had already successfully evaluated the clustering performance by using the coupling similarities on several tiny data sets. In this thesis, the experiment has been done on three much bigger data sets. For the comparison, two classic similarity metrics for the categorical data were used. They are overlap similarity and frequency based similarity used as the baseline. Overlap similarity accumulates the exactly matched value count, and the frequency based similarity converts the categorical data into the numerical value by using the frequency for each value, and computing the Euclidean distance by the numerical values. In this thesis, the *Purity* is used as the metric to indicate the clustering performance.

Purity is a simple and transparent evaluation measure for clustering algorithms. To compute purity , each cluster is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned documents and

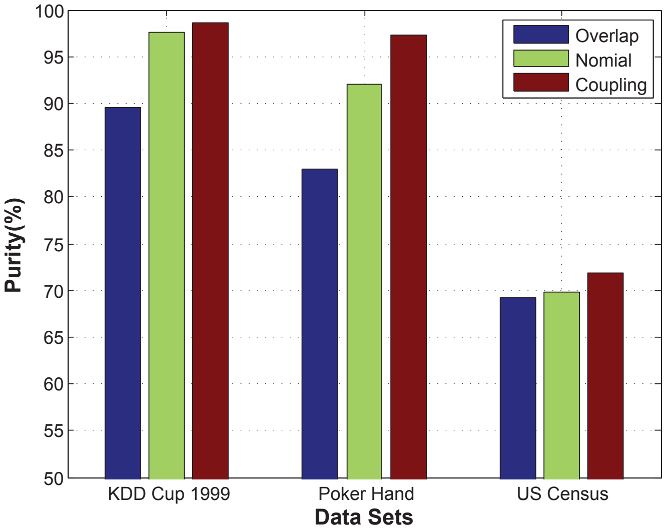


Figure 6.9: The Purity Results on 10^4 instances.

dividing it by N. Formally:

$$Purity = \frac{1}{N} \sum_k \max_j |\omega_k \cap c_j| \tag{6.7}$$

where ω_k is the kth cluster and c_j is the jth class.

The performance is evaluated on three different scales which are 10^4 , 10^5 and 10^6 instances, randomly taking the instances from each complete data set for the clustering task. Figure 6.9 shows the clustering purity for three different data sets. It is clear that the coupled similarity metric outperforms the overlap and frequency metrics. Figure 6.10 and Figure 6.11 drew the similar conclusion on different scales. This experiment indicates that the coupled similarity has its advantages over the classic method in various data sets. Furthermore, this experiment also proved the proposed parallel clustering method via Spark cluster is scalable and stable on different data scales.

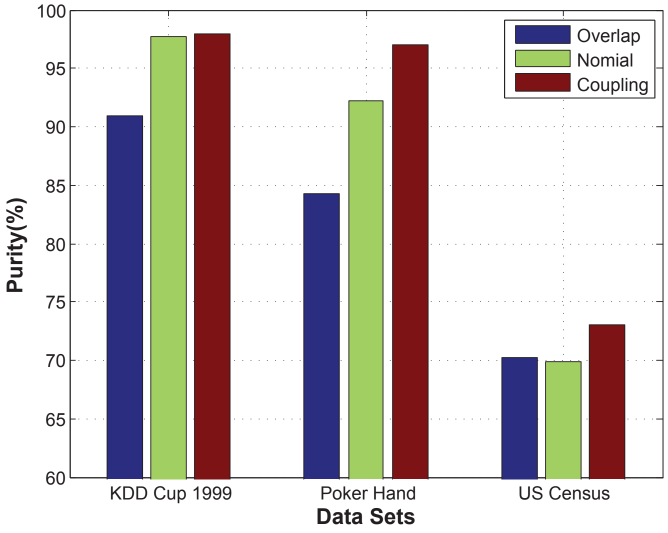


Figure 6.10: The Purity Results on 10^5 instances.

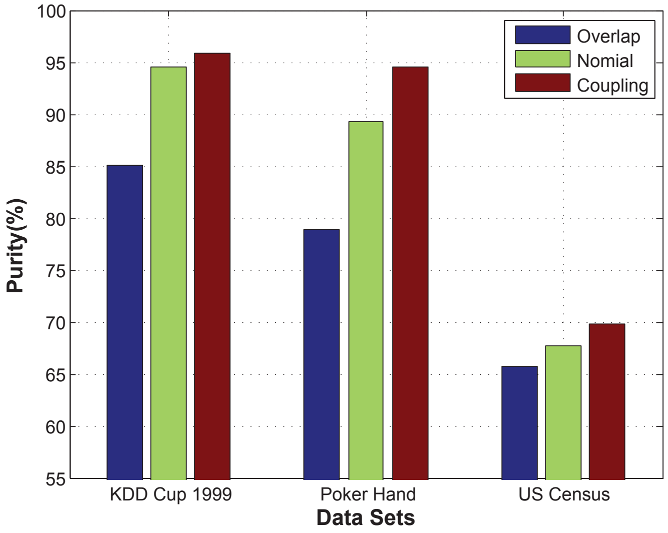


Figure 6.11: The Purity Results on 10^6 instances.

6.5.4 The Performance of SKNN

This section evaluates the classification performance and scalability of the proposed parallel coupled similarity based KNN via spark cluster. (Liu, Cao & Yu 2014) presented the classification performance on some small data set. Similar to the previous section, in this thesis, the experiment on three much larger data sets has been performed. The comparable baseline similarity metrics are identical with the previous section that are the overlap similarity and the frequency based similarity. The scalability of the proposed parallel coupled similarity based KNN has also been explored. Similar to the clustering task, there is random selection of 10^4 , 10^5 and 10^6 instances for testing and the rest of the data is used as the training(KNN search range)set. Figure 6.12 demonstrate the proposed method outperforming the two classical similarity metric on the classification accuracy. Moreover, Figure 6.13 and 6.14 combined show that the proposed method has a table classification performance on different scale of data set, and also has the potential to cope with even larger data sets. The metric indicates that the performance of the classification is *accuracy*.

Accuracy is a widely used evaluation criteria to describe a classifier's performance, and it is used to quantify how good and reliable a classifier is. Accuracy is defined as:

$$\text{Accuracy} = \frac{(TN + TP)}{(TN + TP + FN + FP)} \tag{6.8}$$

where TP, FP, FN and TN is True positive, False positive, False negative and True negative, respectively.

6.6 Application

As in the previous chapters, after the experiment has been performed on the public data set, the proposed method is run on the student related data source again. In this section, the proposed parallelized k-nn with coupling similarity was readministered to on the student at risk data set(Chapter 3) and student

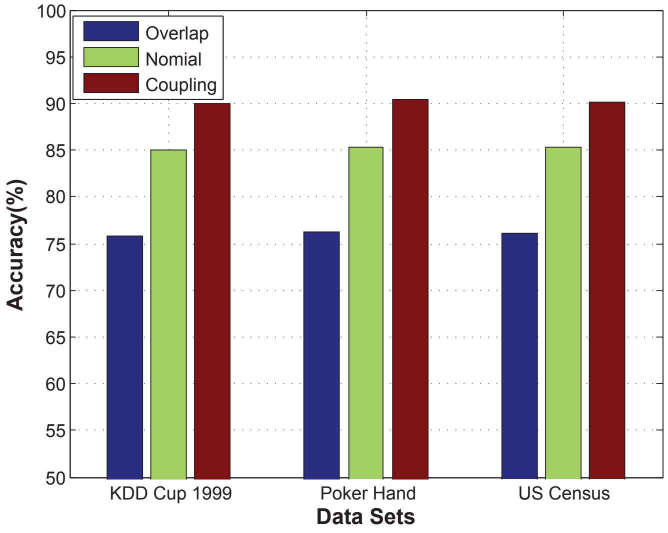


Figure 6.12: The Accuracy Results on 10^5 instances.

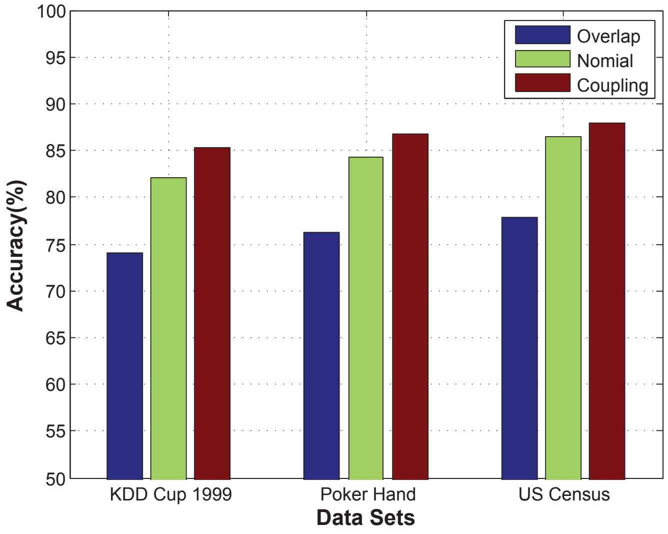


Figure 6.13: The Accuracy Results on 10^5 instances.

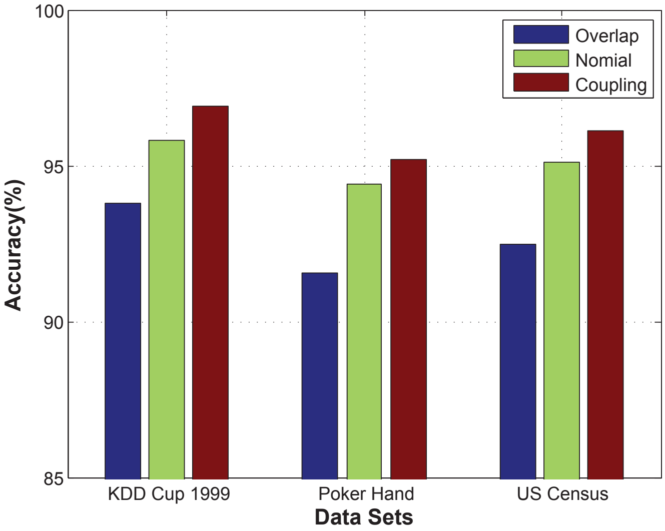


Figure 6.14: The Accuracy Results on 10^6 instances.

sentiment analysis data set(Chapter 5), evidence of 10 years historical student related data with grades as labels, and three million labeled social media data with sentiment tags, respectively. Since the scale of data size of the chapter 4 experiment is not fit with the big data platform, the experiment on that data set was not rerun. Figure 6.15 indicates that when the number of cores in the cluster increases, the time cost of the testing time reduces constantly and significantly, therefore, it inherits the scalability of the previous experiment. The Chapter 5 does not mentioned the time cost of the proposed method, because it takes nearly two weeks to run. Fortunately, benefitting from the proposed parallel coupling analysis algorithm, the sentiment analysis of the student social media can be done within an acceptable time. Figure 6.16 illustrates how the proposed framework reduces the time cost of the student sentiment analysis task.

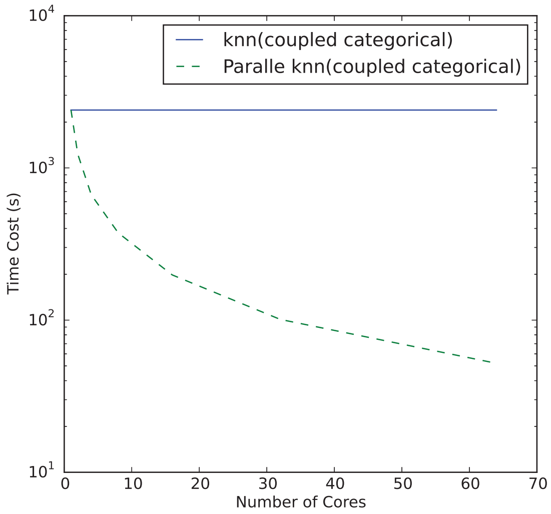


Figure 6.15: Time Cost via Different Number of CPU Cores(Student Performance)

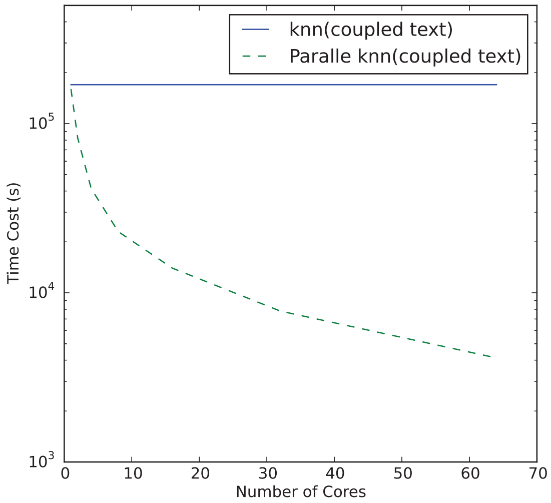


Figure 6.16: Time Cost via Different Number of CPU Cores(Student Sentiment)

6.7 Summary

Traditional instance-based learning methodologies mainly focus on dealing with small data sets if they overlook the coupling relationships between attributes. Because of the high computational cost, learning the coupling relations on the large scale categorical data is very challenging. The proposed a novel coupled similarity for large scale data, it can not only be easily integrated with the classic method (such as kNN and kMeans), but also can extend to any distance based machine learning algorithms. It effectively extracts the inter and intra coupling relationships between categorical values. The experiment results show that this framework has a more stable and higher average performance than the baseline algorithms; moreover, it can be adapted to large scale categorical data.

Chapter 7

Conclusions and Future Work

7.1 Conclusions

This thesis discussed a coupling analysis framework for educational data. T-wo algorithms had been introduced that employ coupling analysis for student performance prediction tasks, a student sentiment analysis tool for social media data, and a scalable spark cluster based coupling analysis system. More precisely, a coupling distance based K-nearest weighted centroid classification method has been proposed in Chapter 3, and applied a coupling object similarity metric which involves both an attributes value frequency distribution (intra-coupling) and a feature dependency aggregation (inter-coupling) in measuring attribute value similarity for the classification of nominal data. Experiments have shown that applied inter-coupling relative similarity measures significantly outperform the other methods and manifest distinct advantages in the student performance prediction task. In chapter 4 proposed a novel pairwise SVM classification method with a coupling similarity. It involves the coupling relation between categorical data and adaptively intergrades with the pairwise SVM. Substantial experimental results have demonstrated the advantage of the proposed method especially for the ”killer subject” analysis. However, the efficiency of the proposed method is not that good. Even though classic SVM is already time consuming, our method is worse than that. Unfortunately, it cannot be performed to a larger data set. In Chapter 5 introduced the concept of coupled similarity from term to term based on its co-occurrence as neighborhood in a sliding window, a new paradigm for text representation and processing. The coupling relation maintains information about the relative placement of words with respect to each other, and this provides a richer representation for document classification purposes. This similarity can be used in order to exploit the recent advancements in text mining algorithms. Furthermore, the coupled similarity criterion can be used with minimal changes to existing data mining algorithms if it is suited. Thus, the new coupled similarity framework does not require additional development of new data mining algorithms. This is a huge advantage since existing text processing and mining infrastructure can be used directly with the coupled similarity based model. Meanwhile, the proposed method has been successfully applied in the student social media sentiment analysis task. Though coupling analysis shows its advantage in the algorithm's performance, due to its high computational cost, it cannot be widely applied to other environments. Finally, chapter 6 proposed a novel coupled similarity for large scale data which solves the scalability problem of all aforementioned methods, Indeed it not only can be easily integrated with the classic method (such as kNN and kMeans), but can be applied to any distance based machine learning algorithms. It effectively extracts the inter and intra coupling relationships between categorical values. The experiment results show that the proposed framework can outperform the traditional method's performance and finish the student performance prediction and student sentiment analysis task within an acceptable time.

7.2 Future Work

7.2.1 Algorithms

Coupling analysis is very foundation part of the machine learning algorithm, it can be applied to most of instance or similarity based method. This thesis applied the coupling analysis concept in two basic algorithms, it can be easily applied to other kernel machines and RBF networks. For the coupled similarity computation itself also has the great potential to improve. In terms of scalability, this thesis extended the k-NN and k-Means with coupling similarity into a scalable version. As both kNN and kMeans are very basic methods, it can be applied the proposed parallel coupling distance similarity to more sophisticated method.

In terms of the coupled term to term analysis for the text mining, the future works are the follows. Firstly, this thesis only considered term frequencyCinverse document frequency as the feature to build the vector presentation, it can be improved if involve more information such as semantic annotation for each term or position of the term in the document. Secondly, calculating the intra-coupled similarity, this thesis equally treated every feature in the vector presentation; however, they are not homogeneous, it should be found a better way to measure the relations of each feature, and it also useful to consider the same thing when deal with the relation from term to term and follow by document to document. Finally, in the longer plan, the optimization of the parameter should be taken into account, and some approximation also should be added because of the computational cost is high.

For the scalability, it still has potential to optimize. For instance, design a decent data structure of the RDD storage for the map-reduce process could make it more efficient than using the default data format which provide by Spark; apply a proper cache scheme in the Spark cluster also could make a significant improvement of the efficiency. In conclusion, the future work aims to integrate the algorithm to more applications and increase the algorithm efficiency and decrease the time complexity.

7.2.2 Applications

After discussing the algorithm part, it is also has enormous opportunities to apply them into educational data. In the system level, the proposed scalable coupling analysis framework can be the educational data analysis tools that designed to be easier for educators or non-expert users in data analytics. Data analysis tools are normally designed more for power and flexibility than for simplicity. Most of the current data analysis tools are too complex for educators to use and their features go well beyond the scope of what an educator may want to do. For example, on the one hand, users have to select the specific data mining method algorithm they want to apply use from the wide range of methods algorithms available on data mining. On the other hand, most of the data mining algorithms need to be configured before they are executed. Users have to provide appropriate values for the parameters in advance in order to obtain good results models and so, the user must possess a certain amount of expertise in order to find the right settings. One possible solution is the development of wizard tools that use a default algorithm for each task and parameter-free data mining algorithms to simplify the configuration and execution for nonexpert users. education data analysis tools must also have a more intuitive interface that is easy to use and with good visualization facilities to make their results meaningful to educators and e-learning designers (Romero, Ventura, García & de Castro 2009). It is also very important to develop specific preprocessing tools in order to automate and facilitate all the preprocessing functions or tasks that educational data analysis users currently must do manually.

Appendix A

7.1 List of Publications

Papers Published

- **Mu Li**, Jinjiu Li, Yuming Ou, Ya Zhang, Dan Luo, Maninder Bahtia, Longbing Cao: **Coupled K-Nearest Centroid Classification for Non-iid Data**. T. Computational Collective Intelligence 15, pp. 89-100 (2014)

- **Mu Li**, Jinjiu Li, Yuming Ou, Ya Zhang, Dan Luo, Maninder Bahtia, Longbing Cao: **Learning Heterogeneous Coupling Relationships Between Non-IID Terms**. ADMI 2013, pp. 79-91

- **Mu Li**, Jinjiu Li, Yuming Ou, Longbing Cao: **An Coupled Similarity Kernel for Pairwise Support Vector Machine**. ADMI 2014

- Jinjiu Li, Can Wang, Wei Wei, **Mu Li**, **Chunming Liu: Efficient Mining of Contrast Patterns on Large Scale Imbalanced Real-Life Data**. PAKDD (1) 2013, pp. 62-73

- Wei Wei, Jinyan Li, Longbing Cao, Jingguang Sun, Chunming Liu, **Mu Li**: **Optimal Allocation of High Dimensional Assets through Canonical Vines**. PAKDD (1) 2013, pp. 366-377

Papers to be Submitted/Under Review

- **Mu Li**, Chuming Liu, Longbing Cao: **Learning Large-Scale Coupling Relationships**. Submitted to DSAA 2015.

Bibliography

Abbas, S. & Sawamura, H. (2008), A first step towards argument mining and its use in arguing agents and its, *in* ‘Knowledge-based intelligent information and engineering systems’, Springer, pp. 149–157.

Abel, F., Bittencourt, I. I., Henze, N., Krause, D. & Vassileva, J. (2008), A rule-based recommender system for online discussion forums, *in* ‘Adaptive Hypermedia and Adaptive Web-Based Systems’, Springer, pp. 12–21.

Ahmad, A. & Dey, L. (2007*a*), ‘A k-mean clustering algorithm for mixed numeric and categorical data’, *Data and Knowledge Engineering* **63**, 503–527.

Ahmad, A. & Dey, L. (2007*b*), ‘A k-mean clustering algorithm for mixed numeric and categorical data’, *Data and Knowledge Engineering* **63**(2), 503–527.

Ahmed, R. & Karypis, G. (2012), ‘Algorithms for mining the evolution of conserved relational states in dynamic networks’, *Knowledge and Information Systems* **33**(3), 603–630.

Alam, M. H., Ha, J. & Lee, S. (2012), ‘Novel approaches to crawling important pages early’, *Knowledge and Information Systems* **33**(3), 707–734.

Alfonseca, E., Rodríguez, P. & Pérez, D. (2007), ‘An approach for automatic generation of adaptive hypermedia in education with multilingual knowledge discovery techniques', *Computers & Education* **49**(2), 495–513.

Amershi, S. & Conati, C. (2009), 'Combining unsupervised and supervised classification to build user models for exploratory', *JEDM-Journal of Educational Data Mining* **1**(1), 18–71.

Anaya, A. R. & Boticario, J. G. (2009), 'A data mining approach to reveal representative collaboration indicators in open collaboration frameworks.', *International Working Group on Educational Data Mining* .

Andrejko, A., Barla, M., Bieliková, M. & Tvarozek, M. (2007), 'User characteristics acquisition from logs with semantics.', *ISIM* **7**, 103–110.

Anozie, N. & Junker, B. W. (2006), Predicting end-of-year accountability assessment scores from monthly student records in an online tutoring system, Educational Data Mining: Papers from the AAAI Workshop. Menlo Park, CA: AAAI Press.

Antunes, C. (2008), Acquiring background knowledge for intelligent tutoring systems., *in* 'EDM', pp. 18–27.

Arnold, A., Scheines, R., Beck, J. E. & Jerome, B. (2005), Time and attention: Students, sessions, and tasks, *in* 'Proceedings of the AAAI 2005 Workshop Educational Data Mining', pp. 62–66.

Arroyo, I., Murray, T., Woolf, B. P. & Beal, C. (2004), Inferring unobservable learning variables from students help seeking behavior, *in* 'Intelligent tutoring systems', Springer, pp. 782–784.

Avouris, N., Komis, V., Fiotakis, G., Margaritis, M. & Voyiatzaki, E. (2005), Logging of fingertip actions is not enough for analysis of learning activities, *in* '12th International Conference on Artificial Intelligence in Education, AIED 05 Workshop 1: Usage analysis in learning systems', pp. 1–8.

Ayers, E. & Junker, B. (2006), Do skills combine additively to predict task difficulty in eighth grade mathematics, Educational Data Mining: Papers from the AAAI Workshop. Menlo Park, CA: AAAI Press.

Ayers, E., Nugent, R. & Dean, N. (2009), 'A comparison of student skill knowledge estimates.', *International Working Group on Educational Data Mining* .

Ba-Omar, H., Petrounias, I. & Anwar, F. (2007), A framework for using web usage mining to personalise e-learning, *in* 'Advanced Learning Technologies, 2007. ICALT 2007. Seventh IEEE International Conference on', IEEE, pp. 937–938.

Baker, R. S. (2007), Modeling and understanding students' off-task behavior in intelligent tutoring systems, *in* 'Proceedings of the SIGCHI conference on Human factors in computing systems', ACM, pp. 1059–1068.

Baker, R. S., Corbett, A. T. & Aleven, V. (2008), 'Improving contextual models of guessing and slipping with a truncated training set', *Human-Computer Interaction Institute* p. 17.

Baker, R. S., Corbett, A. T. & Koedinger, K. R. (2004), Detecting student misuse of intelligent tutoring systems, *in* 'Intelligent tutoring systems', Springer, pp. 531–540.

Baker, R. et al. (2010), 'Data mining for education', *International encyclopedia of education* **7**, 112–118.

Bari, M. & Lavoie, B. (2007), Predicting interactive properties by mining educational multimedia presentations, *in* 'Information and Communications Technology, 2007. ICICT 2007. ITI 5th International Conference on', IEEE, pp. 231–234.

Barker-Plummer, D., Cox, R. & Dale, R. (2009), 'Dimensions of difficulty in translating natural language into first order logic.', *International Working Group on Educational Data Mining* .

Barnes, T. (2005), The q-matrix method: Mining student response data for knowledge, *in* 'American Association for Artificial Intelligence 2005 Educational Data Mining Workshop'.

Barnes, T. & Stamper, J. (2008), Toward automatic hint generation for logic proof tutoring using historical student data, *in* 'Intelligent Tutoring Systems', Springer, pp. 373–382.

Baruque, C. B., Amaral, M. A., Barcellos, A., da Silva Freitas, J. C. & Longo, C. J. (2007), Analysing users' access logs in moodle to improve e learning, *in* 'Proceedings of the 2007 Euro American conference on Telematics and information systems', ACM, p. 72.

Beal, C. R. & Cohen, P. R. (2008), Temporal data mining for educational applications, *in* 'PRICAI 2008: Trends in Artificial Intelligence', Springer, pp. 66–77.

Beck, J., Baker, R., Corbett, A., Kay, J., Litman, D., Mitrovic, T. & Ritter, S. (2004), Workshop on analyzing student-tutor interaction logs to improve educational outcomes, *in* 'Intelligent Tutoring Systems', Springer, pp. 909–909.

Beck, J. E. & Woolf, B. P. (2000), High-level student modeling with machine learning, *in* 'Intelligent tutoring systems', Springer, pp. 584–593.

Beikzadeh, M. R., Phon-Amnuaisuk, S. & Delavari, N. (2008), 'Data mining application in higher learning institutions', *Informatics in Education-An International Journal* (Vol 7_1), 31–54.

Bellaachia, A., Vommina, E. & Berrada, B. (2006), Minel: A framework for mining e-learning logs, *in* 'Proceedings of the 5th IASTED international conference on Web-based education', ACTA Press, pp. 259–263.

Bellegarda, J. R. (2004), 'Statistical language model adaptation: review and perspectives', *Speech communication* **42**(1), 93–108.

Bellman, R., Bellman, R. E., Bellman, R. E. & Bellman, R. E. (1961), *Adaptive control processes: a guided tour*, Vol. 4, Princeton university press Princeton.

Ben-Naim, D., Bain, M. & Marcus, N. (2009), 'A user-driven and data-driven approach for supporting teachers in reflection and adaptation of adaptive tutorials.', *International Working Group on Educational Data Mining* .

Ben-Naim, D., Marcus, N. & Bain, M. (2008), Visualization and analysis of student interaction in an adaptive exploratory learning environment, *in* 'International Workshop on Intelligent Support for Exploratory Environment, EC-TEL', Vol. 8.

Bengio, Y., Schwenk, H., Senécal, J.-S., Morin, F. & Gauvain, J.-L. (2006), Neural probabilistic language models, *in* 'Innovations in Machine Learning', Springer, pp. 137–186.

Birney, E. (2012), 'The making of encode: lessons for big-data projects', *Nature* **489**(7414), 49–51.

Bordes, A., Glorot, X., Weston, J. & Bengio, Y. (2012), Joint learning of words and meaning representations for open-text semantic parsing, *in* 'International Conference on Artificial Intelligence and Statistics', p-p. 127–135.

Borgatti, S. P., Mehra, A., Brass, D. J. & Labianca, G. (2009), 'Network analysis in the social sciences', *science* **323**(5916), 892–895.

Boriah, S., Chandola, V. & Kumar, V. (2008), Similarity measures for categorical data: A comparative evaluation, *in* 'Proceedings of the 8th SIAM International Conference on Data Mining', pp. 243–254.

Bravo, J. & Ortigosa, A. (2009), 'Detecting symptoms of low performance using production rules.', *International Working Group on Educational Data Mining* .

Brunner, C., Fischer, A., Luig, K. & Thies, T. (2012), ‘Pairwise support vector machines and their application to large scale problems’, *Journal of Machine Learning Research* **13**, 2279–2292.

Burr, L. & Spennemann, D. H. (2004), ‘Patterns of user behaviour in university online forums’, *International Journal of Instructional Technology and Distance Learning* **1**(10), 11–28.

Cakir, M., Xhafa, F., Zhou, N. & Stahl, G. (2005), Thread-based analysis of patterns of collaborative interaction in chat., *in* ‘AIED’, Vol. 125, pp. 120–127.

Calvo-Flores, M. D., Galindo, E. G., Jiménez, M. P. & Pineiro, O. P. (2006), ‘Predicting students marks from moodle logs using neural network models’, *Current Developments in Technology-Assisted Education* **1**, 586–590.

Campbell, A. R. & Dickson, C. J. (1996), ‘Predicting student success: A 10-year review using integrative review and meta-analysis’, *Journal of Professional Nursing* **12**(1), 47–59.

Cao, L. (2013), ‘Non-iidness learning: an overview’, *The Computer Journal* pp. 1–18.

Cao, L. (2014), ‘Coupling learning of complex interactions’, *Information Processing & Management* .

Cao, L. & Philip, S. Y. (2012), *Behavior Computing: Modeling, Analysis, Mining and Decision*, Springer.

Castro, F., Vellido, A., Nebot, A. & Minguillon, J. (2005), Detecting atypical student behaviour on an e-learning system, *in* ‘VI Congreso Nacional de Informática Educativa, Simposio Nacional de Tecnologías de la Información y las Comunicaciones en la Educación, SINTICE’, pp. 14–16.

Castro, F., Vellido, A., Nebot, À. & Mugica, F. (2007), Applying data mining techniques to e-learning problems, *in* 'Evolution of teaching and learning paradigms in intelligent environment', Springer, pp. 183–221.

Cetintas, S., Si, L., Xin, Y. P. & Hord, C. (2009), 'Predicting correctness of problem solving from low-level log data in intelligent tutoring systems.', *International Working Group on Educational Data Mining* .

Chan, C.-C. (2007), A framework for assessing usage of web-based e-learning systems, *in* 'Innovative Computing, Information and Control, 2007. I-CICIC'07. Second International Conference on', IEEE, pp. 147–147.

Chanchary, F. H., Haque, I. & Khalid, S. (2008), Web usage mining to evaluate the transfer of learning in a web-based learning environment, *in* 'Knowledge Discovery and Data Mining, 2008. WKDD 2008. First International Workshop on', IEEE, pp. 249–253.

Chang, C.-C. & Lin, C.-J. (2011), 'LIBSVM: A library for support vector machines', *ACM Transactions on Intelligent Systems and Technology* **2**, 27:1–27:27.

Chang, E. Y., Bai, H. & Zhu, K. (2009), Parallel algorithms for mining large-scale rich-media data, *in* 'Proceedings of the 17th ACM international conference on Multimedia', ACM, pp. 917–918.

Chang, K.-m., Beck, J., Mostow, J. & Corbett, A. (2006), A bayes net toolkit for student modeling in intelligent tutoring systems, *in* 'Intelligent Tutoring Systems', Springer, pp. 104–113.

Chang, Y.-C., Kao, W.-Y., Chu, C.-P. & Chiu, C.-H. (2009), 'A learning style classification mechanism for e-learning', *Computers & Education* **53**(2), 273–285.

Chen, C.-M. & Chen, M.-C. (2009), 'Mobile formative assessment tool based on data mining techniques for supporting web-based learning', *Computers & Education* **52**(1), 256–273.

Chen, C.-M., Duh, L.-J. & Liu, C.-Y. (2004), A personalized courseware recommendation system based on fuzzy item response theory, *in* 'e-Technology, e-Commerce and e-Service, 2004. EEE'04. 2004 IEEE International Conference on', IEEE, pp. 305–308.

Chen, G.-D., Liu, C.-C., Ou, K.-L., Liu, B.-J. et al. (2000), 'Discovering decision knowledge from web log portfolio for managing classroom processes by applying decision tree and data cube technology', *Journal of Educational Computing Research* **23**(3), 305–332.

Chen, N.-S., Wei, C.-W., Chen, H.-J. et al. (2008), 'Mining e-learning domain concept map from academic articles', *Computers & Education* **50**(3), 1009–1021.

Chen, Y.-C., Peng, W.-C. & Lee, S.-Y. (2012), 'Efficient algorithms for influence maximization in social networks', *Knowledge and information systems* **33**(3), 577–601.

Chen, Y.-L. & Weng, C.-H. (2009), 'Mining fuzzy association rules from questionnaire data', *Knowledge-Based Systems* **22**(1), 46–56.

Cheng, X., Miao, D., Wang, C. & Cao, L. (2013), Coupled term-term relation analysis for document clustering, *in* 'Proceedings of the 2013 International Joint Conference on Neural Networks', accepted.

Chu, H.-C., Hwang, G.-J., Tseng, J. C. & Hwang, G.-H. (2006), 'A computerized approach to diagnosing student learning problems in health education', *Asian Journal of Health and Information Sciences* **1**(1), 43–60.

Cocea, M. & Weibelzahl, S. (2007), Cross-system validation of engagement prediction from log files, *in* 'Creating new learning experiences on a global scale', Springer, pp. 14–25.

Collobert, R. & Weston, J. (2008), A unified architecture for natural language processing: Deep neural networks with multitask learning, *in* 'Proceedings of the 25th international conference on Machine learning', ACM, pp. 160–167.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K. & Kuksa, P. (2011), 'Natural language processing (almost) from scratch', *The Journal of Machine Learning Research* **12**, 2493–2537.

Cost, S. & Salzberg, S. (1993), 'A weighted nearest neighbor algorithm for learning with symbolic features', *Machine Learning* **10**(1), 57–78.

Crespo, R. M., Pardo, A., Pérez, J. P. S. & Kloos, C. D. (2005), An algorithm for peer review matching using student profiles based on fuzzy classification and genetic algorithms, *in* 'Innovations in Applied Artificial Intelligence', Springer, pp. 685–694.

Cristian, M. & Dan, B. (2006), Testing attribute selection algorithms for classification performance on real data, *in* 'Intelligent Systems, 2006 3rd International IEEE Conference on', IEEE, pp. 581–586.

Das, G. & Mannila, H. (2000), Context-based similarity measures for categorical databases, *in* 'Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases 2000', pp. 201–210.

Das, S. R. & Chen, M. Y. (2007), 'Yahoo! for amazon: Sentiment extraction from small talk on the web', *Management Science* **53**(9), 1375–1388.

Dave, K., Lawrence, S. & Pennock, D. M. (2003), Mining the peanut gallery: Opinion extraction and semantic classification of product reviews, *in* 'Proceedings of the 12th international conference on World Wide Web', ACM, pp. 519–528.

Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W. & Harshman, R. A. (1990), 'Indexing by latent semantic analysis', *JASIS* **41**(6), 391–407.

Dekker, G. W., Pechenizkiy, M. & Vleeshouwers, J. M. (2009), 'Predicting students drop out: A case study.', *International Working Group on Educational Data Mining* .

Desmarais, M. C. & Gagnon, M. (2006), *Bayesian student models based on item to item knowledge structures*, Springer.

Domingos, P. & Hulten, G. (2000), Mining high-speed data streams, *in* 'Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining', ACM, pp. 71–80.

Dong, G. & Pei, J. (2007), *Sequence data mining*, Vol. 33, Springer Science & Business Media.

dos Santos Machado, L. & Becker, K. (2003), Distance education: A web usage mining case study for the evaluation of learning sites, *in* 'Advanced Learning Technologies, 2003. Proceedings. The 3rd IEEE International Conference on', IEEE, pp. 360–361.

Drachsler, H., Hummel, H. G. & Koper, R. (2008), 'Personal recommender systems for learners in lifelong learning networks: the requirements, techniques and model', *International Journal of Learning Technology* **3**(4), 404–423.

Draper, N. R., Smith, H. & Pownell, E. (1966), *Applied regression analysis*, Vol. 3, Wiley New York.

Dringus, L. P. & Ellis, T. (2005), 'Using data mining as a strategy for assessing asynchronous discussion forums', *Computers & Education* **45**(1), 141–160.

Duan, K.-B. & Keerthi, S. S. (2005), Which is the best multiclass svm method? an empirical study, *in* 'Multiple Classifier Systems', Springer, pp. 278–285.

El-Kechaï, N. & Després, C. (2007), Proposing the underlying causes that lead to the trainees erroneous actions to the trainer, *in* 'Creating New Learning Experiences on a Global Scale', Springer, pp. 41–55.

Elman, J. L. (1991), 'Distributed representations, simple recurrent networks, and grammatical structure', *Machine learning* **7**(2-3), 195–225.

Espejo, P. G., Ventura, S. & Herrera, F. (2010), 'A survey on the application of genetic programming to classification', *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* **40**(2), 121–144.

Esuli, A. & Sebastiani, F. (2006), Sentiwordnet: A publicly available lexical resource for opinion mining, *in* 'Proceedings of LREC', Vol. 6, Citeseer, pp. 417–422.

Etchells, T. A., Nebot, À., Vellido, A., Lisboa, P. J. & Mugica, F. (2006), Learning what is important: feature selection and rule extraction in a virtual course., *in* 'ESANN', pp. 401–406.

Farzan, R. & Brusilovsky, P. (2006), Social navigation support in a course recommendation system, *in* 'Adaptive hypermedia and adaptive web-based systems', Springer, pp. 91–100.

Fausett, L. & Elwasif, W. (1994), Predicting performance from test scores using backpropagation and counterpropagation, *in* 'Neural Networks, 1994. IEEE World Congress on Computational Intelligence., 1994 IEEE International Conference on', Vol. 5, IEEE, pp. 3398–3402.

Fayyad, U., Piatetsky-Shapiro, G. & Smyth, P. (1996), 'From data mining to knowledge discovery in databases', *AI magazine* **17**(3), 37.

Feng, M. & Beck, J. (2009), 'Back to the future: A non-automated method of constructing transfer models.', *International Working Group on Educational Data Mining* .

Feng, M., Heffernan, N. T. & Koedinger, K. R. (2005), Looking for sources of error in predicting students knowledge, *in* 'Educational Data Mining: Papers from the 2005 AAAI Workshop', pp. 54–61.

Fok, A. W., Wong, H.-S. & Chen, Y. (2005), Hidden markov model based characterization of content access patterns in an e-learning environment, *in* 'Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on', IEEE, pp. 201–204.

Freyberger, J. E., Heffernan, N. & Ruiz, C. (2004), Using association rules to guide a search for best fitting transfer models of student learning, PhD thesis, Citeseer.

Frias-Martinez, E., Chen, S. Y. & Liu, X. (2006), 'Survey of data mining approaches to user modeling for adaptive hypermedia', *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* **36**(6), 734–749.

Gan, G., Ma, C. & Wu, J. (2007), *Data Clustering: Theory, Algorithms, and Applications*, ASA-SIAM Series on Statistics and Applied Probability, Philadelphia.

García, E., Romero, C., Ventura, S. & De Castro, C. (2009), 'An architecture for making recommendations to courseware authors using association rule mining and collaborative filtering', *User Modeling and User-Adapted Interaction* **19**(1-2), 99–132.

García, P., Amandi, A., Schiaffino, S. & Campo, M. (2007), 'Evaluating bayesian networks precision for detecting students learning styles', *Computers & Education* **49**(3), 794–808.

Gaudioso, E., Montero, M., Talavera, L. & Hernandez-del Olmo, F. (2009), 'Supporting teachers in collaborative student modeling: A framework and an implementation', *Expert Systems with Applications* **36**(2), 2260–2265.

Gedeon, T. & Turner, S. (1993), Explaining student grades predicted by a neural network, *in* 'Neural Networks, 1993. IJCNN'93-Nagoya. Proceedings of 1993 International Joint Conference on', Vol. 1, IEEE, pp. 609–612.

Go, A., Bhayani, R. & Huang, L. (2009), 'Twitter sentiment classification using distant supervision', *CS224N Project Report, Stanford* pp. 1–12.

Golding, P. & Donaldson, O. (2006), Predicting academic performance, *in* 'Frontiers in Education Conference, 36th Annual', IEEE, pp. 21–26.

Gong, Y., Rai, D., Beck, J. E. & Heffernan, N. T. (2009), 'Does self-discipline impact students' knowledge and learning?.', *International Working Group on Educational Data Mining* .

Grob, H. L., Bensberg, F. & Kaderali, F. (2004), Controlling open source intermediaries-a web log mining approach, *in* 'Information Technology Interfaces, 2004. 26th International Conference on', IEEE, pp. 233–242.

Guo, Q. & Zhang, M. (2009), 'Implement web learning environment based on data mining', *Knowledge-Based Systems* **22**(6), 439–442.

Ha, S. H., Bae, S. M. & Park, S. C. (2000), Web mining for distance education, *in* 'Management of Innovation and Technology, 2000. ICMIT 2000. Proceedings of the 2000 IEEE International Conference on', Vol. 2, IEEE, pp. 715–719.

Hadwin, A. F., Nesbit, J. C., Jamieson-Noel, D., Code, J. & Winne, P. H. (2007), 'Examining trace data to explore self-regulated learning', *Metacognition and Learning* **2**(2-3), 107–124.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. & Witten, I. H. (2009), 'The weka data mining software: an update', *ACM SIGKDD Explorations Newsletter* **11**(1), 10–18.

Hämäläinen, W., Suhonen, J., Sutinen, E. & Toivonen, H. (2004), Data mining in personalizing distance education courses, *in* 'World conference on open learning and distance education', p. 16.

Hämäläinen, W. & Vinni, M. (2006), Comparison of machine learning methods for intelligent tutoring systems, *in* 'Intelligent Tutoring Systems', Springer, pp. 525–534.

Hanna, M. (2004), 'Data mining in the e-learning domain', *Campus-wide information systems* **21**(1), 29–34.

Hardof-Jaffe, S., Hershkovitz, A., Abu-Kishk, H., Bergman, O. & Nachmias, R. (2009), 'How do students organize personal information spaces?.', *International Working Group on Educational Data Mining* .

Hatzivassiloglou, V. & McKeown, K. R. (1997), Predicting the semantic orientation of adjectives, *in* 'Proceedings of the 35th annual meeting of the association for computational linguistics and eighth conference of the european chapter of the association for computational linguistics', Association for Computational Linguistics, pp. 174–181.

Heathcote, E. A. & Dawson, S. P. (2005), 'Data mining for evaluation, benchmarking and reflective practice in a lms'.

Heathcote, E. A. & Prakash, S. (2007), 'What your learning management system is telling you about supporting your teachers: monitoring system information to improve support for teachers using educational technologies at queensland university of technology'.

Heraud, J.-M., France, L. & Mille, A. (2004), Pixed: An its that guides students with the help of learners' interaction log, *in* 'International Conference on Intelligent Tutoring Systems, Workshop Analyzing Student Tutor Interaction Logs to Improve Educational Outcomes. Maceio, Brazil', pp. 57–64.

Hernándeza, J.-A., Ochoab, A., Muñozd, J. & Burlaka, G. (2006), Detecting cheats in online student assessments using data mining, *in* 'Conference on Data Mining— DMIN', Vol. 6, Citeseer, p. 205.

Hershkovitz, A. & Nachmias, R. (2008), Developing a log-based motivation measuring tool., *in* 'EDM', pp. 226–233.

Hershkovitz, A. & Nachmias, R. (2009), 'Consistency of students' pace in online learning.', *International Working Group on Educational Data Mining* .

Hien, N. T. N. & Haddawy, P. (2007), A decision support system for evaluating international student applications, *in* 'Frontiers In Education Conference-Global Engineering: Knowledge Without Borders, Opportunities Without Passports, 2007. FIE'07. 37th Annual', IEEE, pp. F2A–1.

Hill, S. I. & Doucet, A. (2007), 'A framework for kernel-based multi-category classification.', *J. Artif. Intell. Res.(JAIR)* **30**, 525–564.

Hinton, G. E. & Salakhutdinov, R. R. (2006), 'Reducing the dimensionality of data with neural networks', *Science* **313**(5786), 504–507.

Hinton, G. E. & Sejnowski, T. J. (1986), 'Learning and relearning in boltzmann machines', *MIT Press, Cambridge, Mass* **1**, 282–317.

Hinton, G. & Salakhutdinov, R. (2011), 'Discovering binary codes for documents by learning deep generative models', *Topics in Cognitive Science* **3**(1), 74–91.

Hofmann, T. (2001), 'Unsupervised learning by probabilistic latent semantic analysis', *Machine learning* **42**(1-2), 177–196.

Houle, M., Oria, V. & Qasim, U. (2010), Active caching for similarity queries based on shared-neighbor information, *in* 'Proceedings of the 19th ACM International Conference on Information and Knowledge Management', pp. 669–678.

Hsia, T.-C., Shie, A.-J. & Chen, L.-C. (2008), 'Course planning of extension education to meet market demand by using data mining techniques–an example of chinkuo technology university in taiwan', *Expert Systems with applications* **34**(1), 596–602.

Hsu, C.-W., Chang, C.-C., Lin, C.-J. et al. (2003), 'A practical guide to support vector classification'.

Hsu, C.-W. & Lin, C.-J. (2002), 'A comparison of methods for multi-class support vector machines', *Neural Networks, IEEE Transactions on* **13**(2), 415–425.

Huang, C.-T., Lin, W.-T., Wang, S.-T. & Wang, W.-S. (2009), 'Planning of educational training courses by data mining: Using china motor corporation as an example', *Expert Systems with Applications* **36**(3), 7199–7209.

Huang, E. H., Socher, R., Manning, C. D. & Ng, A. Y. (2012), Improving word representations via global context and multiple word prototypes, *in* 'Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1', Association for Computational Linguistics, pp. 873–882.

Huang, J., Zhu, A. & Luo, Q. (2007), Personality mining method in web based education system using data mining, *in* 'Grey Systems and Intelligent Services, 2007. GSIS 2007. IEEE International Conference on', IEEE, pp. 155–158.

Huang, T.-C., Cheng, S.-C. & Huang, Y.-M. (2009), 'A blog article recommendation generating mechanism using an sbacpso algorithm', *Expert Systems with Applications* **36**(7), 10388–10396.

Hübscher, R., Puntambekar, S. & Nye, A. H. (2007), Domain specific interactive data mining, *in* 'Proceedings of Workshop on Data Mining for User Modeling at the 11th International Conference on User Modeling', pp. 81–90.

Hurley, T. & Weibelzahl, S. (2007), Using motsart to support on-line teachers in student motivation, *in* 'Creating New Learning Experiences on a Global Scale', Springer, pp. 101–111.

Hwang, G.-J. (2005), 'A data mining approach to diagnosing student learning problems in sciences courses', *International Journal of Distance Education Technologies (IJDET)* **3**(4), 35–50.

Hwang, G.-J., Tsai, P.-S., Tsai, C.-C. & Tseng, J. C. (2008), 'A novel approach for assisting teachers in analyzing student web-searching behaviors', *Computers & Education* **51**(2), 926–938.

Hwang, W.-Y., Chang, C.-B. & Chen, G.-J. (2004), 'The relationship of learning traits, motivation and performance-learning response dynamics', *Computers & Education* **42**(3), 267–287.

Ibrahim, Z. & Rusli, D. (2007), Predicting students academic performance: comparing artificial neural network, decision tree and linear regression, *in* '21st Annual SAS Malaysia Forum, 5th September'.

Ingram, A. L. (2000), 'Using web server logs in evaluating instructional web sites', *Journal of Educational Technology Systems* **28**(2), 137–158.

Jiawei, H. & Kamber, M. (2001), 'Data mining: concepts and techniques', *San Francisco, CA, itd: Morgan Kaufmann* **5**.

Jin, H., Wu, T., Liu, Z. & Yan, J. (2009), Application of visual data mining in higher-education evaluation system, *in* '2009 First International Workshop on Education Technology and Computer Science', Vol. 2, p-p. 101–104.

Joachims, T. (1999), ‘Making large scale svm learning practical’.

Jong, B.-S., Chan, T.-Y. & Wu, Y.-L. (2007), ‘Learning log explorer in e-learning diagnosis’, *Education, IEEE Transactions on* **50**(3), 216–228.

Jonsson, A., Johns, J., Mehranian, H., Arroyo, I., Woolf, B., Barto, A., Fisher, D. & Mahadevan, S. (2005), Evaluating the feasibility of learning student models from data, *in* ‘Educational Data Mining: Papers from the AAAI Workshop’, pp. 1–6.

Jovanović, J., Gašević, D., Brooks, C., Devedžić, V. & Hatala, M. (2007), Loco-analyst: A tool for raising teachers awareness in online learning environments, *in* ‘Creating New Learning Experiences on a Global Scale’, Springer, pp. 112–126.

Juan, A. A., Daradoumis, T., Faulin, J. & Xhafa, F. (2009), ‘Samos: a model for monitoring students’ and groups’ activities in collaborative e-learning’, *International Journal of Learning Technology* **4**(1), 53–72.

Karampiperis, P. & Sampson, D. (2005), ‘Adaptive learning resources sequencing in educational hypermedia systems’, *Educational Technology & Society* **8**(4), 128–147.

Kay, J., Maisonneuve, N., Yacef, K. & Zaïane, O. (2006), Mining patterns of events in students teamwork data, *in* ‘Proceedings of the Workshop on Educational Data Mining at the 8th International Conference on Intelligent Tutoring Systems (ITS 2006)’, pp. 45–52.

Kelly, D. & Tangney, B. (2005), ’first aid for you’: getting to know your learning style using machine learning, *in* ‘Advanced Learning Technologies, 2005. ICALT 2005. Fifth IEEE International Conference on’, IEEE, pp. 1–3.

Khajuria, S. (2007), A model to predict student matriculation from admissions data, PhD thesis, Ohio University.

Khribi, M. K., Jemni, M. & Nasraoui, O. (2008), Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval, *in* 'Advanced Learning Technologies, 2008. ICALT'08. Eighth IEEE International Conference on', IEEE, p-p. 241–245.

Kiang, M. Y., Fisher, D. M., Chen, J.-C. V., Fisher, S. A. & Chi, R. T. (2009), 'The application of som as a decision support tool tdo identify aacsb peer schools', *Decision Support Systems* **47**(1), 51–59.

Kim, J., Chern, G., Feng, D., Shaw, E. & Hovy, E. (2006), Mining and assessing discussions on the web through speech act analysis, *in* 'Proceedings of the Workshop on Web Content Mining with Human Language Technologies at the 5th International Semantic Web Conference'.

Kiu, C.-C. & Lee, C.-S. (2007), Learning objects reusability and retrieval through ontological sharing: A hybrid unsupervised data mining approach, *in* 'Advanced Learning Technologies, 2007. ICALT 2007. Seventh IEEE International Conference on', IEEE, pp. 548–550.

Koedinger, K. R., Cunningham, K., Skogsholm, A. & Leber, B. (2008), 'An open repository and analysis tools for fine-grained, longitudinal learner data.', *EDM* **157**, 166.

Kosba, E., Dimitrova, V. & Boyle, R. (2005), Using student and group models to support teachers in web-based distance education, *in* 'User Modeling 2005', Springer, pp. 124–133.

Kotsiantis, S. B., Pierrakeas, C. & Pintelas, P. E. (2003), Preventing student dropout in distance learning using machine learning techniques, *in* 'Knowledge-Based Intelligent Information and Engineering Systems', Springer, pp. 267–274.

Kotsiantis, S. B. & Pintelas, P. E. (2005), Predicting students marks in hellenic open university, *in* 'Advanced Learning Technologies, 2005. ICALT 2005. Fifth IEEE International Conference on', IEEE, pp. 664–668.

Krištofič, A. (2005), Recommender system for adaptive hypermedia applications, *in* 'IIT. SRC 2005: Student Research Conference', p. 229.

Lara, J. A., Lizcano, D., Martínez, M. A., Pazos, J. & Riera, T. (2014), 'A system for knowledge discovery in e-learning environments within the european higher education area–application to student data from open university of madrid, udima', *Computers & Education* **72**, 23–36.

Lee, C.-H., Lee, G.-G. & Leu, Y. (2009), 'Application of automatically constructed concept map of learning to conceptual diagnosis of e-learning', *Expert Systems with Applications* **36**(2), 1675–1684.

Lee, C.-S. (2007), 'Diagnostic, predictive and compositional modeling with data mining in integrated learning environments', *Computers & Education* **49**(3), 562–580.

Lee, M. W., Chen, S. Y., Chrysostomou, K. & Liu, X. (2009), 'Mining s-tudents behavior in web-based learning programs', *Expert Systems with Applications* **36**(2), 3459–3464.

Lee, M. W., Chen, S. Y. & Liu, X. (2007), Mining learners behavior in accessing web-based interface, *in* 'Technologies for E-Learning and Digital Entertainment', Springer, pp. 336–346.

Lemire, D., Boley, H., McGrath, S. & Ball, M. (2005), 'Collaborative filtering and inference rules for context-aware learning object recommendation', *Interactive Technology and Smart Education* **2**(3), 179–188.

Li, C. & Li, H. (2010), 'A survey of distance metrics for nominal attributes', *Journal of Software* **5**(11), 1262–1269.

Li, H. & Yamanishi, K. (2001), Mining from open answers in questionnaire data, *in* 'Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining', ACM, pp. 443–449.

Li, X., Luo, Q. & Yuan, J. (2007), Personalized recommendation service system in e-learning using web intelligence, *in* 'Computational Science–ICCS 2007', Springer, pp. 531–538.

Licchelli, O., Basile, T. M., Di Mauro, N., Esposito, F., Semeraro, G. & Ferilli, S. (2004), Machine learning approaches for inducing student models, *in* 'Innovations in Applied Artificial Intelligence', Springer, pp. 935–944.

Lin, F.-R., Hsieh, L.-S. & Chuang, F.-T. (2009), 'Discovering genres of online discussion threads via text mining', *Computers & Education* **52**(2), 481–495.

Liu, B. (2010), 'Sentiment analysis and subjectivity', *Handbook of natural language processing* **2**, 627–666.

Liu, C., Cao, L. & Yu, P. S. (2014), A hybrid coupled k-nearest neighbor algorithm on imbalance data, *in* 'Neural Networks (IJCNN), 2014 International Joint Conference on', IEEE, pp. 2011–2018.

Liu, F.-j. & Shih, B.-j. (2007), Learning activity-based e-learning material recommendation system, *in* 'Multimedia Workshops, 2007. ISMW'07. Ninth IEEE International Symposium on', IEEE, pp. 343–348.

Liu, W. & Wang, T. (2012), 'Online active multi-field learning for efficient email spam filtering', *Knowledge and Information Systems* **33**(1), 117–136.

Lu, F., Li, X., Liu, Q., Yang, Z., Tan, G. & He, T. (2007), Research on personalized e-learning system using fuzzy set based clustering algorithm, *in* 'Computational Science–ICCS 2007', Springer, pp. 587–590.

Lu, J. (2004), Personalized e-learning material recommender system, *in* 'International conference on information technology for application', pp. 374–379.

Lykourentzou, I., Giannoukos, I., Nikolopoulos, V., Mpardis, G. & Loumos, V. (2009), 'Dropout prediction in e-learning courses through the combination of machine learning techniques', *Computers & Education* **53**(3), 950–965.

Ma, Y., Liu, B., Wong, C. K., Yu, P. S. & Lee, S. M. (2000), Targeting the right students using data mining, *in* 'Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining', ACM, pp. 457–464.

Madhyastha, T. & Hunt, E. (2009), 'Mining diagnostic assessment data for concept similarity', *JEDM-Journal of Educational Data Mining* **1**(1), 72–91.

Markellou, P., Mousourouli, I., Spiros, S. & Tsakalidis, A. (2005), 'Using semantic web mining technologies for personalized e-learning experiences', *Proceedings of the web-based education* pp. 461–826.

Martinez, D. (2001), 'Predicting student outcomes using discriminant function analysis.'.

Mazza, R. (2009), *Introduction to information visualization*, Springer Science & Business Media.

Mazza, R. & Milani, C. (2004), Gismo: a graphical interactive student monitoring tool for course management systems, *in* 'TEL04 Technology Enhanced Learning04 International Conference', pp. 18–19.

McLaren, B. M., Koedinger, K. R., Schneider, M., Harrer, A. & Bollen, L. (2004), 'Bootstrapping novice data: Semi-automated tutor authoring using student log files'.

Merceron, A., Oliveira, C., Scholl, M. & Ullrich, C. (2004), Mining for content re-use and exchange-solutions and problems, *in* 'Poster Proceedings of the 3rd International Semantic Web Conference, ISWC2004', pp. 39–40.

Merceron, A. & Yacef, K. (2005), Educational data mining: a case study., *in* 'AIED', pp. 467–474.

Merceron, A. & Yacef, K. (2008), 'Interestingness measures for associations rules in educational data.', *EDM* **8**, 57–66.

Mikolov, T., Karafiát, M., Burget, L., Cernockỳ, J. & Khudanpur, S. (2010), Recurrent neural network based language model., *in* 'INTERSPEECH', pp. 1045–1048.

Mikolov, T., Kombrink, S., Burget, L., Cernocky, J. & Khudanpur, S. (2011), Extensions of recurrent neural network language model, *in* 'Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on', IEEE, pp. 5528–5531.

Miller, G. A. (1995), 'Wordnet: a lexical database for english', *Communications of the ACM* **38**(11), 39–41.

Minaei-Bidgoli, B., Kashy, D. A., Kortemeyer, G. & Punch, W. F. (2003), Predicting student performance: an application of data mining methods with an educational web-based system, *in* 'Frontiers in education, 2003. FIE 2003 33rd annual', Vol. 1, IEEE, pp. T2A–13.

Minaei-Bidgoli, B., Tan, P.-N. & Punch, W. F. (2004), Mining interesting contrast rules for a web-based educational system, *in* 'Machine Learning and Applications, 2004. Proceedings. 2004 International Conference on', IEEE, pp. 320–327.

Mnih, A. & Hinton, G. E. (2008), A scalable hierarchical distributed language model, *in* 'Advances in neural information processing systems', pp. 1081–1088.

Monk, D. (2005), 'Using data mining for e-learning decision making', *The Electronic Journal of e-Learning* **3**(1), 41–54.

Mor, E. & Minguillón, J. (2004), E-learning personalization based on itineraries and long-term navigational behavior, *in* 'Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters', ACM, pp. 264–265.

Mostow, J. & Beck, J. (2006), 'Some useful tactics to modify, map and mine data from intelligent tutors', *Natural Language Engineering* **12**(02), 195–208.

Mostow, J., Beck, J., Cen, H., Cuneo, A., Gouvea, E. & Heiner, C. (2005), An educational data mining tool to browse tutor-student interactions: Time will tell, *in* 'Proceedings of the Workshop on Educational Data Mining, National Conference on Artificial Intelligence', AAAI Press, Pittsburgh, PA, pp. 15–22.

Muehlenbrock, M. (2005), Automatic action analysis in an interactive learning environment, *in* 'The 12th international conference on artificial intelligence in education, AIED', pp. 73–80.

Myller, N., Suhonen, J. & Sutinen, E. (2002), Using data mining for improving web-based course design, *in* 'Computers in Education, International Conference on', IEEE Computer Society, pp. 959–959.

Myszkowski, P. B., Kwasnicka, H. & Markowska-Kaczmar, U. (2008), Data mining techniques in e-learning celgrid system, *in* 'Computer Information Systems and Industrial Management Applications, 2008. CISIM'08. 7th', IEEE, pp. 315–319.

Nagata, R., Takeda, K., Suda, K., Kakegawa, J. & Morihiro, K. (2009), 'Edu-mining for book recommendation for pupils.', *International Working Group on Educational Data Mining* .

Nankani, E., Simoff, S., Denize, S. & Young, L. (2009), Supporting strategic decision making in an enterprise university through detecting patterns of academic collaboration, *in* 'Information Systems: Modeling, Development, and Integration', Springer, pp. 496–507.

Nebot, A., Castro, F., Vellido, A. & Mugica, F. (2006), Identification of fuzzy models to predict students performance in an e-learning environment, *in* 'The Fifth IASTED International Conference on Web-Based Education, WBE', pp. 74–79.

Nesbit, J., Xu, Y., Winne, P. & Zhou, M. (2008), 'Sequential pattern analysis software for educational event data', *Measuring Behavior 2008* p. 160.

Nugent, R., Ayers, E. & Dean, N. (2009), 'Conditional subspace clustering of skill mastery: Identifying skills that separate students.', *International Working Group on Educational Data Mining* .

Ogor, E. N. (2007), Student academic performance monitoring and evaluation using data mining techniques, *in* 'Electronics, Robotics and Automotive Mechanics Conference, 2007. CERMA 2007', IEEE, pp. 354–359.

Oladokun, V., Adebanjo, A. & Charles-Owaba, O. (2008), 'Predicting s-tudents academic performance using artificial neural network: A case study of an engineering course', *The Pacific Journal of Science and Technology* **9**(1), 72–79.

Orzechowski, T., Ernst, S. & Dziech, A. (2007), Profiled search methods for e-learning systems., *in* 'LODE'.

Pahl, C. & Donnellan, D. (2002), 'Data mining technology for the evaluation of web-based teaching and learning systems'.

Pak, A. & Paroubek, P. (2010), Twitter as a corpus for sentiment analysis and opinion mining., *in* 'LREC', Vol. 10, pp. 1320–1326.

Pang, B. & Lee, L. (2004), A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts, *in* 'Proceedings of the 42nd annual meeting on Association for Computational Linguistics', Association for Computational Linguistics, p. 271.

Pang, B. & Lee, L. (2008), 'Opinion mining and sentiment analysis', *Foundations and trends in information retrieval* **2**(1-2), 1–135.

Pang, B., Lee, L. & Vaithyanathan, S. (2002), Thumbs up?: sentiment classification using machine learning techniques, *in* 'Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10', Association for Computational Linguistics, pp. 79–86.

Papadimitriou, C. H., Tamaki, H., Raghavan, P. & Vempala, S. (1998), Latent semantic indexing: A probabilistic analysis, *in* 'Proceedings of the seventeenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems', ACM, pp. 159–168.

Pardos, Z. A., Beck, J. E., Ruiz, C. & Heffernan, N. T. (2008), The composition effect: Conjunctive or compensatory? an analysis of multi-skill math questions in its, *in* 'THE 1ST INTERNATIONAL CONFERENCE ON EDUCATIONAL DATA MINING', Citeseer.

Pardos, Z. A. & Heffernan, N. T. (2009), 'Determining the significance of item order in randomized problem sets.', *International Working Group on Educational Data Mining* .

Pardos, Z. A., Heffernan, N. T., Anderson, B. & Heffernan, C. L. (2007), The effect of model granularity on student performance prediction using bayesian networks, *in* 'User Modeling 2007', Springer, pp. 435–439.

Pavlik Jr, P. I., Cen, H. & Koedinger, K. R. (2009), 'Learning factors transfer analysis: Using learning curve analysis to automatically generate domain models.', *Online Submission* .

Pechenizkiy, M., Calders, T., Vasilyeva, E. & De Bra, P. (2008), Mining the student assessment data: Lessons drawn from a small scale case study., *in* 'EDM', pp. 187–191.

Pechenizkiy, M., Trcka, N., Vasilyeva, E., van der Aalst, W. & De Bra, P. (2009), 'Process mining online assessment data.', *International Working Group on Educational Data Mining* .

Perera, D., Kay, J., Koprinska, I., Yacef, K. & Zaïane, O. R. (2009), 'Clustering and sequential pattern mining of online collaborative learning data', *Knowledge and Data Engineering, IEEE Transactions on* **21**(6), 759–772.

Pollack, J. B. (1990), 'Recursive distributed representations', *Artificial Intelligence* **46**(1), 77–105.

Prata, D. N., d Baker, R. S., Costa, E. d. B., Rosé, C. P., Cui, Y. & de Carvalho, A. M. (2009), 'Detecting and understanding the impact of cognitive and interpersonal conflict in computer supported collaborative learning environments.', *International Working Group on Educational Data Mining* .

Pritchard, D. & Warnakulasooriya, R. (2005), Data from a web-based homework tutor can predict students final exam score, *in* 'World Conference on Educational Multimedia, Hypermedia and Telecommunications', Vol. 2005, pp. 2523–2529.

Psaromiligkos, Y., Orfanidou, M., Kytagias, C. & Zafiri, E. (2011), 'Mining log data for the analysis of learners behaviour in web-based learning management systems', *Operational Research* **11**(2), 187–200.

Quevedo, J. & Montañés, E. (2009), 'Obtaining rubric weights for assessments by more than one lecturer using a pairwise learning model.', *International Working Group on Educational Data Mining* .

Rahkila, M. & Karjalainen, M. (1999), Evaluation of learning in computer based education using log systems, *in* 'Frontiers in Education Conference, 1999. FIE'99. 29th Annual', Vol. 1, IEEE, pp. 12A3–16.

Rai, D., Gong, Y. & Beck, J. E. (2009), 'Using dirichlet priors to improve model parameter plausibility.', *International Working Group on Educational Data Mining* .

Ramli, A. (2005), Web usage mining using apriori algorithm: Uum learning care portal case, *in* 'International conference on knowledge management, Malaysia', pp. 1–19.

Ranjan, J. & Khalil, S. (2008), 'Conceptual framework of data mining process in management education in india: An institutional perspective', *Information Technology Journal* **7**(1), 16–23.

Reffay, C. & Chanier, T. (2003), How social network analysis can help to measure cohesion in collaborative distance-learning, *in* 'Designing for change in networked learning environments', Springer, pp. 343–352.

Retalis, S., Papasalouros, A., Psaromiligkos, Y., Siscos, S. & Kargidis, T. (2006), Towards networked learning analytics–a concept and a tool, *in* 'Proceedings of the fifth international conference on networked learning'.

Reyes, A., Rosso, P. & Buscaldi, D. (2012), 'From humor recognition to irony detection: The figurative language of social media', *Data & Knowledge Engineering* **74**, 1–12.

Reyes, P. & Tchounikine, P. (2005), Mining learning groups' activities in forum-type tools, *in* 'Proceedings of th 2005 conference on Computer support for collaborative learning: learning 2005: the next 10 years!', International Society of the Learning Sciences, pp. 509–513.

Rifkin, R. & Klautau, A. (2004), 'In defense of one-vs-all classification', *The Journal of Machine Learning Research* **5**, 101–141.

Ritter, S., Harris, T. K., Nixon, T., Dickison, D., Murray, R. C. & Towle, B. (2009), 'Reducing the knowledge tracing space.', *International Working Group on Educational Data Mining* .

Robinet, V., Bisson, G., Gordon, M. & Lemaire, B. (2007), Searching for student intermediate mental steps, *in* '11th International Conference on User Modeling', pp. 35–39.

Romero, C., Santos, S. G., Freire, M. & Ventura, S. (2008), Mining and visualizing visited trails in web-based educational systems., *in* 'EDM', pp. 182–186.

Romero, C. & Ventura, S. (2006), *Data mining in e-learning*, Wit Press.

Romero, C. & Ventura, S. (2007), 'Educational data mining: A survey from 1995 to 2005', *Expert systems with applications* **33**(1), 135–146.

Romero, C. & Ventura, S. (2010), 'Educational data mining: a review of the state of the art', *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* **40**(6), 601–618.

Romero, C., Ventura, S. & De Bra, P. (2004), 'Knowledge discovery with genetic programming for providing feedback to courseware authors', *User Modeling and User-Adapted Interaction* **14**(5), 425–464.

Romero, C., Ventura, S. & García, E. (2008), 'Data mining in course management systems: Moodle case study and tutorial', *Computers & Education* **51**(1), 368–384.

Romero, C., Ventura, S., García, E. & de Castro, C. (2009), Collaborative data mining tool for education, *in* 'Educational Data Mining 2009'.

Romero, C., Ventura, S., Pechenizkiy, M. & Baker, R. S. (2010), *Handbook of educational data mining*, CRC Press.

Romero, C., Ventura, S., Zafra, A. & De Bra, P. (2009), 'Applying web usage mining for personalizing hyperlinks in web-based adaptive educational systems', *Computers & Education* **53**(3), 828–840.

Romesburg, C. (2004), *Cluster analysis for researchers*, Lulu. com.

Rus, V., Lintean, M. & Azevedo, R. (2009), 'Automatic detection of student mental models during prior knowledge activation in metatutor.', *International Working Group on Educational Data Mining* .

Sanjeev, A. P. & Zytkow, J. M. (1995), Discovering enrollment knowledge in university databases., *in* 'KDD', pp. 246–251.

Schönbrunn, K. & Hilbert, A. (2007), Data mining in higher education, *in* 'Advances in Data Analysis', Springer, pp. 489–496.

Selmoune, N. & Alimazighi, Z. (2008), A decisional tool for quality improvement in higher education, *in* 'Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on', IEEE, pp. 1–6.

Shangping, D. & Ping, Z. (2008), A data mining algorithm in distance learning, *in* 'Computer Supported Cooperative Work in Design, 2008. C-SCWD 2008. 12th International Conference on', IEEE, pp. 1014–1017.

Sheard, J., Ceddia, J., Hurst, J. & Tuovinen, J. (2003), 'Inferring student learning behaviour from website interactions: A usage analysis', *Education and Information Technologies* **8**(3), 245–266.

Shen, L.-p. & Shen, R.-m. (2004), Learning content recommendation service based-on simple sequencing specification, *in* 'Advances in Web-Based Learning–ICWL 2004', Springer, pp. 363–370.

Shen, R., Yang, F. & Han, P. (2002), Data analysis center based on e-learning platform, *in* 'The Internet Challenge: Technology and Applications', Springer, pp. 19–28.

Simko, M. & Bieliková, M. (2009), 'Automatic concept relationships discovery for an adaptive e-course.', *International Working Group on Educational Data Mining* .

Singley, M. K. & Lam, R. B. (2005), The classroom sentinel: supporting data-driven decision-making in the classroom, *in* 'Proceedings of the 14th international conference on World Wide Web', ACM, pp. 315–321.

Song, D., Lin, H. & Yang, Z. (2007), Opinion mining in e-learning system, *in* 'Network and Parallel Computing Workshops, 2007. NPC Workshops. IFIP International Conference on', IEEE, pp. 788–792.

Spacco, J., Winters, T. & Payne, T. (2006), Inferring use cases from unit testing, *in* 'AAAI workshop on educational data mining', pp. 1–7.

Stamper, J. & Barnes, T. (2009), 'Unsupervised mdp value selection for automating its capabilities.', *International Working Group on Educational Data Mining* .

Stevens, R., Soller, A., Giordani, A., Gerosa, L., Cooper, M. & Cox, C. (2005), Developing a framework for integrating prior problem solving and knowledge sharing histories of a group to predict future group performance, *in* 'Collaborative Computing: Networking, Applications and Worksharing, 2005 International Conference on', IEEE, pp. 9–pp.

Su, K., Huang, H., Wu, X. & Zhang, S. (2006), 'A logical framework for identifying quality knowledge from different data sources', *Decision Support Systems* **42**(3), 1673–1683.

Su, Z., Song, W., Lin, M. & Li, J. (2008), Web text clustering for personalized e-learning based on maximal frequent itemsets, *in* 'Computer Science and Software Engineering, 2008 International Conference on', Vol. 6, IEEE, pp. 452–455.

Superby, J.-F., Vandamme, J. & Meskens, N. (2006), Determination of factors influencing the achievement of the first-year university students using data mining methods, *in* 'Workshop on Educational Data Mining', pp. 37–44.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K. & Stede, M. (2011), 'Lexicon-based methods for sentiment analysis', *Computational linguistics* **37**(2), 267–307.

Talavera, L. & Gaudioso, E. (2004), Mining student data to characterize similar behavior groups in unstructured collaboration spaces, *in* 'Workshop on artificial intelligence in CSCL. 16th European conference on artificial intelligence', pp. 17–23.

Tang, T. Y. & McCalla, G. (2002), Student modeling for a web-based learning environment: a data mining approach, *in* 'AAAI/IAAI', pp. 967–968.

Tang, T. Y. & McCalla, G. (2003), Smart recommendation for an evolving e-learning system, *in* 'Workshop on Technologies for Electronic Documents for Supporting Learning, AIED'.

Teknomo, K. (2006), 'K-means clustering tutorial', *Medicine* **100**(4), 3.

Thomas, E. H. & Galambos, N. (2004), 'What satisfies students? mining student-opinion data with regression and decision tree analysis', *Research in Higher Education* **45**(3), 251–269.

Tian, F., Wang, S., Zheng, C. & Zheng, Q. (2008), Research on e-learner personality grouping based on fuzzy clustering analysis, *in* 'Computer Supported Cooperative Work in Design, 2008. CSCWD 2008. 12th International Conference on', IEEE, pp. 1035–1040.

Ting, I.-H., Ouyang, Y. & Zhu, M. (2008), 'elorm: learning object relationship mining-based repository', *Online Information Review* **32**(2), 254–265.

Tsai, C.-J., Tseng, S.-S. & Lin, C.-Y. (2001), A two-phase fuzzy mining and learning algorithm for adaptive learning environment, *in* 'Computational Science-ICCS 2001', Springer, pp. 429–438.

Tsantis, L. & Castellani, J. (2001), 'Enhancing learning environments through solution-based knowledge discovery tools: Forecasting for self-perpetuating systemic reform', *Journal of Special Education Technology* **16**(4), 39–52.

Tseng, S.-S., Sue, P.-C., Su, J.-M., Weng, J.-F. & Tsai, W.-N. (2007), 'A new approach for constructing the concept map', *Computers & Education* **49**(3), 691–707.

Turian, J., Ratinov, L. & Bengio, Y. (2010), Word representations: a simple and general method for semi-supervised learning, *in* 'Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics', Association for Computational Linguistics, pp. 384–394.

Turney, P. D. (2002), Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews, *in* 'Proceedings of the 40th annual meeting on association for computational linguistics', Association for Computational Linguistics, pp. 417–424.

Ueno, M. (2004), Data mining and text mining technologies for collaborative learning in an ilms" ssamurai", *in* 'Advanced Learning Technologies, 2004. Proceedings. IEEE International Conference on', IEEE, pp. 1052–1053.

Ueno, M. & Nagaoka, K. (2002), Learning log database and data mining system for e-learning–on-line statistical outlier detection of irregular learning processes, *in* 'IEEE International Conference on Advanced Learning Technologies (ICALT 2002) Proceedings'.

Vee, M., Meyer, B. & Mannock, K. L. (2006), Understanding novice errors and error paths in object-oriented programming through log analysis, *in*

‘Proceedings of workshop on educational data mining at the 8th international conference on intelligent tutoring systems (ITS 2006)’, pp. 13–20.

Vellido, A., Castro, F., Etchells, T. A., Nebot, À. & Mugica, F. (2007), Data mining of virtual campus data, *in* ‘Evolution of Teaching and Learning Paradigms in Intelligent Environment’, Springer, pp. 223–254.

Ventura, S., Romero, C. & Hervás, C. (2008), Analyzing rule evaluation measures with educational datasets: A framework to help the teacher., *in* ‘EDM’, pp. 177–181.

Vialardi, C., Bravo, J. & Ortigosa, A. (2008), ‘Improving aeh courses through log analysis’, *Journal of Universal Computer Science* **14**(17), 2777–2798.

Vialardi, C., Bravo, J., Shafti, L. & Ortigosa, A. (2009), ‘Recommendation in higher education using data mining techniques.’, *International Working Group on Educational Data Mining* .

Viola, S. R., Graf, S., Leo, T. et al. (2006), Analysis of felder-silverman index of learning styles by a data-driven statistical approach, *in* ‘Multimedia, 2006. ISM’06. Eighth IEEE International Symposium on’, IEEE, pp. 959–964.

Vranic, M., Pintar, D. & Skocir, Z. (2007), The use of data mining in education environment, *in* ‘Telecommunications, 2007. ConTel 2007. 9th International Conference on’, IEEE, pp. 243–250.

Wang, C., Cao, L., Li, J., Wei, W. & Ou, Y. (2012), Coupled nominal similarity analysis in unsupervised learning, *in* ‘CIKM 2012’.

Wang, C., Cao, L., Wang, M., Li, J., Wei, W. & Ou, Y. (2011), Coupled nominal similarity in unsupervised learning, *in* ‘Proceedings of the 20th ACM Conference on Information and Knowledge Management’, pp. 973–978.

Wang, C., She, Z. & Cao, L. (2013*a*), Coupled attribute analysis on numerical data, *in* 'Proceedings of the 23rd International Joint Conference on Artificial Intelligence', p. accepted.

Wang, C., She, Z. & Cao, L. (2013*b*), Coupled clustering ensemble: incorporating coupling relationships both between base clusterings and objects, *in* 'Proceedings of the 29th IEEE International Conference on Data Engineering', p. accepted.

Wang, F.-H. (2002), On using data-mining technology for browsing log file analysis in asynchronous learning environment, *in* 'World Conference on Educational Multimedia, Hypermedia and Telecommunications', Vol. 2002, pp. 2005–2006.

Wang, F.-H. (2008), 'Content recommendation based on education-contextualized browsing events for web-based personalized learning.', *Educational Technology & Society* **11**(4), 94–112.

Wang, F.-H. & Shao, H.-M. (2004), 'Effective personalized recommendation based on time-framed navigation clustering and association mining', *Expert Systems with Applications* **27**(3), 365–377.

Wang, W., Weng, J.-F., Su, J.-M. & Tseng, S.-S. (2004), Learning portfolio analysis and mining in scorm compliant environment, *in* 'Frontiers in Education, 2004. FIE 2004. 34th Annual', IEEE, pp. T2C–17.

Wang, Y.-h., Tseng, M.-H. & Liao, H.-C. (2009), 'Data mining for adaptive learning sequence in english language instruction', *Expert Systems with Applications* **36**(4), 7681–7686.

Wang, Y.-T., Cheng, Y.-H., Chang, T.-C. & Jen, S. (2008), On the application of data mining technique and genetic algorithm to an automatic course scheduling system, *in* 'Cybernetics and Intelligent Systems, 2008 IEEE Conference on', IEEE, pp. 400–405.

Wen-Shung Tai, D., Wu, H.-J. & Li, P.-H. (2008), 'Effective e-learning recommendation system based on self-organizing maps and association mining', *The Electronic Library* **26**(3), 329–344.

Wilson, D. & Martinez, T. (1997), 'Improved heterogeneous distance functions', *Journal of Artificial Intelligence Research* **6**, 1–34.

Winters, T., Shelton, C., Payne, T. & Mei, G. (2005), Topic extraction from item-level grades, *in* 'American Association for Artificial Intelligence 2005 Workshop on Educational Datamining, Pittsburgh, PA', Vol. 1, p. 3.

Wu, A. K. & Leung, C. (2002), Evaluating learning behavior of web-based training (wbt) using web log, *in* 'Computers in Education, 2002. Proceedings. International Conference on', IEEE, pp. 736–737.

Wu, X., Zhang, C. & Zhang, S. (2005), 'Database classification for multi-database mining', *Information Systems* **30**(1), 71–88.

Wu, X. & Zhang, S. (2003), 'Synthesizing high-frequency rules from different data sources', *Knowledge and Data Engineering, IEEE Transactions on* **15**(2), 353–367.

Yang, F., Han, P., Shen, R. & Hu, Z. (2005), A novel resource recommendation system based on connecting to similar e-learners, *in* 'Advances in Web-Based Learning–ICWL 2005', Springer, pp. 122–130.

Yang, T., Cao, L. & Zhang, C. (2010), A novel prototype reduction method for the k-nearest neighbor algorithm with $k \geq 1$, *in* 'PAKDD 2010', pp. 89–100.

Yang, T., Lin, T. & Wu, K. (2002), 'An agent-based recommender system for lesson plan sequencing', *Proceedings of 2nd ICALT, Kazan, Russia (September 2002)* .

Yoo, J., Yoo, S., Lance, C. & Hankins, J. (2006), Student progress monitoring tool using treeview, *in* 'ACM SIGCSE Bulletin', Vol. 38, ACM, pp. 373–377.

Yu, C. H., Digangi, S., Jannasch-Pennell, A. K. & Kaprolet, C. (2008), 'Profiling students who take online courses using data mining methods', *Online Journal of Distance Learning Administration* **11**(2).

Yu, P., Own, C. & Lin, L. (2001), On learning behavior analysis of web based interactive environment, *in* 'Proc. of the Int. Conf. on Implementing Curricular Change in Engineering Education', pp. 1–10.

Yudelson, M. V., Medvedeva, O., Legowski, E., Castine, M., Jukic, D. & Rebecca, C. (2006), Mining student learning data to develop high level pedagogic strategy in a medical its, *in* 'AAAI Workshop on Educational Data Mining', pp. 1–8.

Zafra, A. & Ventura, S. (2009), 'Predicting student grades in learning management systems with multiple instance genetic programming.', *International Working Group on Educational Data Mining* .

Zaíane, O. R. (2002), Building a recommender agent for e-learning systems, *in* 'Computers in Education, 2002. Proceedings. International Conference on', IEEE, pp. 55–59.

Zaïane, O. R. & Luo, J. (2001), Web usage mining for a better web-based learning environment, *in* 'Proceedings of conference on advanced technology for education', pp. 60–64.

Zakrzewska, D. (2008), Cluster analysis for users modeling in intelligent e-learning systems, *in* 'New Frontiers in Applied Artificial Intelligence', Springer, pp. 209–214.

Zhang, K., Cui, L., Wang, H. & Sui, Q. (2007), An improvement of matrix-based clustering method for grouping learners in e-learning, *in* 'Computer Supported Cooperative Work in Design, 2007. CSCWD 2007. 11th International Conference on', IEEE, pp. 1010–1015.

Zhang, L., Liu, X. & Liu, X. (2008), Personalized instructing recommendation system based on web mining, *in* 'Young Computer Scientists, 2008. ICYCS 2008. The 9th International Conference for', IEEE, pp. 2517–2521.

Zhang, X., Mostow, J., Duke, N., Trotochaud, C., Valeri, J. & Corbett, A. T. (2008), Mining free-form spoken responses to tutor prompts., *in* 'EDM', pp. 234–241.

Zheng, S., Xiong, S., Huang, Y. & Wu, S. (2008), Using methods of association rules mining optimizationin in web-based mobile-learning system, *in* 'Electronic Commerce and Security, 2008 International Symposium on', IEEE, pp. 967–970.

Zhong, S. (2005), Efficient online spherical k-means clustering, *in* 'Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on', Vol. 5, IEEE, pp. 3180–3185.

Zhu, F., Ip, H. H., Fok, A. W. & Cao, J. (2008), Peres: A personalized recommendation education system based on multi-agents & scorm, *in* 'Advances in Web Based Learning–ICWL 2007', Springer, pp. 31–42.

Zinn, C. & Scheuer, O. (2006), Getting to know your student in distance learning contexts, *in* 'Innovative Approaches for Learning and Knowledge Sharing', Springer, pp. 437–451.

Zorrilla, M. E., Menasalvas, E., Marin, D., Mora, E. & Segovia, J. (2005), Web usage mining project for improving web-based learning sites, *in* 'Computer Aided Systems Theory–EUROCAST 2005', Springer, pp. 205–210.

Zoubek, L. & Burda, M. (2009), 'Visualization of differences in data measuring mathematical skills.', *International Working Group on Educational Data Mining* .

Zukhri, Z. & Omar, K. (2008), ‘Solving new student allocation problem with genetic algorithm: a hard problem for partition based approach’, *International Journal of Soft Computing Applications. Euro Journal Publishing Inc* pp. 6–15.