Mining Special Features to Improve the Performance of Product Selection in E-commerce Environment and Resume Extraction System

Thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science (by Research)
in
Computer Science

by

Sumit Maheshwari
200402041
sumitm@research.iiit.ac.in
sumitmaheshwari.com@gmail.com

Center for Data Engineering
International Institute of Information Technology
Hyderabad, India
January 2010

INTERNATIONAL INSTITUTE OF INFORMATION TECHNOLOGY
Hyderabad, India

CERTIFICATE

It is certified that the work contained in this thesis, titled "Mining Special Features to Improve the Performance of Product Selection in E-commerce Environment and Resume Extraction System" by Sumit Maheshwari, has been carried out under my supervision and is not submitted elsewhere for a degree.

Date

Advisor: Prof. P. Krishna. Reddy

To my Father ***Mr. Shiv Shankar Maheshwari***
and Mother ***Mrs. Saroj Maheshwari***
for their everlasting love and support.

Acknowledgements

First and foremost, all praise to God who gave me all the help, guidance, and courage to finish my work. May God help me to convey all what I learned for the benefit of the society.

I am indebted to many people without whom this work would not be possible. First and foremost, I would like to thank my advisor, P. Krishna Reddy, for his guidance, suggestions and constructive criticism throughout this project. Through his support, care, and patience, he has transformed a normal graduate student into an experienced researcher. His insight and ideas formed the foundation of this dissertation as much as mine did, and his guidance and care helped me get over various hurdles during my graduate years. He not only gave me technical support and supervision that a graduate student could expect from his supervisor, but he also encouraged and gave moral support without which I would have never made it this far.

A list that alas has far too many names on it to mention separately is that of all researchers and colleagues at IT in Agriculture Lab and Center of Data Engineering Lab (IIIT-H)- my working place. It was pleasure, fun and stimulating of being in such an environment and I am definitely sure that I am going to miss all of them.

I would also like to give a special thanks to some of my friends Gaurav, Pratyush, Anshul, Sandeep, Swati, Ankit, Abhishek, Poonam, Avinash, Sushanta and many more who were always with we in my ups and downs specially during this period of one and half year. Thanks for all of the encouragement and support throughout this time. Without these people the journey would have been incomplete.

I would like to thank my parents for being with me even from before the beginning, and sometimes giving everything they have and more. Thank you for all the love, support and guidance and for always believing in me. I am also thankful for the support of my brother Amit and all the other members of my family.

Abstract

In E-commerce environment, a customer faces several difficulties while selecting the product. In this thesis we have investigated the issue of product selection and proposed an improved framework to help the customers in efficient product selection. In addition, we have studied the problem of resume selection and proposed an improved approach to help the Human Resource (HR) managers for selecting the appropriate candidate.

We first discuss the contribution related to the problem of product selection.

Suppose a customer wants to buy a Nokia mobile phone. To select a mobile phone, the customer has to carefully go through the features of about 70 Nokia mobile phone models, where each mobile phone description consists of around 30 features. As the different variants of mobile phones are available, the customer has to manually browse several Web pages to identify the specialty of each mobile phone. Even after spending sufficient amount of time, sometimes, a customer may not be confident of his/her choice.

Customers face the similar problem in several scenarios such as selecting a laptop, television, iPod camera and so on.

We have investigated the issue of selecting the appropriate product from the group of similar products and proposed an improved framework for the efficient product selection. It can be observed that every product has some specialty and possesses corresponding special features. The basic idea is as follows: a customer is shown special feature of each product along with the common features. As a result, the customer can quickly browse through all the special features and make the appropriate selection. We propose a method to compute the specialness value of features. On the basis of specialness value, we propose a clustering algorithm to organize the features into two-level and three-level organization of features.

We have conducted the experiments on three real world data-sets related to Nokia-mobile phones, Sony-cameras and HP-laptops for the validation of proposed framework. The results indicate that the proposed approach has a potential to reduce the number of features customers need to browse for selecting a product and thus help customers to select an appropriate product from a set of similar products efficiently.

The contribution related to the problem of resume selection is discussed as follows: Big enterprises and corporations receive thousands of resumes every day. Currently available techniques or services employed by these enterprises help their HR managers to filter from thousands of resumes to

hundred potential ones. Since these filtered resumes are similar to each other, they have to manually look through each resume to select the appropriate candidate. We have investigated the problem of resume selection from set of similar resumes and proposed an efficient framework to solve the same.

We have used the notion of special features as proposed above to come up with a solution to resume selection problem. It can be observed that normally resume information is described in a hierarchical structure with several sections such as education, experience, skills and achievements. There may exist special information present in some resumes when compared to others. For example resume may contain specialty in education, specialty in experience, special skills or special achievements. In this work we have experimented with extracting 'special skills' from a given set of similar resumes to ease the process of resume selection. It can be observed that all the resumes have some common skills and each resume might contain some special skills when compared to others. The proposed approach identifies the special skills (if present) in each resume and organizes the skills information in an efficient manner to help the HR managers select resumes efficiently.

We have performed experiments on real world data-set of resumes and obtained encouraging results. The results show that it is possible to select resume on the basis of special skills present in each resume.

Overall in this thesis, we have proposed the notion of special features and shown that how the knowledge of special feature could help the users in product selection in E-commerce environment and help the HR managers to select appropriate candidate through filtering of resumes based on the skills present in each resume.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The Internet and the World Wide Web have revolutionized our daily lives and the way business is conducted. The amount of trade conducted electronically has grown extraordinarily with widespread Internet usage. E-commerce has been carried out by companies through Internet/WWW. The basic E-commerce means conducting business on the Internet. It is mostly referred to buying and selling items on-line. This has often been described as ideal for the consumers, who are expected to have all the choices in the world conveniently and inexpensively on the screens of their computers.

E-commerce is driven on both buyer and supplier sides by a number of factors: access to an affluent customer base, lower information dissemination costs, lower transaction costs, broader market reach, and increased service, additional channels for customer feedback, and consumer and market research. But there are several issues that are often overlooked.

The introduction of E-commerce has lead to the problem of information overload for customers. While E-commerce has not necessarily allowed businesses to produce more products, it has allowed them to provide consumers with more choices. For example, almost daily a new mobile phone is introduced in the market. Since each mobile phone has number of features and there are hundreds of mobile phones to choose from, customers are drowned in large amount of information. Even though information is a valuable resource, but overwhelming information can pose a serious problem of time wastage in sorting through the disorganized textual mass for the relevant information. The other issues that are being faced in E-commerce are to provide choices for the customer that matches their needs, to build credibility and long term relationships with the customers, use the rich data collected by online business to promote products appropriate to specific consumers, to provide the customers with professional support to make efficient choices etc.

The systems like Recommendation systems, top-N lists, book and movie recommenders, advanced

search engines and intelligent avatars are being developed to address the above mentioned issues. The forms of recommendation include suggesting products to the consumer, providing personalized product information, summarizing community opinion, and providing community critiques.

In E-commerce environment a customer faces several difficulties while selecting the product. In this thesis we have investigated the issue of product selection and proposed an improved framework to help the customers in efficient product selection. In addition, we have studied the problem of resume selection and proposed an improved approach to help the Human Resource (HR) managers for selecting the appropriate candidate.

In this chapter we give an overview on traditional commerce and electronic commerce, discuss the issues that are being faced in e-commerce along with the research efforts to address the issues. Then we give an overview of the proposed approach in the thesis. Finally we mention the major contributions made in the thesis and organization of the thesis.

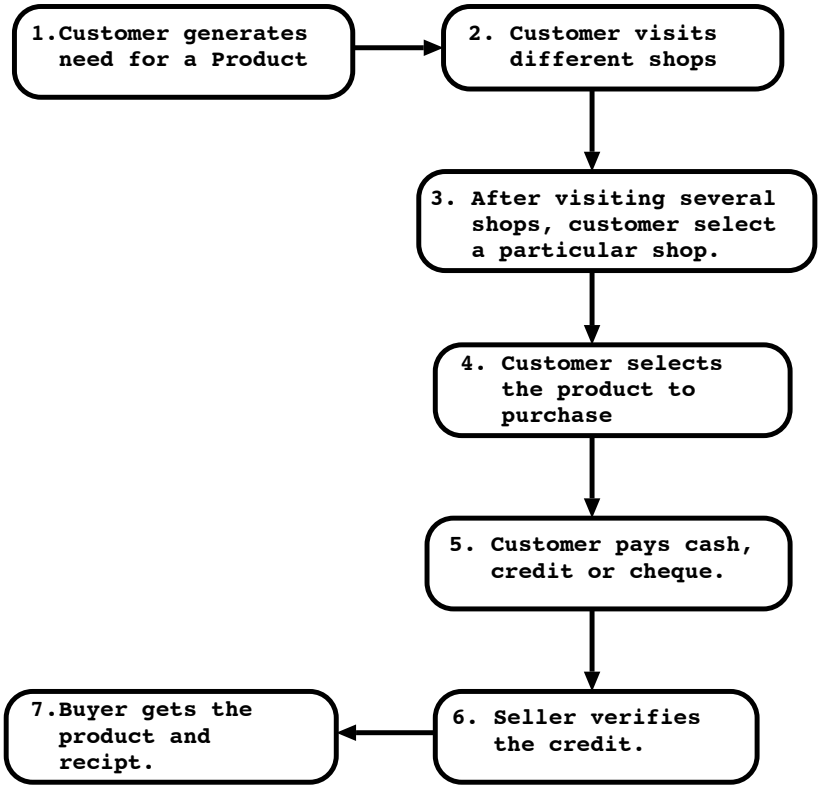

Figure 1.1: Traditional Commerce

1.1 Traditional Commerce

Traditionally commerce was defined as the exchange of valuable objects or services between at least two parties (barter system). It included all activities that each party undertakes to complete the transaction. In Traditional commerce products or services are physical, the process of the transaction is physical and the delivery agent is physical. For example a corner shop stocks newspapers that are bought with cash over the counter and are taken away by the customer out of the shop.

The traditional commerce consists of following steps (refer Figure 1.1): Firstly consumer identify a specific need and searches for products or services that will satisfy its specific need. Then the consumer visits different shops that sell the needed product or provide the required services. After visiting several of them, consumer finally selects a vendor. The consumer then negotiates a purchase transaction including cost of product, delivery logistics, testing and acceptance. Finally consumer makes the payment and in future performs regular maintenance and makes warranty claims.

The advantage of traditional commerce is you get to see or try out the products physically before you buy, and there is no delivery time (get the product at the instant rather than waiting for 2-3 working days) as well.

1.2 Electronic Commerce

With the explosive growth of the Internet, E-commerce has become an important part of today's economy. Most major corporations and organizations now have public Web sites for E-commerce-related activities. Corporations want to maintain high availability, sufficient capacity, and satisfactory performance for their E-commerce Web systems, and want to provide satisfactory services to the users of their systems. With the advent of the Internet, the term e-commerce began to include:

- Electronic trading of physical goods and of intangibles such as information.
- All the steps involved in trade, such as on-line marketing, ordering payment and support for delivery.
- The electronic provision of services such as after sales support or on-line legal advice.
- Electronic support for collaboration between companies such as collaborative on-line design and engineering or virtual business consultancy teams.

Electronic Commerce, commonly known as (electronic marketing) e-commerce or eCommerce, consists of the buying and selling of products or services over electronic systems such as the Internet

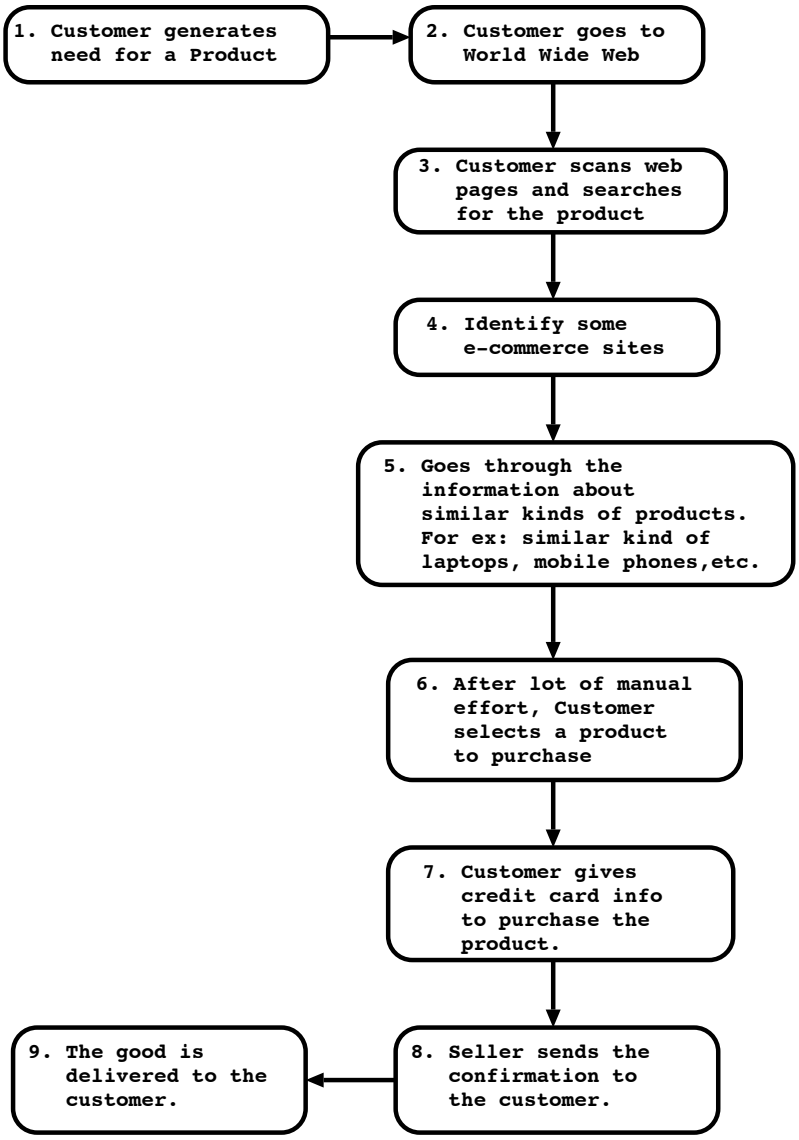

Figure 1.2: Electronic Commerce

and other computer networks. The amount of trade conducted electronically has grown extraordinarily with widespread Internet usage. Modern electronic commerce typically uses the World Wide Web at least at some point in the transaction's lifecycle.

E-commerce consists of following steps (refer Figure 1.2): The advent of e-commerce creates new need for the users. Customer researches about the product on-line using internet and selects the vendor through which he/she wants to buy the product. Then the payment is made on-line with the help of credit cards or other services. Finally the product is delivered at your door. For support and maintenance customer just needs to make a call or deliver an e-mail. Thus it can be observed that the customer can purchase or shop for any product sitting at home using few clicks.

E-commerce can be broken up into two main categories:

Online Purchasing: This business presents the customer with those technologies that make it easier for them to find data and buy commodities. The businesses encompassed in this category serve the customer by giving them the option of order placement, purchase order submission or requisition of quotes.

Online Shopping: This involves businesses giving information to customers so that they can make a decision and buy a certain commodity from you.

The introduction of electronic commerce has revolutionized the business processes for companies and made the shopping experience convenient for customers, but at the same time it has also opened several research issues.

Issues in E-commerce and Research Efforts

In this section we mention several issues that are being faced in electronic commerce and also present the respective research efforts that are being made to solve each of the issue.

(i) **Information Overload:** It refers to an excess amount of information being provided, making processing and absorbing tasks very difficult for the individual because sometimes we cannot see the validity behind the information. Customers are provided with lots of choices before selecting a product. For example, instead of choosing from tens of thousands of books in a superstore, consumers may choose among millions of books in an online store. As each product is described by several features they are faced with increased amount of information before they are able to select the product that meets their needs.

Research Efforts: Recommendation systems [32], personalized systems are being devel-

oped. A recommender system learns from a customer and recommends products that she will find most valuable from among the available products. Personalized systems adapt their behavior to the goals, tasks, interests, and other characteristics of their users. Based on models that capture important user characteristics, these personalized systems maintain their user's profiles and take them into account to customize the content generated or its presentation to the different individuals.

(ii) **Problem in Selecting products with more Variations:** The Web can provide aggregate information and interactive transmission, for example, to make the presentation more interesting. It is especially good at achieving remote accessibility while delivering rich information content at the same time. However, there are still barriers against consumers using the Web for retail shopping. Yes, selling products such as books, music CDs, and computer equipment over the Web has been relatively successful, but the type of text-based interface design used by, say, Amazon.com may not be able to cope with products that have more variations. To make the Web the prime place for shopping, more efforts will be needed to make the Web a better interface for consumers.

Research Efforts: One enhancement to the human computer interface (HCI) incorporates virtual reality (VR) with 3D visual and audio displays to enrich the Web shopping experience. Some of the techniques and tools developed in this area can be useful in implementing virtual storefronts.

(iii) **Non Automation of Electronic purchases:** The potential of the Internet for truly transforming commerce is largely unrealized to date. Electronic purchases remain mostly non-automated. While information about different products and vendors is easily accessible and orders and payments can be dealt with electronically, a human is still in the loop in all stages of the buying process. Traditional shopping activities require a large effort from a human buyer collecting and interpreting information on merchants, products and services, making an optimal purchase decisions and finally entering appropriate purchase and payment information.

Research Efforts: Comparative Web Search systems [34] are being developed to help the customers for easy shopping. The task of CWS is to seek relevant and comparative information from the Web to help users conduct comparisons among a set of topics. A system called CWS is developed to effectively facilitate Web users' comparison needs.

(iv) **Disorganization of Websites:** Commercial site often has a large number of pages (hundreds, thousands or more), which makes it very difficult for manual browsing without any

automated assistance. Moreover, different companies may organize the same information very differently. One company may use one page and another may use a few pages. Some even put different aspects of the same information in different categories. This makes it hard to obtain the complete information about a particular item. Even for a relatively small Web site, the number of Web pages could overwhelm the human user, not to mention that the number of pages in a typical site is still growing at an alarming rate.

Research Efforts: Authors [25] propose a number of methods to help the user find various types of unexpected information from his/her competitors' Web sites. Another approach proposes a novel visualization technique [26] to help the user find useful information from his/her competitors' Web site easily and quickly. It involves visualizing (with the help of a clustering system) the comparison of the user's Web site and the competitor's Web site to find similarities and differences between the sites. Web site mining [15] treats a web site as one large HTML-document and applies the well-known methods for page classification. The determined topics now represent the information the web site contains and can be used to classify it more accurately.

(v) **Lack of security, reliability, and privacy of personal data:** There is no real control of data that is collected over the Web or Internet. Data protection laws are not universal and so websites hosted in different countries may or may not have laws which protect privacy of personal data. There are numerous reports of websites and databases being hacked into, and security holes in software. For example, Microsoft has over the years issued many security notices and 'patches' for their software. Several banking and other business websites, including Barclays Bank, Powergen and even the Consumers' Association in the UK, have experienced breaches in security where 'a technical oversight' or 'a fault in its systems' led to confidential client information becoming available to all.

Research Efforts: Number of research efforts is being made towards the issue of security and privacy in e-commerce like Enhancing e-commerce security using GSM authentication [21].

1.3 Overview of the Proposed Approach

In the preceding section, we discussed about E-commerce and some of the issues that are being faced by E-commerce. In this thesis we have identified another issue in E-commerce and made an effort to provide the solution. We have made an effort to investigate the issue of product selection process and proposed an improved framework to help the customers in efficient product selection. In

addition we have extended the notion of special features to improve the process of resume selection and proposed an improved approach to help the Human Resource (HR) managers in selecting an appropriate candidate.

In the first section, we give an overview on the problem related to product selection process and in the second section we discuss the contribution related to the problem of resume selection.

1.3.1 Overview of the Proposed Approach for Improving the Product Selection Process

For a customer, it is difficult and time consuming process to select appropriate product in E-commerce environment. Increased choice has also increased the amount of information that customer must process before they are able to select the suitable product. Customers usually get lost in the vast space of product information and cannot find the products they really want.

We have made an effort to find a solution to the following issue of product selection in E-commerce environment. Several times, a customer faces the problem of selecting appropriate product out of several similar products. In such a situation, customer has to carefully go through the features of all the products to select the appropriate product. For example, a customer wants to buy a Nokia [1] [1] mobile phone. To select a mobile phone, the customer has to carefully go through the features of about 70 Nokia mobile phone models, where each mobile phone description consists of around 30 features. As the different variants of mobile phones are available, the customer has to manually browse several Web pages to identify the specialty of each mobile phone. Even after spending sufficient amount of time, sometimes, a customer may not be confident of his/her choice.

Due to the explosive growth of WWW and popularity of E-commerce, the issue of product selection is becoming more complicated. As every company is concentrating on personalization and mass customization [19], many variations of each product are being introduced in the market, with little deviation in their features. Even though, the customer wants to select only one product, several choices are provided to him/her. The customer has to carefully browse through large amount of information for similar products. As all the information looks similar, it generates confusion in the customers mind. As a result, it becomes difficult for the customer to choose the appropriate product from a set of similar products.

It can be observed that every product has some specialty and possesses corresponding special features. The basic idea is as follows: a customer is shown special feature of each product along with the common features. As a result, the customer can quickly browse through all the special

[1] Nokia is a public limited liability company that focuses on wireless and wired telecommunications, Internet and computer software.

features and make the appropriate selection. We propose a method to compute the specialness value of features. On the basis of specialness value, we propose a clustering algorithm to organize the features into two-level and three-level organization of features.

We have conducted experiments on three real world data-sets related to Nokia mobile phones, Sony[2] cameras and HP[3] laptops for the validation of the proposed framework. The results indicates that the proposed approach significantly reduces the number of features customers need to be browse for selecting an appropriate product from set of similar products.

1.3.2 Overview of the Proposed Approach for Resume Extraction System

With the advancement of Internet large number of resumes are received on-line, through e-mails or through services provided by companies (like info edge limited [4]) for searching resumes. For HR managers it has become a difficult and time consuming process to select appropriate resume from such large set of resumes.

The HR managers employ currently available tools and techniques to filter thousands of resumes to few hundred potential ones based on some keywords or criterion. The set of resumes hence obtained are similar to each other. In the current scenario, the HR managers then need to manually browse through each resume, to select appropriate resumes. We define this as Problem of Resume Selection from set of similar resumes.

We have used the notion of special features as proposed above to come up with a solution to resume selection problem. Resume selection process is a more complex problem than product selection process as (i) In product selection scenario features are standardized in the form of specification, whereas in case of resumes features are free-form text which is difficult to compare with each other and (ii) Resume has a hierarchical structure whereas products have single layered structure.

It can be observed that resumes share document-level hierarchical contextual structure where the related information units usually occur in the same textual block and text blocks of different information categories usually occur in relatively fixed order. Resume information is described as a hierarchical structure with several layers. The first layer consists of different sections such as

[2]Sony Corporation is the electronics business unit and the parent company of the Sony Group, which is engaged in business through its five operating segments electronics, games, entertainment (motion pictures and music), financial services and other

[3]HP specializes in developing and manufacturing computing, storage, and networking hardware, software and services. Major product lines include personal computing devices, enterprise servers, related storage devices, as well as a diverse range of printers and other imaging products

[4]Info Edge is a leading provider of online recruitment, matrimonial, real estate and educational classifieds and related services in India.

education, experience, skills and experience. Second layer consists of text describing each section that acts as features for respective sections separated by a delimiter.

The set of resumes obtained after the filtering from thousand of resumes are similar to each other. Since the resumes are similar, HR managers need to manually browse through the resumes to select the appropriate resumes. But there may exist special information in some resumes when compared to others. For example resume may contain specialty in education, specialty in experience, special skills or special achievements. There may be special information in one or more sections of a resume. Thus identifying such special information and organizing them efficiently could help in improving the performance of resume selection process.

Overall the resume selection is a complex process as resume contains free-text information and thus it is difficult to differentiate between two resumes. In this thesis we have proposed an approach to extract 'special skills' from a given set of similar resumes to ease the process of resume selection.

It can be observed that all the resumes have some common skills and each resume might contain some special skills when compared to others. The proposed approach helps the HR managers to identify the special skills (if present) in each resume and organizes the skills information in an efficient manner. The basic approach is as follows: first we divide the skills features into an hierarchical structure containing skill type and skill value features, then we compute the specialness of both type of features and then on the basis of specialness value we apply the clustering algorithm to organize the skill type and skill value features efficiently.

We experimented on the resumes of computer science domain and got encouraging results that could reduce time and effort put by the HR managers in selecting appropriate resumes from set of similar resumes.

1.4 Contributions

The major contributions are as follows:

- Formulated the problem of Product Selection from Similar Products.
- Proposed a notion of degree of specialness to identify special features.
- Proposed a clustering algorithm to organize the features using two-level and three-level feature organization methods.
- Formulated the problem of Resume Selection from Similar Resumes.
- Extended the notion of special features to solve the problem of resume selection.

- Evaluated the performance of the proposed approaches through the experimental results.

1.5 Organization of the Thesis

The thesis has been divided into 6 chapters. Chapter 2 explains the work related to problem of product selection and resume selection. Chapter 3 presents the problem of product selection from similar products and explains the proposed approach that identifies the special features and organizes the features of the product in an effective manner. It also presents the experimental results to validate the proposed framework. Chapter 4 presents how the notion of special feature can be extended to improve the process of Resume Selection and experimental results. Chapter 5 gives the summary of the work discussed in the thesis, conclusions and discusses about the future work. Chapter 6 is appendix which contains more results for the experiments in Chapter 4 and in the last we mention the Publications resulting from the work explained in the thesis.

Chapter 2

Related Work

In this chapter we review selected publications related to the topic covered in this thesis. First section outlines the work related to the problem of discovering special features to improve the process of product selection from similar products and second section tells the difference from the discussed approaches. Third and fourth section describes the additional work related to the problem of resume selection and difference from the discussed approaches respectively. Last section gives the summary of the chapter.

2.1 Literature Survey Related to Product Selection Scenario

The notion of special features is related to the research in the following areas:

- Unexpected Information
- Comparative Web Search Systems
- Web Site Mining
- Uniqueness Mining
- Outliers

We below discuss each related area.

2.1.1 Discovering Unexpected Information from Competitors Websites

The Web is increasingly becoming an important channel for conducting businesses, for disseminating information, and for communicating with people on a global scale [18]. This is not only true for

businesses, but also true for individuals. With all the information publicly available, it is natural that companies and individuals would like to find useful/interesting information from these Web pages.

The drawback of traditional approaches is that it is hard for the user to find unexpected information. They can only help the user find anticipated information because what the user specifies can only be derived from his/her existing knowledge space. In this work, authors argue that finding only what the user explicitly specifies is not sufficient. Those pieces of information that have not been specified by the user may also be of great interests. It is just that the user does not know about them, or has forgotten about them. Such information may be unexpected and can be of great importance in practice. For instance, it is important for a company to know what it does not know about its competitors e.g., unexpected services and products that its competitors offer. With this information, the company can learn from its competitors and/or design counter measures to improve its competitiveness. Such business intelligence information is increasingly becoming crucial to the survival and growth of any company. Existing web information extraction techniques cannot find such unexpected information, as it is unlikely (or impossible) for one to specify something that one has no idea of.

In the context of the Web, the unexpectedness of a piece of information is defined as follows:

Unexpectedness: A piece of information is unexpected if it is relevant but unknown to the user, or it contradicts the user's existing beliefs or expectations. Note that in this definition, the condition "it is relevant" is important. Not every piece of unknown information is interesting. A piece of information must first be relevant to the user. For example, if the user is a marketing executive, in his/her professional capacity he/she will not be interested in a piece of information on how to plant a tree, although the piece of information may well be unknown to him/her. However, a piece of information on a new marketing strategy will certainly be of interest as it is relevant and unknown to him/her.

The basic idea of the proposed approach is as follows: Given a user site U, a competitor site C, and some existing knowledge or expectations E about the competitor from the user, system (called WebCompare) first analyzes U to extract all its key information. It then analyzes C, and compares the information contained in C with that in U and E to find various types of unexpected information from C.

Types of Unexpected Information:

1. Finding the corresponding C page(s) of a U page: Here, the user is interested in finding some pages in C that are similar to a page in U.

2. Finding unexpected terms in a C page with respect to a U page: In many cases, the user wants to know unexpected terms given two similar pages, a C page and a U page.

3. Finding unexpected pages in C with respect to U: The aim of this method is to find the pages in C that are most unexpected with respect to the U site.

4. Finding unexpected concepts in a C page with respect to a U page: In many cases, keywords alone may not reveal some important information of a Web page. A combination of words or concept may be very informative.

This approach identifies different types of unexpected information as well as similar pages from competitors' set of Web pages when compared with respect to users set of Web pages. But it can be observed that the number of competitor pages could be large in number and thus the amount of unexpected and similar information generated could be very large making it difficult for the user to understand.

Secondly it does not identify the unexpected information w.r.t. each competitor page which could help the user in comparing with each of its competitors rather than with all at once. To summarize, it does not make an effort to exploit the special properties of each object within the whole set of pages including competitors and users set of pages.

2.1.2 Comparative Web Search Systems

Comparative Web systems deal with providing and seeking relevant and comparative information from a network or database, such as the Internet or Web [17].

The task of CWS is to seek relevant and comparative information from the Web to help users conduct comparisons among a set of topics. A system called CWS is developed to effectively facilitate Web users' comparison needs. Given a set of queries, which represent the topics that a user wants to compare, the system is characterized by: (1) automatic retrieval and ranking of Web pages by incorporating both their relevance to the queries and the comparative contents they contain; (2) automatic clustering of the comparative contents into semantically meaningful themes; (3) extraction of representative keyphrases to summarize the commonness and differences of the comparative contents in each theme .

There are several approaches available which can help people make comparisons on the Web. For example, some newly emerged Web sites began to provide comparison shopping services. Shopping.com and Froogle (http://froogle.google.com) have integrated product comparison services to provide comparative information such as price and customer reviews. However, most of these Web sites are specialized in a certain domain (e.g., products) and can only help fulfill limited comparison

tasks for a certain group of users. What's more, their services are based on the structured information provided by the database. Another method is to use traditional search engines for comparative search tasks. Unfortunately, this is not effective since Web users have to manipulate several search windows for a comparative view. To make comparisons with respect to different aspects, users have to frequently refine the queries appropriately or navigate through the result pages. This obviously is tedious for the users. Thus it is much desired to maintain a general platform on which users can easily retrieve and compare every kind of information they need.

In summary, the CWS system is characterized by:

1. Automatic retrieval and ranking of Web pages based on both their relevance to queries and the comparative contents they contain;
2. Automatic clustering the comparative contents into semantically meaningful themes;
3. Extraction of representative keyphrases to summarize the commonness and differences of the comparative contents in each theme.

Figure 2.1 shows the interface for CWS system. Two text boxes are provided to input the comparative queries. After queries are submitted, two lists of Web pages are generated by the system and are displayed in two columns. The left list of pages corresponds to the query contained in the left textbox, while the right list corresponds to the right query. For each result page, the information including title, URL, and snippet is displayed.

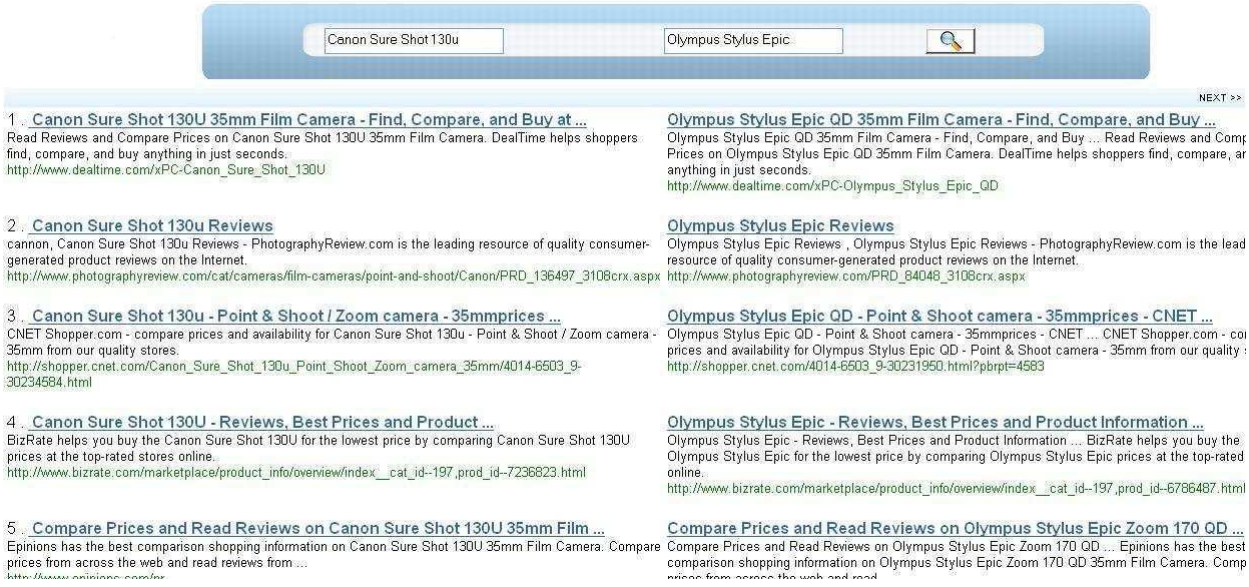

Figure 2.1: CWS Interface

CWS presents customers set of Web pages based on their ranking, clusters the web pages and identify representative keyphrases. This approach is based for comparing two products at a time. But as we have already observed that as a result of personalization and mass customization choices for the customers have increased significantly. Thus for this system to work efficiently customer should already had restricted his choice to two products, which is not the case most of the times. Thus the customer has to query again and again for comparing number of products.

The approach proposed in the thesis works beyond the CWS. It helps the user to identify special or additional information present in the pages or products that user wants to compare. It not only helps customers to compare two pages or products, but 'n' similar products at the same time.

2.1.3 Visualizing Web Site Comparisons

The Web is increasingly becoming an important channel for conducting businesses, disseminating information, and communicating with people on a global scale [29]. More and more companies, organizations, and individuals are publishing their information on the Web. With all this information publicly available, naturally companies and individuals want to find useful information from these Web pages. As an example, companies always want to know what their competitors are doing and what products and services they are offering. Knowing such information, the companies can learn from their competitors and/or design countermeasures to improve their own competitiveness. The ability to effectively find such business intelligence information is increasingly becoming crucial to the survival and growth of any company. Despite its importance, little work has been done in this area.

In this work, authors propose a novel visualization technique to help the user find useful information from his/her competitors Web site easily and quickly. It involves visualizing (with the help of a clustering system) the comparison of the users Web site and the competitors Web site to find similarities and differences between the sites. The visualization is such that with a single glance, the user is able to see the key similarities and differences of the two sites. He/she can then quickly focus on those interesting clusters and pages to browse the details.

The proposed technique and visualization are versatile and can be used to find many types of interesting pages and information.

Unexpected C pages (or clusters): These pages or clusters exist at the C site but not at the U site. As discussed previously, such pages are unexpected and often very interesting. In visualization, these pages or clusters are colored completely in red. In Figure 2.2, we highlighted cluster 1059, which has 13 C pages (5.39% of the total C pages) and it is a pure red cluster - all pages come from the competitors site. As related pages are clustered together, this gave the user a surprise because

it means that the competitor site used 5.39% of their Web pages to devote to something that the user site did not have at all.

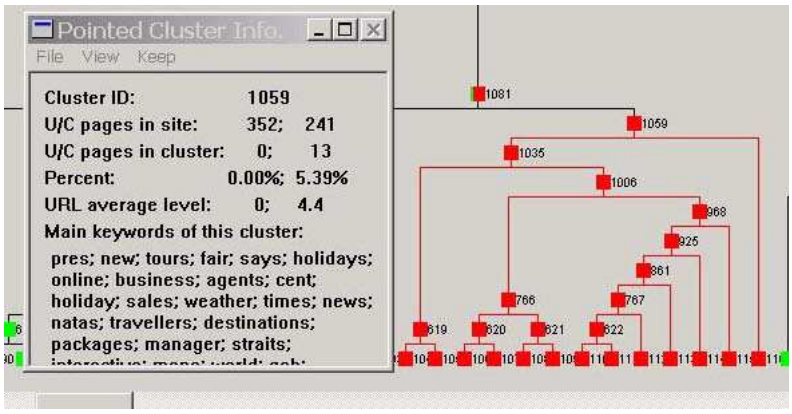

Figure 2.2: An example of unexpected C pages

Different emphases: Different companies often have different emphases of their businesses. These emphases reflect their current directions and/or strengths. In the system, this kind of information is easily noticed and discovered. One simply finds those clusters that have very unbalanced numbers of pages from the two sites. Different color proportions in each cluster on the screen reveal such information clearly. Figure 2.3 shows such an example in our travel domain.

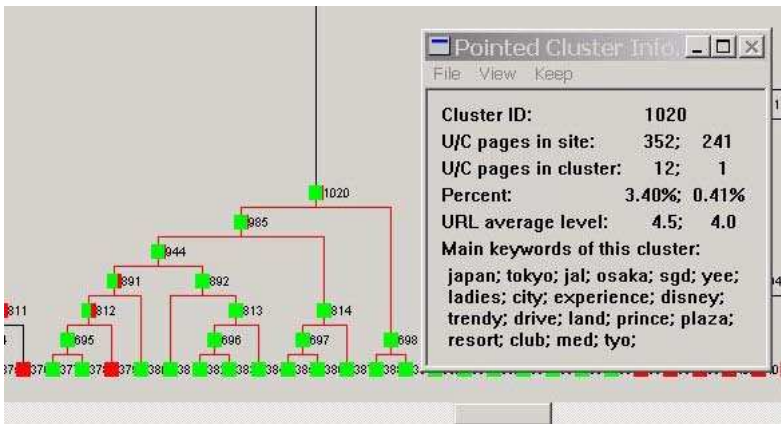

Figure 2.3: An example of unbalanced business emphasis

Expected pages: In most cases, the user knows something about their competitors. He/she may want to confirm his/her knowledge about the competitor. In the travel application, the user believed that their competitor had similar tour packages to Europe as theirs. He then used the find function in the system to find all the Web pages on Europe tours.

Finer differences: Here the authors show that one can drill down from a high level cluster to its lower level clusters to find interesting details. The function "keep" is very useful here as it allows information windows from multiple clusters to be displayed on the screen at the same time

In this approach given a set of web sites it presents the user with clusters of distinct as well as similar pages with respect to competitor and user web sites. Each cluster may contain large number of pages and user needs to manually browse each competitor page to identify the useful content w.r.t. its own set of pages. In this approach the user would again suffer from the problem of information overload.

2.1.4 Uniqueness Mining

It considers the problem of extracting the special properties of any given record in a dataset [28]. They are interested in determining what makes a given record unique or different from the majority of the records in a dataset. In the real world, records typically represent objects or people and it is often worthwhile to know what special properties are present in each object or person, so that the best use of them can be made.

Definition 1 (unique object): An object x in a dataset D is said to be unique with respect to a boolean property P that is computable for all objects in D and a threshold η in the range (0, 1) iff P(x) =True and $|y : y \in D \wedge P(y) = True| < \eta|D|$

When applied to relational datasets, D consists of the set of records in the dataset and P is a boolean property defined on the attributes of D. Without loss of generality, we treat attributes as being either categorical or numerical. As an example consider a student's database. Then P could be "Is the first language of this student = English??" or "Is the marks in Physics of this student in the range (60, 70)??"

Notice that if an object x $\in$ D is unique with respect to a property P , then all objects y $\in$ D for which P(y) = True must also be unique with respect to P. For convenience, we define the notion of uniqueness of a property as follows:

Definition 2 (Unique Property): A boolean property P that is computable for all objects in D is unique with respect to a threshold η in the range (0, 1) iff P(x) = True and $|y : y \in D \wedge P(y) = True| < \eta$ (D), where x is a record in the dataset D.

Definition 3 (Sibling Property): for any boolean property P applicable on objects in a dataset D, the set of all specializations of a generalization of P are said to be sibling properties of P.

By a generalization of property P, we mean any property obtained by relaxing one of the re-

quirements of P. By a specialization of property P, we mean any property obtained by adding one more requirement on P. Notice that any boolean property P is a sibling of itself.

Definition 4 (Trivially Unique Property): If according to definition 1, the total number of unique objects with respect to (D, P', η) over each sibling P' of some boolean property P,is large ($\geq \eta\ |D|$),then every unique P' is trivially unique.

Based on the above properties algorithm is proposed to find uniqueness of records. The user provides an input record x and desires to see all the ways in which x is unique. The algorithm then proceeds to first identify if x is unique with respect to singleton properties extracted during the preprocessing phase. If that fails, then combinations of singleton properties are considered.

Some of the examples scenarios where mining unique information can be useful are:

1. Deans of universities and principals of schools generally send letters to parents of students describing their childs performance. It may be easy to identify students who got the top rank in each subject. However,it will be nice to be able to identify a student who is the only one who got more than (or less than!) 70% in some subjects A, B and C. If some unique property of each student were mentioned, it would be interesting to the readers of their reports.

2. A student comes to a faculty member in a university asking for a project. The best outcome can be expected if a project is chosen that makes use of the special skills of this student. One way to identify these special skills would be to mine the database of student marks in various subjects, to find unique combinations of subjects in which this student performs well.

3. Special skills of people working in a company can be used to assign them to various projects.

4. Consider an exam paper having 10 questions. Rather than deciding the weightage of marks for each question,grades to students can be given depending on what unique combinations of questions they answered correctly. For example, if everyone answered questions 1,2 and 3 correctly, these questions wouldnt count much in the final grade. If some student was the only one to answer questions 5 and 6, then it indicates that this student deserves a good grade.

Uniquess mining deals with mining special properties of any given record in a dataset. They have worked on determining what makes a given record unique or different from the majority of the records in a dataset. The uniqueness mining approach has been applied mostly to the relational databases. This approach identifies only the special properties of an object and neglect the properties common between all objects. There can be some properties which could be common for some set of objects, which is not considered in this approach.

2.1.5 Outliers

There are several definitions existing for outliers.

Outliers are data objects with different characteristics compared to other data objects. Outliers are observations that deviate so much from other observations to arouse suspicion that they might have been generated using a different mechanism [14] or data objects that are inconsistent with the rest of the data objects [9].

Outlier detection refers to the problem of finding patterns in data that do not conform to expected behavior. These non-conforming patterns are often referred to as outliers, anomalies, discordant observations, exceptions, aberrations, surprise, peculiarities or contaminants in different application domain.

Outliers are generally categorized into 2 classes:

1. Web Outliers
2. Numerical Outliers

(i) Web Outliers

Outliers identified in web data are referred to as web outliers to distinguish them from traditional outliers. The mining process for web outliers is called web outlier mining [7].

The web consists of data from different sources made up of different data types. The different sources and data types pose a real challenge for automatic discovery of web-based information. For example, algorithms designed for mining outliers from web usage data cannot be applied to the web contents because of the differences in input data types. The taxonomy for web outliers shown in Figure 2.4 allows content-specific algorithms to be designed for mining outliers from specific sources. Thus, separate algorithms can be designed for mining outliers from web servers as well as from the web pages.

Web outlier mining is categorized into three components depending on the source and data types involved in the mining process as shown in Figure 2.4:

- Web usage outlier mining
- Web content outlier mining
- Web structure outlier mining

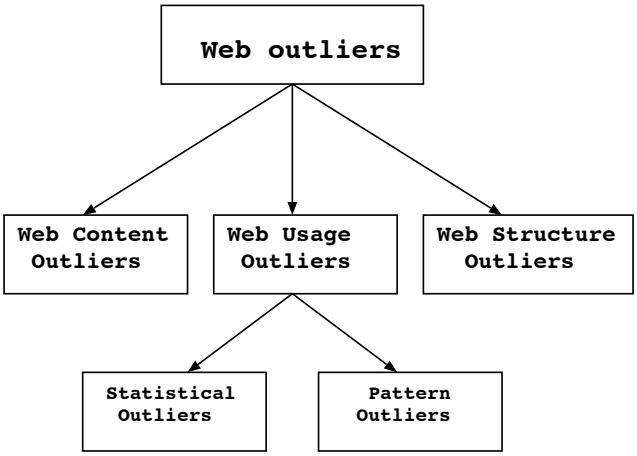

Figure 2.4: Taxonomy of Web Outliers

Web usage outlier mining

Web usage outlier mining is dedicated to finding outliers in web usage data. Web usage data often referred to as web clickstream data consist of user activities and interactions with the web. The usage data are often captured in servers as server logs, referrer logs, browser logs, etc. The contents of a typical web server may include IP addresses, time-in, pages addresses and references, and time spent on each page. Web usage data may contain two types of outliers: web usage statistical outliers and web usage pattern outliers each of which has its own unique importance.

Web usage statistical outliers

Access frequencies (the number of accesses per page) generated from web usage data has been the most commonly used summary statistics. Pages with high access frequencies are declared as very important and their contents usually moved to the main pages during review. But this is not always the case in real as some web designers may use 'catchy phrase' to have high number of hits. The second argument is that access frequencies do not provide information on the relationship between pages.

The problem is addressed by proposing a technique that uses access frequencies and time spent on a page. Thus, combining time and frequency is a better indicator of importance than using the frequency alone.

Web usage pattern outliers

Web usage pattern outliers are present in web usage trends. Consider the trends:
60% of users who visited the site www.chapters.ca purchased book(s).
20% of site visitors accessed Chapters' main page followed by the book page.
Web usage pattern outlier mining concentrates on finding patterns that deviate from normal ones.

In the sample trend above, Web usage pattern outliers would be the sites visited by the fraction of the remaining 40% of visitors who purchased book(s) outside Chapters.

Web content outlier mining

The web consists of interrelated web pages grouped into different categories depending on their contents. A web content outlier is described as page(s) with completely different contents from similar pages within the same category [5] [6]. For instance, Bombardier is an airplane manufacturer that also manufactures snowmobiles. The company web site has pages dedicated to promoting and marketing snowmobiles. Since airplane manufactures do not generally make snowmobiles the pages on snowmobiles constitutes web content outliers.

The motivation for web content outlier mining is enormous. For example, Superstore, Safeway and Co-Op are three grocery stores in Alberta. Superstore has clothing and hardware sections which the others do not. Supposing Superstore makes profit while Safeway and Co-Op do not, then it may be due to good sales from the clothing and hardware sections. However, if Safeway and Co- Op are profitable while Superstore is not, then it might be the case that the clotting and hardware sections are not profitable. This interesting and useful information can be obtained using web content outlier mining algorithms. It is assumed that all the products of the participating companies are available.

Web structure outlier mining

Web structure mining is the discovery of interesting patterns in the hyperlink structure of the web [5] [6]. The motive of web structure mining is to use the hyperlinks on the web to categorize documents into domains so that pages within the same category can be compared in terms of their similarities and differences.

In web structure outlier mining, we concentrate on discovering rare and interesting patterns from the hyperlink structure of the web. It is aimed at finding links whose connecting nodes contained information that are completely unrelated. Mining web content outliers may enhance the quality of hubs and authority sites for a given topic. Authority sites are sites (pages) having quality links to topics of interest whereas hubs are pages linked to by pages addressing the topic of interest.

(ii) Numerical Outliers

Detecting Outlying Properties of Exceptional Objects: This work focuses on defining a criterion to measure the abnormality of subsets of attribute values featured by a given object with respect to a reference data population, and on methods to evaluate it [8]. The measure proposed here does not require the definition of a distance relating pairs of objects but, rather, it is based

on a simple concept of relative frequency. Both global and local forms of abnormal properties are defined, where local properties are ones justified by some others playing the role of explanation for the former, and algorithms to mine the most relevant ones are described. This way, a fully automated support is provided to decode those properties determining the abnormality of the given object within the reference data context.

Thus, let a data population be given, stored into a database DB represented in the form of a relational table. Let o be an object of the population which we know is abnormal on the basis of available external knowledge. The objective here is that of devising techniques by which it is possible and effective to single out those combinations of attributes that justify the abnormality of the object with the highest plausibility. A property, or set of attributes, witnesses the abnormality of the object o if the combination of values o assumes on these attributes is very infrequent with respect to the overall distribution of the attribute values in the dataset: To this end, in the following, we introduce a measure by which we will be able to capture faithfully how much a set of attributes should be considered relevant so to explain the abnormality of the given individual.

Statistical techniques assume that the given dataset has a distribution model. Outliers are those points that satisfy a discordancy test, that is, they are significantly far from what would be their expected position, given the hypothesized distribution [9].

Density-based methods presented in [23], a new notion of local outlier is introduced that measures the degree of an object to be an outlier with respect to the density of its local neighborhood. This measure is called Local Outlier Factor (or LOF) and is assigned to each object.

Distance-based outlier detection has been introduced by [23] and [11] to overcome the limitations of statistical methods. A distance-based outlier is defined as follows: An object o in a dataset is an outlier with respect to parameters p and δ if at least a fraction p of the other objects in the database lies greater than distance δ from p. This definition generalizes the definition of outlier in statistics and is suitable in situations when the dataset does not fit any standard distribution.

In [24], the authors focus on the identification of the **intensional knowledge** associated with distance-based outliers. First, they detect the distance-based outliers P in the full attribute space and then, for each outlier P, they search for the subspaces that better explain why it is exceptional. In particular, this subspace coincides with the minimal subspace in which P is still an outlier. Note that the distance-based outlier measure is monotonic with respect to the subset inclusion relationship: If an object is an outlier in a subspace, then it will be an outlier in all its supersets. Although meaningful when dealing with homogeneous numerical attributes, this kind of monotonicity may

not be in general suitable for categorical attributes, where often objects exhibit outlierness only in characterizing subspaces.

In [4], anomalies are detected by searching for subspaces in which the data density is exceptionally lower than the mean density of the whole dataset. An abnormal lower projection is one in which the density of the data is exceptionally lower than the average.

In [42], the authors' goal is to find **example-based outliers** in a dataset. Given an input set of example outliers, that is, of objects known to be outliers, they search for those objects of the dataset which mostly exhibit the same exceptional characteristics as the example outliers.

In [41] the algorithm **HighDOD** is presented. The interest here is in finding the subspaces in which a given point is an outlier. The Outlying Degree (OD) of a point is measured using the sum of the distances between this point and its k-nearest neighbors [Angiulli and Pizzuti 2005]. OD has a monotonic property: If a point is an outlier in a subspace, then it will be an outlier in any superset of this subspace.

In [40], the authors present a variant of the task described in [41]. They focus on outlying **subspace detection**, that is, finding subsets of attributes in which a given data point significantly deviates from the rest of the population. Let p be an object, k be a positive integer, and s be a subspace. Then $D_k^s(\mathrm{p})$ denotes the distance from p to its kth-nearest neighbor in the dataset projected on the subspace s. The Subspace Outlying Factor (SOF) of a subspace s with respect to a given object p is defined as the ratio of $D_k^s(\mathrm{p})$ over the average D_k^s for points in the dataset. Outlying subspaces are detected by using a genetic algorithm.

In [36] the authors deal with **outlier detection for categorical data**. They present a novel definition of outlier based on a hypergraph model. By means of this modeling they aim at capturing the characteristics of data distribution in subspaces. In particular, they focus on detecting outliers by discovering sets of categorical attributes, called common attributes, being able to single out a portion of the dataset in which the value assumed by some objects on a single additional attribute, called exceptional attribute, becomes infrequent with respect to the mean frequency of the values in the domain of that attribute.

In [36] an algorithm is presented, called **HypergraphbasedOutlierTest (HOT)**, to mine hypergraph-based outliers. The HOT works by executing the following steps. First, all the frequent itemsets, having a user-specified support min sup, are mined by using the Apriori algorithm [Agrawal and Srikant 1994]. Then, these frequent itemsets are arranged in a hierarchy according to the contain-

ment relationship. Next, the hierarchy is visited using a bottom-up strategy. Frequent itemsets I represent common attributes, while each attribute A not in I represents a potential exceptional attribute. For each itemset I , the histogram of frequencies associated with each attribute A not in I is stored, and used to compute the deviation of each value taken by A in the database. Finally, the objects assuming a value on the attribute A whose deviation is smaller than a user provided threshold are returned as outliers.

The **Subgroup Discovery Task (SDT)** [22] aims at finding an interesting subgroup of objects with common characteristics with respect to a given attribute, called the target variable. Typically, a subgroup is identified by using a conjunction of selection expressions. For example, for the target variable coronary heart disease = true, an interesting subgroup might be one including the objects selected by the expression: smoke = true AND family history = true. The usefulness of a subgroup is typically obtained by means of a statistical test. The SDT outputs a group of exceptional objects, and it handles the target variable which is given as the input.

The outler algorithms are mostly categorized into two classes namely web outliers and numerical outliers. There are number of algorithms designed to identify numerical outliers but most of them have been applied to numerical data-sets.

The outlier algorithms discussed in data mining area deals with numerical data-sets and have not applied in case of Web documents. The outlier algorithms proposed in Web mining area concentrates more on classifying objects as an outlier, rather than mining specialness of each object.

2.2 Difference Over Existing Approaches

In this section we summarize how the notion of special features differs from the above discussed approaches.

Discovering unexpected information approach identifies different types of unexpected information as well as similar pages from competitors's set of Web pages when compared with respect to users set of Web pages.But it can be observed that the number of competitor pages could be large in number and thus the amount of unexpected and similar information generated could be very large making it difficult for the user to understand. Secondly it does not identify the unexpected information w.r.t each competitor page which could help the user in comparing with each of its competitors rather than with all at once. To summarize, it does not make an effort to exploit the special properties of each object within the whole set of pages including competitors and users set of pages.

CWS presents customers set of Web pages based on their ranking, clusters the web pages and identify representative keyphrases. This approach is based for comparing two products at a time. But as we have already observed that as a result of personalization and mass customization choices for the customers have increased significantly. Thus for this system to work efficiently customer should already had restricted his choice to two products, which is not the case most of the times.Thus the customer has to query again and again and has to for comparing number of products.

In Visualizing websites comparisons approach given a set of web sites it presents the user with clusters of distinct as well as similar pages with respect to competitor and user web sites. Each cluster may contain large number of pages and user needs to manually browse each competitor page to identify the useful content w.r.t its own set of pages. In this approach the user would again suffer from the problem of information overload.

Uniquess mining deals with mining special properties of any given record in a dataset. They have worked on determining what makes a given record unique or different from the majority of the records in a dataset. The uniqueness mining approach has been applied mostly to the relational databases. This approach identifies only the special properties of an object and neglect the properties common between all objects. There can be some properties which could be common for some set of objects, which is not considered in this approach.

The outlier algorithms discussed in data mining area deals with numerical data-sets and have not applied in case of Web documents. The outlier algorithms proposed in Web mining area concentrates more on classifying objects as an outlier, rather than mining specialness of each object.

2.3 Literature Survey Related to Resume Selection Scenario

The approaches related to the problem of Resume Selection are as follows:

- Learning Pinocchio, a toolkit of Information Extraction
- Resume Information Extraction with Cascaded Hybrid Model
- Other Information Extraction Systems Analogous to Resume IE

2.3.1 Learning Pinocchio, a toolkit of Information Extraction

One of the published results on resume Information Extraction (IE) is shown in [13]. In this work, they applied Learning Pinocchio $(LP)^2$, a toolkit of IE, to learn information extraction rules. for resumes written in English. The information defined in their task includes a flat structure of Name, Street, City, Province, Email, Telephone, Fax and Zip code.

Learning Pinocchio is an adaptive system for IE, based on a kind of transformation based like rule learning. Rules are learnt by generalising over a set of examples marked via XML tags in a training corpus. The system performs IE as tagging, i.e. the information is extracted by annotating texts using XML tags (e.g. the speaker in a seminar will be identified by surrounding it in the text with two tags: $< speaker >$ and $< /speaker >$)

IE is performed by applying a user-defined architecture. An architecture is always based on a preprocessor performing tokenization and optionally morphological analysis, part of speech tagging and gazet teer lookup.

Then the user can define a number of adaptive modules for IE. A module defines a scenario that summarizes the information to be extracted as a set of annotations (XML tags). A session of learning is associated with each module and a set of rules will be learnt for each module. Different modules in an architecture perform different steps in the IE process, for example we have developed modules for text zoning, named entity recognition and for more complex IE tasks. The user interface of Learning Pinocchio is web-based. All the interaction takes place via HTML forms. Each module (and therefore each architecture) can work in three modes: training mode, test mode and production mode.

The training mode is used to induce rules, to learn how to perform IE in a specific application scenario.The testing mode is used to test a module on an unseen tagged corpus, so to understand how well it performs on a specific application. Finally the system results are automatically compared with the user-provided results. The production mode is used when an application is released. The module receives the text as tagged by the previous modules in the architecture and adds XML tags to the corpus.

2.3.2 Resume Information Extraction with Cascaded Hybrid Model

Big enterprises and head-hunters receive hundreds of resumes from job applicants every day [39]. Automatically extracting structured information from resumes of different styles and formats is needed to support the automatic construction of database, searching and resume routing. The definition of resume information fields varies in different applications.

Normally, resume information is described as a hierarchical structure with two layers. The first layer is composed of consecutive general information blocks such as Personal Information, Education etc. Then within each general information block, detailed information pieces can be found, e.g., in Personal Information block, detailed information such as Name, Address, Email etc. can be further extracted. Based on the requirements of an ongoing recruitment management system which incorporates database construction with IE technologies and resume recommendation (routing), as shown in Table 2.1, 7 general information fields are defined. Then, for Personal Information, 14 detailed information fields are designed; for Education, 4 detailed information fields are designed. The IE task, as exemplified in Table 2.1, includes segmenting a resume into consecutive blocks labelled with general information types, and further extracting the detailed information such as Name and Address from certain blocks. Extracting information from resumes with high precision and recall is not an easy task. given the hierarchy of resume information, a cascaded two-pass IE framework is designed.

Table 2.1: Predefined Information Types

Info Hierarchy	**Info Type (Label)**	
General Info Personal	Information(G1); Education(G2); Research Experience(G3); Award(G4); Activity(G5); Interests(G6); Skill(G7)	
Detailed Info	Personal Detailed Info (Personal Information)	Name(P1); Gender(P2); Birthday(P3); Address(P4); Zip Code(P5); Phone(P6); Mobile(P7); Email(P8); Registered Residence(P9); Marriage(P10); Residence(P11); Graduation School(P12); Degree(P13); Major(P14) ;
	Educational Detailed Info (Education)	Graduation School(D1); Degree(D2); Major(D3); Department(D4)

In the first pass, the general information is extracted by segmenting the entire resume into consecutive blocks and each block is annotated with a label indicating its category. In the second pass, detailed information pieces are further extracted within the boundary of certain blocks. Moreover, for different types of information, the most appropriate extraction method is selected through experiments. For the first pass, since there exists a strong sequence among blocks, a HMM model is applied to segment a resume and each block is labelled with a category of general information. They also apply HMM for the educational detailed information extraction for the same reason. In addition, classification based method is selected for the personal detailed information extraction where information items appear relatively independently.

2.3.3 Other Information Extraction Systems Analogous to Resume IE

There are some applications that are analogous to resume IE, such as seminar announcement IE [30], job posting IE [33] [16] and address segmentation [10] [27] Most of the approaches employed in these applications view a text as flat and extract information from all the texts directly [30] [27] [31] [16] Only a few approaches extract information hierarchically. In [33] authors present a double classification approach to perform IE by extracting words from pre-extracted sentences. In [10] authors develop a nested model, where the outer HMM captures the sequencing relationship among elements and the inner HMMs learn the finer structure within each element. But these approaches employ the same IE methods for all the information types. Compared with them, our model applies different methods in different subtasks subtasks to fit the special contextual structure of information in each sub-task well.

The work done on resumes focuses on information extraction of resume and building a classifier to extract the information from resume and storing it in structured manner. None of the above approach tries to extract special information from given set of resumes

2.4 Difference Over Existing Approaches

There has been various resume IE systems built to extract structured information from resumes. These techniques are based on tokenization, pattern matching, Hidden Markov Models etc. There has been very little work on resume mining problem. The work done on resumes mostly focuses on information extraction of resume or building a classifier to extract the information from resume and storing it in structured manner. None of the above approach tries to extract special information from given set of resumes

In the approach proposed in this thesis we have made an effort to mine special skills from given set of resumes and also proposed an clustering approach for efficient organization of skills features that could help HR managers in efficient selection of appropriate candidates.

2.5 Summary of the Chapter

In this chapter firstly we discussed the literature related to the notion of special features. There are several systems that mine different types of information to help the users overcome the problem of information overload. Then we mention briefly how the notion of special features differ from each of the discussed approach.

Secondly we discussed the literature related to the problem of resime selection. There has been little work on resume mining problem. Most of the related work concentrates on extraction of structured information from the resumes, whereas in the approach proposed in this thesis we have made an effort to mine special information from structured resumes.

Chapter 3

Discovering Special Product Features to Improve the Process of Product Selection in E-commerce Environment

In the E-commerce environment, a customer faces several difficulties for selecting the product. There are large variants of each product available in the market with little deviation in their features. Even though, the customer wants to select only one product, several choices are provided to him/her. The customer has to carefully browse through large amount of information for similar products. Customers usually get lost in the vast space of product information and cannot find the products they really want. We have investigated the issue of selecting the appropriate product from the group of similar products and proposed an improved framework for the efficient product selection.

In this chapter, we explain the problem of product selection and explain the motivation behind the problem of product selection. In the next section we describe the proposed approach that includes the notion of degree of specialness and clustering algorithm to organize the features into two-level and three-level. Finally we discuss about the experimental results that consists of evaluation metrics, description of data-sets, results on the data-set and performance measure of the proposed approach. We also discuss the literature survey related to the problem of product selection.

3.1 Problem of Product Selection

In the E-commerce environment, a customer faces several difficulties for selecting the product. Several times, a customer faces the problem of selecting appropriate product out of several similar

Table 3.1: Sample Product Features of N-77 and N-82 mobile phone separated by delimiter (,)

N-77 Features	**N-82 Features**
network umts gsm 900 gsm 1800 gsm 1900,	network gsm 850 900 1800 1900 hsdpa,
announced 2007 1q (february), weight 114 g	announced 2007 4q (november),weight 114 g,
display type tft 16m colors,	display type tft 16m colors,
display size 240 x 320 pixels 2.4 inches,	display size 240 x 320 pixels 2.4 inches,
ringtones type polyphonic (64 channels) mp3,	ringtones type polyphonic monophonic true tones mp3,
vibration yes, phonebook yes,	vibration yes, phonebook practically unlimited-
call records yes, card slot microsd (transflash) hotswap,	entries and fields photocall,
operating system symbian os 9.2 s60 rel 3.1,	call records detailed max 30 days,
camera 2 mp 1600x1200 pixels video(cif)-	camera 5 mp 2592 x 1944 pixels carl zeiss-
flash secondary cif video call camera,	optics autofocus video(vga 30fps),
gprs/data speed class 11,	operating system symbian os 9.2 s60 rel 3.1,
messaging sms mms email instant messaging,	card slot microsd hot swap 2 gb card included,
infrared port no	gprs/data speed class 32 107 kbps,
games yes java downloadable,colors black	messaging sms mms email instant messaging,
3g 384 kbps, bluetooth v1.2 with a2dp	infrared port no, games yes downloadable,
dvb-h tv broadcast receiver, video calling	wlan wi-fi 802.11 b/g upnp technology,
push to talk, java midp 2.0,mp3/m4a/aac/eaac+/wma player	bluetooth v2.0 with a2dp, t9,
t9,stereo fm radio, voice command/dial	usb v2.0 microusb,browser wap 2.0/xhtml html,
pim including calendar to-do list and printing	built-in gps receiver, motion sensor (with ui auto-rotate),
document viewer, photo/video editor	java midp 2.0, mp3/aac/aac+/eaac+/wma player

products. In such a situation, customer has to carefully go through the features of all the products to select the appropriate product. For example, a customer wants to buy a Nokia mobile phone. (Table 3.1 shows a sample features for N-77 and N-82 models of Nokia mobile phone [2] separated by delimiter comma (,).) To select a mobile phone, the customer has to carefully go through the features of about 70 Nokia mobile phone models, where each mobile phone description consists of around 30 features. As the different variants of mobile phones are available, the customer has to manually browse several Web pages to identify the specialty of each mobile phone. Even after spending sufficient amount of time, sometimes, a customer may not be confident of his/her choice.

In our problem we assume that each product is described by a text document which contains several features separated by a delimiter. Each feature consists of word or set of words. Table 3.1 shows Nokia N-73 and N-77 mobile phones and their respective features separated by comma.

Before explaining the problem statement, we define the term 'similar products'.
Two products are said to be similar if the number of common features between them is greater than a given threshold. For example, different variants of mobile phones form a group of similar products as they have common features.

The problem of product selection out of similar products is defined as follows:

Let 'P' be a set of 'n' similar products and 'τ' be the total number of time units spent by the customer to make a purchase decision after browsing the details of 'n' products. The problem of product selection is to develop an approach that minimizes 'τ'.

Considering that a customer on an average takes 't' time units to go through the information related to each product to make a decision on whether to buy a product or not. If there are 'n' similar types of products, the customer will spend total n * t time units to browse through all the products and to make an appropriate choice. Thus, the estimated value in this case 'τ' is equal to n * t time units. Our objective is to propose an approach that could help the customers in making the decision in less than 'τ' time units.

3.2 Motivation for the Product Selection Problem

In this section we view the effect of Mass Customization (MC) and Personalization which has lead to the production of large number of alike products i.e. products with very little deviation in their features. For example: mobile phones, laptops, ipods etc. At the same time it has also provided the customers with number of choices in the product selection process.

Mass Customization
In recent scenario companies have came up with a notion of mass customization and personalization. Mass Customization is the customization and personalization of products and services for individual customers at a mass production price. Mass customization relates to the ability to provide individually designed products and services to every customer through high process flexibility and integration. Mass customization has been identified as a competitive strategy by an increasing number of companies.

"Mass Customization" is the new frontier in business competition for both manufacturing and service industries. At its core is a tremendous increase in variety and customization without a corresponding increase in costs. At its limit, it is the mass production of individually customized goods and services. At its best, it provides strategic advantage and economic value.

The concept of mass customization is attributed to Stan Davis in Future Perfect [37] and was defined by [35] as "producing goods and services to meet individual customer's needs with near mass production efficiency". In [20] authors concurred, calling it a strategy that creates value by some form of company-customer interaction at the fabrication and assembly stage of the operations level to create customized products with production cost and monetary price similar to those of mass-produced products.

The development of MC systems is based on three main ideas: (i) New flexible manufacturing and information technologies enable production systems to deliver higher variety at lower cost. (ii) There is an increasing demand for product variety and customization. (iii) The shortening of product life cycles and expanding industrial competition has led to the breakdown of many mass

industries, increasing the need for production strategies focused on individual customers.

MC has the ability to provide individually designed products and services to every customer through high process agility, flexibility and integration. MC systems may thus reach customers as in the mass market economy but treat them individually as in the pre-industrial economies.

MC has become an important manufacturing strategy. Agility and quick responsiveness to changes have become mandatory to most companies in view of current levels of market globalization, rapid technological innovations, and intense competition. MC broadly encompasses the ability to provide individually designed products and services to customers in the mass-market economy.

Personalization

Personalization is the process of tailoring content to individual users' characteristics or preferences. It is a means of meeting the customer's needs more effectively and efficiently, making interactions faster and easier and, consequently, increasing customer satisfaction and retention.

There is much current interest in the personalization of products be they software or hardware. Thus mobile phones are now sold with replaceable colored covers; e-commerce sites learn a users' preferences and word processors allow you to customize the menus and tool bars. Personalization is defined here as a process that changes the functionality, interface, information content, or distinctiveness of a system to increase its personal relevance to an individual.

In an age of information explosion, personalization offers a way to focus on customers/users. Personalization doesn't waste time showing you what everyone else sees and then asking you to search or display/hide things.

Thus we can see that as a result of mass customization and personalization companies are offering similar kinds of products to fulfill the need for each individual. There is very little deviation in the features for similar kind of products and each product possess very few special features. It not only helps them to meet the needs of every individual but also at the same time it helps them to decrease their production costs and increase their profits as well as the customer base.

3.3 Proposed Approach

The problem is to reduce the time taken by a customer to select a product from a set of similar products. It can be observed that even though the products are similar to each other on several aspects, each product possesses some specialty which is exhibited through its special features. Due to its special features, each product can be distinguished from the other similar products. The intuition here is that if the special features of each product are identified and shown to the

customer, the time taken by the customer to make a decision for selecting a product would be reduced in comparison to showing all the information about the products. Note that the number of special features is much smaller when compared to the total information about the product. So the customer can quickly browse through the special features of all the products to take the decision as compared to all the features for every product.

The main issue here is how to measure the specialness of features of all products and organize the features according to their specialness value in an effective manner. To solve the issue, we propose an approach to compute specialness of features. Then, on the basis of specialness value of features, we propose two different approaches to organize the features.

3.3.1 Degree of Specialness

Degree of specialness: Let P be the set of 'n' similar products, where product $p_i \in$ P and each product p_i possesses set of features f(p_i). Let F be multiset containing all features such that F = $\cup_{i=1}^{n} f(p_i)$. Each feature in F is denoted by f_j where $0 \leq$ j $\leq |F|$ and n(f_j) denote the number of products to which feature f_j belongs. Note that, the multiset F contains duplicate features. We consider these features as distinct features as they belong to distinct products.

Let f_j be a feature, such that $f_j \in$ f(p_i). The degree of specialness (DS) of a feature f_j is its capability of making the product p_i, separate/distinct/unique/special from other products. The DS value for a feature varies between zero to one (both inclusive). The DS value of the feature f_j is denoted by DS(f_j). Then,

$$DS(f_j) = \begin{cases} 1 & \text{if } n(f_j) = 1 \\ 1 - (n(f_j)/|P|) & \text{otherwise} \end{cases} \tag{3.1}$$

Based on the DS values of features, features can be classified as common features, common cluster features and special features. Features for which the DS value is '0' are called common features. Feature for which the DS value is closer to 1 are called special features. The other features are called as common cluster features.

3.3.2 Feature Organization Approaches

Feature Organization Approaches

After assigning the DS values to all the features in the set F, the next issue is to organize the features in an effective manner by taking into account the corresponding specialness value. We

propose two approaches to organize the features. Based on the specialness value of features, the products along with the features can be organized in 'L' levels, where (L≥1). We have proposed approaches to organize the features into two-level and three-level. Note that, we can consider the existing method, where a customer has to browse through all the features of each product, as one-level approach. To improve the performance of one-level approach, we propose two-level and three-level approaches. We plot a graph between feature rank versus DS value, where feature rank denotes the rank of feature based on its specialness value. Based on the property of feature rank versus DS value curve, one can follow either two-level or three-level to organize the features. The overall framework is given in section 3.3.3

1. Two-level approach

Sort the features based on DS values. Analyze the graph of feature rank versus the corresponding DS value. As feature rank increases, if there is a sharp decrease observed in DS values, two-level approach should be followed.

For two-level approach, features are divided into two groups: common features (I-level) and special features (II-level). If DS value of a feature is 0, it is a common feature otherwise, it is a special feature. Note that, for any product p_i, its complete set of features f(p_i) equals to the common features shown at the I-level and special features of product p_i shown at the II-level.

Common Features of all the products **(I-level)**	
Product	Special Features **(II-level)**
P1	special features of P1
P2	special features of P2
P3	special features of P3
P4	special features of P4

Figure 3.1: Two-level feature organization

Figure 3.1 depicts the organization of features using the two-level approach. The I-level entry in the table contains the features common to all the set of products and the II-level entry shows the special features possessed by each product.

Algorithm 1 in Table 3.2 describes the steps of two-level approach. In the algorithm, we check the DS value of the feature. For any feature, if the DS value is 0, it is considered as an I-level feature. Otherwise, it is considered as an II-level feature.

Table 3.2: Algorithm: Two-level algorithm

Input. P: a set of 'n' products; T: Threshold, F: Set of features for all products in P **Output:** I-level / II-level
1. Notation used f_i: a feature in F 2. **for** i=1 to $\|F\|$ 3. if DS(f_i) = 0 ; 4. f_i as a I-level feature 5. **else**, f_i is a II-level feature

2. Three-level approach

Sort the features based on DS values. Analyze the graph of feature rank versus the corresponding DS value. As feature rank increases, if there is a gradual decrease observed in DS values, three-level approach should be followed.

In three-level approach, the features are distributed into three levels: I-level, II-level and the III-level. Figure 3.2 depicts the organization of features using the three-level approach. I-level contains the common features, II-level contains common cluster features and III-level contains special features. It can be noted that, for any product p_i, its complete set of features f(p_i) is a combination of (i) the common features at I-level (ii) common cluster features for the cluster in which p_i is a member and (iii) special features of product p_i at III-level.

Common Features of all the products **(I-level)**		
(II-level) Common Cluster Features	Product	**(III-level)** Special Features
Common features for P1 and P2	P1	special features of P1
	P2	special features of P2
Common features for P3 and P4	P3	special features of P3
	P4	special features of P4

Figure 3.2: Three-level feature organization

Description of Three-level algorithm

We discuss procedure to organize the features using three-level approach. Three-level algorithm takes set of products P, similarity threshold (ST) and feature set F as input and returns common features, common cluster features and special features for each product with formation of clus-

Table 3.3: Algorithm: Three-level algorithm

Input: n: is a number of products; P: set of 'n' products;
F: set of features for all 'n' products; ST: similarity threshold.
Output: I-level / II-level / III-level features

1. **Formation of Clusters**
2. Notations used
2.1. nc: number of clusters; i, j: integers;
2.2. CL(i): the i'th cluster where (i <= n);
2.3. CF(i): set of features of i'th cluster;
2.4. f(p_i): set of features for product p_i.
3. nc = 0; for i=1 to n {CL(i) = ϕ and CF(i) = ϕ}
4. Select the first product p_1 ;
4.1. CL(1) = CL(1) $\bigcup$ p_1 ;
4.2. CF(1) = CF(1) $\bigcup$ f(p_1);
4.3. nc = nc+1
5. **for** each product $p_i \in$ P - {p_1}
6. **for** each cluster CL(j) ($1 \leq$ j $\leq$ nc)
7 **if** sim(p_i,CL(j)) = max_j(sim(p_i,CL(j))) $\geq$ ST, **then**
8. {CL(j) = CL(j) $\bigcup$ p_i ;
9. CF(j) = CF(j) $\bigcap$ f(p_i) }
10. **else** { nc = nc+1;
11. CL(nc) = CL(nc) $\bigcup$ p_i;
12. CF(nc) = CF(nc) $\bigcup$ f(p_i) } .
13. end
14. end
15. **Calculate I-level Features.**
16. Select all f_k from F such that DS(f_k) = 0.
17. **Calculate II-level Features.**
18. **For** each Cluster CL(j) ($1 \leq$ j $\leq$ nc),
19. Corresponding CF(j) contains II-level features.
20. **Calculate III-level Features**
21. **For** each $p_i \in P$
22. Calculate the cluster number of p_i
23. Special Feature of p_i = f(p_i) - CF(clno) - Common features.

ters as an intermediate step. Each cluster CL(i) is associated with feature set CF(i) where CF(i) represents the features that are common among all the products present in the cluster CL(i).

Algorithm 2 in Table 3.3 shows the steps of three-level approach. It contains two major parts. In the first part, products are distributed into the clusters. In the second part, each cluster of products is organized using three-level organization as shown in Figure 3.2. The summary of the first part is as follows. Initialize the first cluster with the first product. Then the following steps are repeated for each product: (i) For each other product p_j, if the similarity of p_j with the existing cluster or clusters is greater than similarity threshold, the product p_j is put into into the cluster with maximum similarity; Otherwise, new cluster is initialized with p_j.

In the second part, the features of each cluster are organized into three-levels. I-level contains the features of all clusters with DS value as '0'. The II-level contains the common features of each cluster. The III-level contains the remaining special features of each product. The similarity between the products p_i and CL(i) is denoted by sim(p_i, CL(i)) and is calculated as follows:

$$sim(p_i, CL(i)) = |f(p_i) \bigcap CF(i)|$$

Organizing the features using three-level method is an iterative process. The value of similarity threshold should be chosen such that the products are clustered into a reasonable number of clusters and the number of features shown to user can be reduced. For example, ST could be chosen as fifty percent of the average number of features in a product eliminating the common features. Then the threshold can be gradually increased, and the number of clusters formed and total number of features shown to user can be observed. If the number of features to be shown decreases significantly, we can increase the threshold and check the same. It can be observed that if the threshold is decreased, the number of common features for each cluster would decrease and consequently number of clusters shown to user would be increased. The objective of clustering the products is to reduce the effort of users by providing them with more convenient view and also less number of features. In case of large number of clusters, it leads to more confusion. Finally, we can set the ST threshold to particular value which gives minimum number of clusters and minimum number of features to be shown to user.

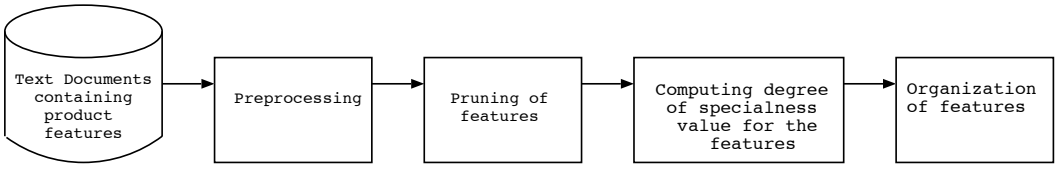

Figure 3.3: Flow diagram of the overall framework

3.3.3 Overall framework

The input to the proposed approach is the text documents where each text document contains the features of one product separated by a delimiter (Figure 3.3). The steps of the proposed framework are discussed below.

1. **Preprocessing** Preprocessing involves processing of data in order to prepare it for the further analysis. There are several steps required to prepare data before applying the proposed algorithm.

2. **Pruning of features**: There are some features which are repetitive or subset of another feature. Such features are removed. For example, the feature '65 w ac power adapter' is a part of the feature 'cq45-106au 65 w ac power adapter'.

3. **Computing DS value**: The DS values for all the features are computed. The feature rank versus DS value curve can be approximated to a linear curve and the corresponding slope can be calculated. If the slope of the graph is greater than a threshold value, two-level approach can be followed. Otherwise, three-level approach can be followed.

4. **Organization of features**: We organize the features using either two-level or three-level organization methods.

3.4 Experimental Results

To evaluate the performance, we have applied the proposed framework on three real world data-sets and reported the results. Specification for different models of mobile phones, cameras and laptops are extracted from the web sites. The details of the data-sets are as follows:

- Mobile phones Data-set [2]: It contains the details of 16 Nokia mobile phone models: N-70, N-72, N-73, N-75, N-77, N-80, N-81, N-82, N-85, N-90, N-91, N-92, N-93, N-95, N-96, N-97. Total number of features comes to 382.

- Camera Data-set [3]: It contains the details of 7 Sony camera models: DSLR-A350K, DSLR-A350, DSLR-A350X, DSLR-A700H, DSLR-A700K, DSLR-A700. Total number of features comes to 600.

- Laptop Data-set [1]: It contains the details of 10 HP laptop models: CQ50Z, HDX, dv2700t, dv2800t, dv5t, dv7z, dv6700t, dv9700t, tx2000z and tx2500z. Total number of features comes to 320.

3.4.1 Evaluation Metric

We define the performance metric called 'reduction factor' (rf) to measure the performance improvement. The rf denotes the reduction in the number of features that the customer needs to browse to select a product from set of 'n' similar products as compared to one-level approaches. More the reduction factor more advantageous it is for the customers as they need to browse less number of features.

Let 'F' denote the total number of features for all the products, F(i) denote the number of features in 'i'-level and 'L' denotes the number of levels. The 'rf' is defined as,

$$rf = 1 - \frac{\sum_{i=1}^{L} F(i)}{F}$$

F can be verified for II-level and III-level as follows:
For II-level:

$$F = F(1) * n + F(2) + PrunedFeatures.$$

where 1^{st} term denotes Common Features and 2^{nd} term denotes total Special Features.
For III-level, let n(f_j) denotes the number of features in cluster 'j', n(p_j) denotes the number of products in cluster 'j' and n_c denotes the total number of clusters formed.

$$F = F(1) * n + \sum_{j=1}^{n_c} n(f_j) * n(p_j) + F(3) + PrunedFeatures.$$

where 1^{st} term denotes Common Features, 2^{nd} term denotes Common Cluster Features and 3^{rd} term denotes Special Features.

3.4.2 Preprocessing

The input to the preprocessing step is the set of text documents where each text document contains the features of one product separated by a delimiter. The output contains the preprocessed text documents. Following steps are performed for the preprocessing of text documents:

- Sentences are identified from the document by considering of comma ',' (',' is the delimiter).
- Convert the entire input text to lower case.
- The string length of keyword must be greater than 2 or it must be alphanumeric[a-zA-Z0-9].
- We have used functions to remove text which does not create visible output.

- Trimmed special characters like # $ % ^ & _ { } ~.
- Removed stop words.

3.4.3 Pruning

Pruning helps us to eliminate duplicate features in a product. There are cases where a feature is a subset of another feature. Thus such features can be eliminated. The Table 3.4 shows the examples of pruned features. Feature1 shows the pruned feature and Feature2 show the parent feature of the pruned one.

3.4.4 Experimental Results

Analysis of Feature Rank Versus Degree of Specialness Graph

We named the graph plotted in Figure 3.4 as Frequency graph. The frequency graph is plotted between the feature rank and their respective degree of specialness value. In general we can observe two types of variation in the graph. In the first case the value of degree of specialness decreases gradually. In such cases there are some steps present in the graph for some specialness values. These steps indicate the number of features with that particular specialness value. Thus we can infer that in such case there is a possibility of getting cluster of features that are common to some set of products. Therefore, in such a scenario it is advantageous to go with three-level approach for the organization of features. In Figure 3.4 graph for Camera data-set and Laptop data-set shows the described scenario.

In second case the value of degree of specialness decreases sharply. In such cases there are no steps in the graph, which infers that there are very few features common to some set of products. Thus in such cases it is not possible to form clusters of products based on the common features between them. Therefore, it is advantageous to follow two-level approach for the organization of features. In Figure 3.4 graph for Mobile phones data-set shows the described scenario.

Results on Data-set

We discuss the results for each data set.

Mobile Phones data-set
Feature Statistics: Table 3.5 and Table 3.6 shows the feature statistics for Two-level and Three-level feature organization for Mobile Phones data-set. Feature statistics table indicate the

Table 3.4: Examples of Pruned Features

Laptop Data-set
Feature1 :: processor Feature2 :: amd sempron(tm) processor for notebook pcs si-40 (2.0ghz)
Feature1 :: microphone Feature2 :: microphone + webcam
Feature1 :: hard drive Feature2 :: 120gb 5400rpm sata hard drive
Feature1 :: adapter Feature2 :: 65w adapter
Feature1 :: memory Feature2 :: 2gb ddr2 system memory (2 dimm)
Feature1 :: 6 cell lithium ion battery Feature2 :: high capacity 6 cell lithium ion battery
Feature1 :: support Feature2 :: supermulti 8x dvd+/-r/rw with double layer support
Feature1 :: hp imprint finish (mesh) + microphone + webcam Feature2 :: hp imprint finish (mesh) + microphone + webcam for hp brightview infinity
Feature1 :: 802.11b/g wlan Feature2 :: 802.11b/g wlan and bluetooth
Feature1 :: intel(r) wifi link 5100agn Feature2 :: intel(r) wifi link 5100agn and bluetooth(tm)
Mobile Data-set
Feature1 :: colors Feature2 :: display t ype tft 256k colors
Feature1 :: colors Feature2 :: display type tft touchscreen 16m colors
Camera Data-set
Feature1 :: image data Feature2 :: supplied software image data converter sr ver.2.0
Feature1 :: software Feature2 :: supplied software image data converter sr ver.2.0
Feature1 :: professional Feature2 :: xp home and professional vista;
Feature1 :: np-fm500h rechargeable battery Feature2 :: battery type np-fm500h lithium-ion rechargeable battery
Feature1 :: sync Feature2 :: slow sync
Feature1 :: sensor Feature2 :: 1 cross sensor
Feature1 :: spot Feature2 :: adjustable spot af: selectable

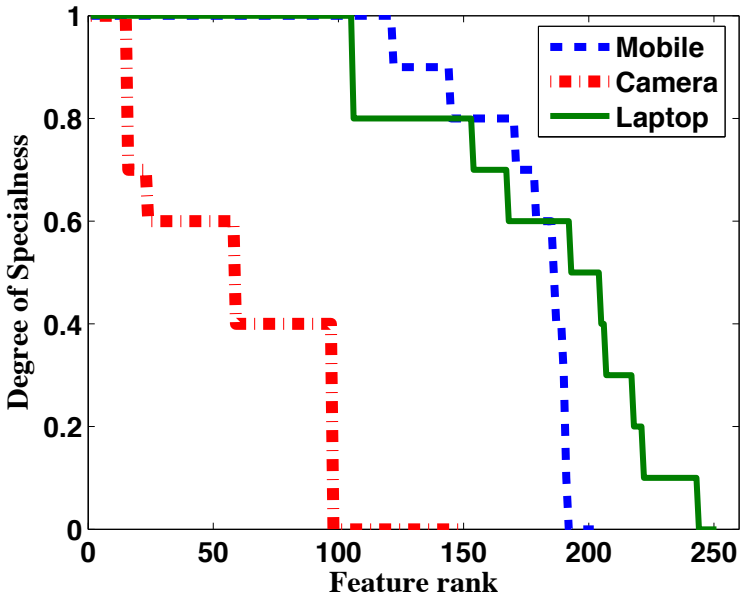

Figure 3.4: Frequency graph

statistics of data-set obtained after applying the proposed approach. It shows the number of common features, total number of clusters, number of features and number of products for each cluster, number of special features, total number of features present in the data-set and total features that are being displayed using the three-level organization approach. These feature statistics are used to calculate the performance of the proposed approach.

It can be seen that in this case using Three-level approach yields large number of clusters which could be difficult for user to view and also there is very little difference in reduction factor between Three-level and Two-level organization of features. Thus in such cases it is more efficient to use Two-level feature organization approach.

It can also be verified using feature rank graph (refer Figure 3.4) between feature rank versus DS values. For the curve related to mobile phone data set, it can be observed that the value of DS decreases sharply. So, in this case, it is efficient to use two-level approach.

Feature Organization: Table 3.7 and Table 3.8 shows the organization of features based on two-level approach and three-level approach respectively. For Two-level feature organization I-level shows the common features and II-level shows the special features and for Three-level feature organization I-level shows common features for all the products, II-level shows the common cluster features for each cluster and III-level shows the special feature of each product.

Performance: The'rf' value is calculated for the two-level and three-level approach (refer Table

3.13). The value of rf comes to 55% for Two-level and 60% for Three-level feature organization. It indicates that there is an atleast 55% reduction in the effort put-up by the user to select the product.

Table 3.5: Mobile Phones Data-set Statistics (II-level)

Number of Common Features = 13
Special Features = 156
Pruned Features = 18
Total Features displayed = 13 (Common Features)
+ 156 (Special Features) = 169
F = (13 * 16) + 156 + 18 = 382

Table 3.6: Mobile Phones Data-set Statistics (III-level)

Number of Common Features = 13
Number of Clusters = 11
Features in Cluster 1 = 0, Products in Cluster 1 = 1
Features in Cluster 2 = 0, Products in Cluster 2 = 1
Features in Cluster 3 = 6, Products in Cluster 3 = 2
Features in Cluster 4 = 4, Products in Cluster 4 = 2
Features in Cluster 5 = 0, Products in Cluster 5 = 0
Features in Cluster 6 = 0, Products in Cluster 6 = 0
Features in Cluster 7 = 5, Products in Cluster 7 = 3
Features in Cluster 8 = 0, Products in Cluster 8 = 0
Features in Cluster 9 = 0, Products in Cluster 9 = 0
Features in Cluster 10 = 0, Products in Cluster 10 = 0
Features in Cluster 11 = 0, Products in Cluster 11 = 0
Features in Cluster 12 = 0, Products in Cluster 12 = 0
Features in Cluster 13 = 0, Products in Cluster 13 = 0
Total Common Cluster Features = 15
Special Features = 121
Pruned Features = 18
Total Features displayed = 13 (Common Features) +
15 (Common Cluster Features) + 121 (Special Features) = 149
F = (13 * 16) + (2 * 6 + 2 * 4 + 3 * 5) + 121 + 18 = 382

Table 3.7: Organization of features using two-level approach for Mobile Phones Data-set

Common Features (I-level)	
network umts G's 900 gsm 1800 gsm 1900, display type tft 256k colors ringtones type polyphonic (64 channels) monophonic true tones mp3, mp3/aac/mpeg4 player, call records yes, messaging sms mms email instant messaging, games yes, java downloadable push to talk, integrated handsfree, pim including calendar to-do list and printing built-in handsfree, voice command/memo	
Product	**SpecialFeatures (II-level)**
N-71	dimensions 98.6 x 51.2 x 25.8 mm 103 cc, operating system s60 3rd edition (symbian os series 60 ui),weight 139 g, card slot minisd (up to 2gb), 128 mb card included 10 mb shared memory,
N-72	dimensions 109 x 53 x 18 mm 103 cc, weight 124 g stand-by up to 260 h, talk time up to 3 h 40 min
N-73	dimensions 110 x 49 x 19 mm, weight 116 g, vibration 3d sound stereo speakers, video(cif), flash secondary vga video call camera, gprs/data speed class 10 (4+1/3+2 slots) 32 48 kbps, stand-by up to 350 h, 240 x 320 pixels, 2.4 inches, 36 x 48 mm, announced 2006 2q (april) camera 3.2 mp 2048x1536 pixels carl zeiss optics autofocus, edge class 10 236.8 kbps, operating system symbian os 9.1 s60 3rd edition, battery type standard battery li-ion 1100 mah(bp-6m)
N-75	dimensions 95 x 52 x 20 mm 93 cc, weight 123.5 g, colors chocolate black, second external 256k colors display (160 x 128 pixels), downloadable themes, operating system symbian os 9.1 s60 rel 3.0u, battery type standard battery li-ion 800mah (bl-5bt), Talk time Up to 4 h, display size 240 x 320 pixels 2.4 inches, 2 MP, 1600x1200 pixels, LED flash card slot microsd (transflash) hotswap, gprs/data speed class 11
N-77	dimensions 111 x 50 x 18.8 mm 92 cc, bluetooth v1.2 with a2dp, stand-by up to 170 h, talk time up to 4 h 30 min, video(cif) flash secondary cif video call camera, battery type standard battery li-ion 1100 mah (bp-6m) Talk time Up to 4 h, display size 240 x 320 pixels 2.4 inches, 2 MP, 1600x1200 pixels, LED flash card slot microsd (transflash) hotswap, gprs/data speed class 11
N-80	dimensions 95 x 50 x 23 mm, card slot rs-dv-mmc (up to 2gb), hot swap 40 mb internal memory, weight 133 g, camera 3 mp 2048x1536 pixels video(cif) flash
N-81	dimensions 102 x 50 x 17.9 mm 86 cc, operating system symbian os 9.2 series 60 v3.1 ui, stand-by up to 410 h, weight 140 g, camera 2 mp 1600x1200 pixels video(vga) flash secondary, cif videocall camera, rotating gallery with navi wheel, colors cobalt blue graphite grey, wlan wi-fi 802.11b/g with upnp
N-82	dimensions 112 x 50.2 x 17.3 mm 90 cc, operating system symbian os 9.2 s60, rel 3.1,xenon flash secondary cif videocall camera, talk time up to 4 h 20 min, 5 MP, 2592x1944 pixels Carl Zeiss optics autofocus LED flash, phonebook practically unlimited entries and fields photocall, wlan wi-fi 802.11 b/g upnp technology, edge class 32 296 kbps dtm class 11 177 kbps, a-gps function
N-83	dimensions 112 x 50.2 x 17.3 mm 90 cc, card slot microsd hot swap, 2 gb card included, stand-by up to 225 h, talk time up to 4 h 20 min, xenon flash secondary cif videocall camera, motion sensor (with ui auto-rotate)
N-85	dimensions 103 x 50 x 16 mm 76 cc, weight 128 g, display type oled 16m colors, stand-by up to 363 h, card slot microsd (transflash) up to 8gb, accelerometer sensor for auto-rotate touch-sensitive navi wheel, operating system symbian os 9.3 s60 rel 3.2, hscsd 43.2 kbps, talk time up to 6 h 50 min, fm transmitter, 5 MP, 2592x1944 pixels Carl Zeiss optics autofocus LED flash phonebook practically unlimited entries and fields photocall, wlan wi-fi 802.11 b/g upnp technology, edge class 32 296 kbps dtm class 11 177 kbps, a-gps function

N-90	dimensions 112 x 51 x 24 mm, weight 173 g, camera 2 mp 1600x1200 pixels carl zeiss optics autofocus video (cif) flash, stand-by up to 280 h, card slot rs-dv-mmc 64 mb card included, hotswap 31 mb shared memory
N-91	dimensions 113.1 x 55.2 x 22 mm, weight 160 g, card slot no 4gb microdrive for music mms ringtones images video, operating system symbian os v9.1 series 60 ui 3rd edition, battery type standard battery li-ion (bl-5c) 900 mah
N-92	dimensions 107.4 x 58.2 x 24.8 mm 136cc, weight 191 g, card slot rs-dv-mmc (up to 2gb), hot swap 90 mb internal memory, stand-by up to 336 h, gprs/data speed class 11 384 kbps (wcdma) 236.8 kbps (edge)
N-93	dimensions 118 x 55.5 x 28.2 mm 133 cc, weight 180 g, second 65k colors, twist and rotating screen, card slot minisd, hot swap 128 mb card included, video(vga 30 fps), flash secondary cif video call camera, tv out support, 240 x 320 pixels, 2.4 inches, 36 x 48 mm, announced 2006 2q (april) camera 3.2 mp 2048x1536 pixels carl zeiss optics autofocus edge class 10 236.8 kbps, operating system symbian os 9.1 s60 3rd edition, battery type standard battery li-ion 1100 mah(bp-6m)
N-95	dimensions 99 x 53 x 21 mm 90 cc, weight 120 g, card slot microsd (up to 2gb), hot swap 128 mb card included, operating system symbian os 9.2 s60 rel 3.0, edge class 32 296 kbps dtm class 11 236.8 kbps, built-in gps
N-96	dimensions 103 x 55 x 18 mm 92 cc, weight 125 g, display size 240 x 320 pixels 2.8 inches, 16 gb internal memory, 128mb ram 256mb system memory, 5 MP, 2592x1944 pixels Carl Zeiss optics autofocus LED flash phonebook practically unlimited entries and fields photocall, wlan wi-fi 802.11 b/g upnp technology, edge class 32 296 kbps dtm class 11 177 kbps, a-gps function
N-97	dimensions 117.2 x 55.3 x 15.9-18.3 mm, weight 150 g, display type tft touchscreen 16m colors, nokia maps 2.0 touch, display size 360 x 640 pixels 3.5 inches, proximity sensor for auto turn-off, accelerometer sensor for auto-rotate, full qwerty keyboard handwriting recognition, card slot microsd (transflash) up to 16gb, operating system symbian os v9.4 series 60 rel 5, camera 5 mp 2584x1938 pixels carl zeiss optics autofocus video(vga@30fps), talk time up to 6 h 40 min, messaging sms mms email push email im, browser wap 2.0/xhtml html rss feeds, a-gps support, flash lite 3, stand-by up to 430 h digital compass, mp3/wma/wav/eaac player, mpeg4/wmv/3gp video player

Table 3.8: Organization of features using three-level approach for Mobile Phones Data-set

Common Features (I-level)		
network umts G's 900 gsm 1800 gsm 1900, display type tft 256k colors ringtones type polyphonic (64 channels) monophonic true tones mp3, mp3/aac/mpeg4 player, call records yes, messaging sms mms email instant messaging, games yes, java downloadable push to talk, integrated handsfree, pim including calendar to-do list and printing built-in handsfree, voice command/memo		
Common cluster features (II-level)	**Product**	**Special Features (III-level)**
-	N-71	dimensions 98.6 x 51.2 x 25.8 mm 103 cc, operating system s60 3rd edition (symbian os series 60 ui),weight 139 g, card slot minisd (up to 2gb), 128 mb card included 10 mb shared memory,
-	N-72	dimensions 109 x 53 x 18 mm 103 cc, weight 124 g stand-by up to 260 h, talk time up to 3 h 40 min
240 x 320 pixels, 2.4 inches, 36 x 48 mm, announced 2006 2q (april) camera 3.2 mp 2048x1536 pixels carl zeiss optics autofocus, edge class 10 236.8 kbps, operating system symbian os 9.1 s60 3rd edition, battery type standard battery li-ion 1100 mah(bp-6m)	N-73	dimensions 110 x 49 x 19 mm, weight 116 g, vibration 3d sound stereo speakers, video(cif), flash secondary vga video call camera, prs/data speed class 10 (4+1/3+2 slots) 32 48 kbps, stand-by up to 350 h,
	N-93	dimensions 118 x 55.5 x 28.2 mm 133 cc, weight 180 g, second 65k colors, twist and rotating screen, card slot minisd, hot swap 128 mb card included, video(vga 30 fps), flash secondary cif video call camera, tv out support
display size 240 x 320 pixels 2.4 inches, 2 MP 1600x1200 pixels, LED flash card slot microsd (transflash) hotswap, gprs/data speed class 11	N-75	dimensions 95 x 52 x 20 mm 93 cc, weight 123.5 g, colors chocolate black, second external 256k colors display (160 x 128 pixels), downloadable themes, operating system symbian os 9.1 s60 rel 3.0u, battery type standard battery li-ion 800mah (bl-5bt), Talk time Up to 4 h,
	N-77	dimensions 111 x 50 x 18.8 mm 92 cc, bluetooth v1.2 with a2dp, stand-by up to 170 h, talk time up to 4 h 30 min, video(cif) flash secondary cif video call camera, battery type standard battery li-ion 1100 mah (bp-6m) Talk time Up to 4 h,
-	N-80	dimensions 95 x 50 x 23 mm, card slot rs-dv-mmc (up to 2gb), hot swap 40 mb internal memory, weight 133 g, camera 3 mp 2048x1536 pixels video(cif) flash
-	N-81	dimensions 102 x 50 x 17.9 mm 86 cc, operating system symbian os 9.2 series 60 v3.1 ui, stand-by up to 410 h, weight 140 g, camera 2 mp 1600x1200 pixels video(vga) flash secondary, cif videocall camera, rotating gallery with navi wheel, colors cobalt blue graphite grey, wlan wi-fi 802.11b/g with upnp
5 MP, 2592x1944 pixels Carl Zeiss optics autofocus LED flash, phonebook practically unlimited entries and fields photocall, wlan wi-fi 802.11 b/g upnp technology, edge class 32 296 kbps dtm class 11 177 kbps, a-gps function	N-82	dimensions 112 x 50.2 x 17.3 mm 90 cc, operating system symbian os 9.2 s60, rel 3.1,xenon flash secondary cif videocall camera, talk time up to 4 h 20 min,
	N-85	dimensions 103 x 50 x 16 mm 76 cc, weight 128 g, display type oled 16m colors, stand-by up to 363 h, card slot microsd (transflash) up to 8gb, accelerometer sensor for auto-rotate touch-sensitive navi wheel, operating system symbian os 9.3 s60 rel 3.2, hscsd 43.2 kbps, talk time up to 6 h 50 min, fm transmitter,
	N-96	dimensions 103 x 55 x 18 mm 92 cc, weight 125 g, display size 240 x 320 pixels 2.8 inches, 16 gb internal memory, 128mb ram 256mb system memory,
-	N-83	dimensions 112 x 50.2 x 17.3 mm 90 cc, card slot microsd hot swap, 2 gb card included, stand-by up to 225 h, talk time up to 4 h 20 min, xenon flash secondary cif videocall camera, motion sensor (with ui auto-rotate)
-	N-90	dimensions 112 x 51 x 24 mm, weight 173 g, camera 2 mp 1600x1200 pixels carl zeiss optics autofocus video (cif) flash, stand-by up to 280 h, card slot rs-dv-mmc 64 mb card included, hotswap 31 mb shared memory
-	N-91	dimensions 113.1 x 55.2 x 22 mm, weight 160 g, card slot no 4gb microdrive for music mms ringtones images video, operating system symbian os v9.1 series 60 ui 3rd edition, battery type standard battery li-ion (bl-5c) 900 mah
-	N-92	dimensions 107.4 x 58.2 x 24.8 mm 136cc, weight 191 g, card slot rs-dv-mmc (up to 2gb), hot swap 90 mb internal memory, stand-by up to 336 h, gprs/data speed class 11 384 kbps (wcdma) 236.8 kbps (edge)
-	N-95	dimensions 99 x 53 x 21 mm 90 cc, weight 120 g, card slot microsd (up to 2gb), hot swap 128 mb card included, operating system symbian os 9.2 s60 rel 3.0, edge class 32 296 kbps dtm class 11 236.8 kbps, built-in gps
-	N-97	dimensions 117.2 x 55.3 x 15.9-18.3 mm, weight 150 g, display type tft touchscreen 16m colors, nokia maps 2.0 touch, display size 360 x 640 pixels 3.5 inches, proximity sensor for auto turn-off, accelerometer sensor for auto-rotate, full qwerty keyboard handwriting recognition, card slot microsd (transflash) up to 16gb, operating system symbian os v9.4 series 60 rel 5, camera 5 mp 2584x1938 pixels carl zeiss optics autofocus video(vga@30fps), talk time up to 6 h 40 min, messaging sms mms email push email im, browser wap 2.0/xhtml html rss feeds, a-gps support, flash lite 3, stand-by up to 430 h digital compass, mp3/wma/wav/eaac player, mpeg4/wmv/3gp video player

Camera data-set

Feature Statistics Table 3.9 shows the feature statistics for the Camera data-set. It can be observed that using Three-level organization is more efficient in this case as the number of resulting clusters are very less that could be easily viewed by the customers. Two-level organization in this case will result in displaying large number of features.

It can also be verified by the feature rank graph (refer Figure 3.4) between feature rank versus degree of specialness. For the curve related to camera data-set, it can be observed that the value of DS decreases gradually. So, the features are organized using three-level approach

Feature Organization: Table 3.10 shows the organization of features using three-level approach. The I-level contains the common features for all the products, II-level contains the common cluster features, and the III-level shows the special features of each product.

Performance: The rf value comes to 81% for III-level organization and 53% for II-level organization of features (refer Table 3.13). So, using III-level approach gives 81% reduction in the effort for the product selection.

Table 3.9: Camera Data-set Statistics (III-level)

Number of Common Features = 45
Number of Clusters = 2
Features in Cluster 1 = 29; Products in Cluster 1 = 3
Features in Cluster 2 = 33; Products in Cluster 2 = 4
Total Common Cluster Features = 62
Special Features = 15
Pruned Features = 50
Total Features displayed = 45 (Common Features) +
62 (Common Cluster Features) + 15 (Special Features) = 122
F = (45 * 7) + (29 * 3 + 33 * 4) + 15 + 50 = 600

Table 3.10: Organization of features using three-level approach for Camera Data-set

Common Features (I-level)		
shoulder strap with eyepiece cap and remote commander clip, battery type np-fm500h lithium-ion rechargeable battery supplied software image data converter sr ver.2.0, self timer yes (10 seconds 2 seconds off), multi-pattern measuring 40 segment limited warranty term 1 year parts labor, red-eye reduction on/off (all modes) focus auto focus ttl phase detection, os must be installed at the factory, lens type interchangeable a-mount, service and warranty information, aperture/shutter priority manual,weights and measurements, operating system compatibility, software/usb driver cd-rom, supports usb 2.0 hi-speed, histogram display yes rgb, ver.2.1.02 (windows only), af illuminator light yes, xp home and professional, viewfinder optical ttl, picture motion browser, battery capacity 7.2v, infinity 95% coverage, accessories supplied, primary color filter, convenience features, operating conditions,ev compensation ev,center weighted, optics/lens, output(s) video yes, ntsc/pal selectable, lightbox sr ver.1.0, auto focus mode yes, inputs and outputs, movie mode(s) n/a, program shift yes, clear raw nr n/a, usb port(s) yes, lens and media, microphone n/a, w/ev indicator, on/off select, playback only, command dial, fill-flash, rear flash, processor, body cap, hardware, general power, bulb		
Common cluster features (II-level)	**Product**	**Special Features (III-level)**
dimensions (approx.) (whd), (130.8 x 98.5 x 74.7 mm) weight (approx.) 1 lb 4.5 oz (582g) body not including battery	DSLR-A350.txt	-
note no memory stick media nor adaptors are included burst mode max 2.5fps with viewfinder 2fps in live view mode 1600 mah cipa standard approx 730 pictures,card is full,	DSLR-A350K.txt	200000 professionalme, sal-1870 dt 18-70mm f3.5-5.6 standard zoom lens
shooting capacity jpeg , (l size fine) until memory raw jpeg 3 frames, raw 4 frames megapixel 14.2mp live view yes,1/3 ev steps, multi-point 9 area, 8 line, 1 cross sensor,os x (v 10.3 or later), shutter speed 1/4000 to 30 sec , wireless off camera flash, bc-vm10 battery charger	DSLR-A350X.txt	sal-1870 dt 18-70mm f3.5 zoom lens (27 105 35mm eq , 55-200mm f4-5.6 telephoto zoom lens
dimensions (approx.) (whd) , (141.7 x 104.8 x 79.7 mm), 0.3 ev 0.5 ev steps selectable , multi-point 11 area, 5 center twin-cross lines, flash mode(s) manual pop-up auto	DSLR-A700.txt	color mode(s) standard vivid- neutral adobe rgb clear deep light portrait landscape sunset night view . autumn b/w sepia
weight (approx.) 1 lb 8 oz , (690g) body not including battery burst mode selectable, raw 17, hi (5fps) lo (3fps),jpeg extra fine 8, adjustable spot af selectable	DSLR-A700H.txt	craw (compressed) 24 c raw+jpeg 12 dt 18-200mm f/3.5-6.3 high magnification zoom lens
jpeg standard/fine unlimited to capacity of media, lcd 3.0 tft xtra fine (921k pixels) lcd,	DSLR-A700K.txt	dro modes include off standard auto advanced manual dt 18-70mm f3.5 zoom lens (27 105 35mm eq)(sal-1870)
wireless off camera flash (with flash hvl-f56am f36am) visual focus confirm direct via spherical acute matte screen wireless remote commander (rmt-dslr1), video cable ,usb cable	SAL-16105.txt	memory stick pro media compatibility tested- to support up to 16gb media capacity dt 16-105mm f3.5 zoom lens (24mm to 157.5mm eq)

Laptop data-set

Feature Statistics: Table 3.11 shows the feature statistics for the Camera data-set. It is the similar case as of Camera data-set.

Figure 3.4 shows the graph between feature rank versus degree of specialness. For the curve related to laptop, it can be seen that the value of DS decreases less gradually as compared to graph related to mobile. So, in this case, organizing the features using three level approach is more appropriate.

Feature Organization: Table 3.12 shows the organization of features using three-level approach.

Performance: The rf value for Three-level comes to 71% and for III-level it comes out to be 43% (refer Table 3.13). Thus using III-level feature organization gives more efficient organization of features.

Table 3.11: Laptop Data-set Statistics (III-level)

Number of Common Features = 9
Number of Clusters = 5
Features in Cluster 1 = 0, Products in Cluster 1 = 1
Features in Cluster 2 = 0, Products in Cluster 2 = 1
Features in Cluster 3 = 21, Products in Cluster 3 = 4
Features in Cluster 4 = 15, Products in Cluster 4 = 2
Features in Cluster 5 = 14, Products in Cluster 5 = 2
Total Common Cluster Features = 50
Special Features = 35
Pruned Features = 43
Total Features displayed = 9 (Common Features) +
50 (Common Cluster Features) + 35 (Special Features) = 94
F = (9 * 10) + (21 * 4 + 15 * 2 + 14 * 2) + 35 + 43 = 320

Table 3.12: Organization of features using three-level approach for Laptop Data-set

Common Features (I-level)		
primary cd/dvd drive, total memory slots 2 dimm, pci expansion, expansion port 3 connector adobe reader 8.x, one year of hardware parts and labor coverage, 24 x 7 support one year of award-winning, e-mail response in as little time as an hour		
Common cluster features (II-level)	**Product**	**Special Features (III-level)**
-	CQ50Z	amd athlon(tm) x2 dual-core processor for notebook pcs ql-60, front-side bus up to 4400 mt/s system bus running, 30-days free software support, trial internet service microsoft office home, microsoft windows media player 11
-	HDX	intel(r) core(tm) 2 duo processor t9300, hp games powered by wild tangent, integrated stereo microphone, hdmi external port, 640gb 5400rpm sata dual hard drive, premium cd/dvd burner software, front-side bus (processor dependent) up to 800 mhz, expansion slots expresscard/54 slot
genuine windows vista home premium, with service pack 1 (32-bit) microsoft windows media player 11, tv out (s-video), intel(r) core(tm) 2 duo , processor t9300 (2.50ghz) 3gb ddr2 system memory (2 dimm), tv entertainment experience finance accounting software, $25 off quicken deluxe 2008, ieee 1394 firewire, 5-in-1 digital media card reader intel(r) core(tm) 2 duo processor t5750 (2.0ghz) verizon wireless v740 expresscard (service activation required) premium photography software, slingbox flash tour, norton internet security(tm), 2008 3 year subscription, os and recovery media, roxio backup mypc(tm)	dv2700t	free upgrade to 160gb 5400rpm sata hard drive intel(r) pro/wireless 4965agn network w/bluetooth
	dv2800t	128mb nvidia geforce 8400m gs 14.1 diagonal wxga high-definition display
	dv6700t	pentium(r) dual-core mobile processor t2390 rj-45 (lan) vga
	dv9700t	17.0 diagonal wsxga high-definition display intel(r) core(tm) 2 duo processor t9500
genuine windows vista home premium with service pack 1 (32-bit), no tv tuner w/remote control, 3gb ddr2 system memory (2 dimm),mobile stereo earbud headphones back-up media management,maximum memory expansion 4gb $25 off quicken deluxe 2008, premium photography software, verizon wireless v740 expresscard (service activation required) 802.11b/g wlan and bluetooth, no high speed 56k modem port roxio photosuite 9 deluxe, hdmi 1.3 connector adobe premiere elements 4,maximum memory expansion 8g microsoft(r) office professional, arcsoft media converter 2.5 internet service microsoft office	dv5t	 blu-ray rom with supermulti dvd+/-r/rw double layer
	dv7z	amd turion(tm) x2 ultra dual-core mobile processor zm-80 cyberlink dvd suite premium
genuine windows vista home premium with service pack 1, roxio photosuite 9 deluxe finance accounting software, back-up media management built-in digitizer pen, $30 off quickbooks pro 2008 maximum memory expansion 4gb, 5-in-1 digital media card reader, verizon wireless v740 expresscard (service activation required) free upgrade to computrace lojack for notebooks $20 off norton 360(tm 15 months), adobe(r) photoshop(r) elements 6.0 2 integrated consumer irs (remote control receiver) dimensions 8.82 (l) x 12.05 (w) x 1.23 (min h)/1.52 (max h)	tx2000z	2 integrated consumer irs
	tx2500z	amd turion(tm) x2 ultra dual-core mobile processor zm-82 ati radeon(tm) hd 3200 graphics

Summary of results

It can be observed (refer Table 3.13) that there is a large reduction in the effort made by customers to select the product from the set of similar products using the proposed two-level and three-level approaches. The proposed solution also facilitates easy comparison of information about different products. Overall, the results show that the proposed approach has a potential to improve the performance of E-commerce applications.

Table 3.13: Comparison between one-level, two-level and three-level for different data-sets

Dataset	**Organization Type**	$\|F\|$	$\sum_{i=1}^{L} F(i)$	**rf**
Mobile Dataset	One-level	382	382	0
	Two-level	382	169	0.55
	Three-level	382	149	0.60
Camera Dataset	One-level	600	600	0
	Two-level	600	279	0.53
	Three-level	600	122	0.81
Laptop Dataset	One-level	320	320	0
	Two-level	320	183	0.43
	Three-level	320	94	0.71

3.5 Summary of the Chapter

Selecting appropriate product from a group of similar products is one of the problems faced by the customer in E-commerce environment. In this chapter, we have investigated the issue and proposed an improved approach. We have introduced the notion of degree of specialness to identify special features of the products and proposed a clustering algorithm to organize the special features in an efficient manner.

We have conducted the experiments on three real world data-sets related to Nokia-mobile phones, Sony-cameras and HP-laptops. The results indicate that the proposed approach has a potential to improve the performance of the product selection.

Chapter 4

An Approach to Extract Special Skills to Improve the Performance of Resume Selection

With the advancement of Internet large number of resumes are received on-line, through e-mails or through services provided by companies (like info edge limited [1]) for searching resumes. The transmission of resumes directly to employers became increasingly popular as late as 2002. Job-seekers were able to circumvent the job application process and reach employers through direct email contact and resume blasting, a term meaning the mass distribution of resumes to increase personal visibility within the job market.

For Human Resource (HR) managers it has become a difficult and time consuming process to select appropriate resume from such large set of resumes. Currently available techniques or services employed by these enterprises help their HR managers filter thousands of resumes to some hundred potential ones. Since these filtered resumes are similar to each other, manual sifting through each resume becomes essential to select appropriate candidates. In this chapter, we have investigated the problem of resume selection from set of similar resumes and proposed an efficient framework to solve the same.

In the chapter we mention the general issues that are being faced in the process of resume extraction and resume selection. In the next section we explain the problem of resume selection that we addressed in this chapter. Then we model the structure of resume and mention the issues that are faced while applying the product selection framework. Next section describes the proposed

[1]Info Edge is a leading provider of online recruitment, matrimonial, real estate and educational classifieds and related services in India.

approach that explains how we extend the notion of special features and overcome the issues in applying product selection framework to solve the problem of resume selection. Finally we present the experimental results to validate the proposed approach and also mention briefly about the work related to the problem of resume selection.

4.1 Issues in Resume Selection

(i) Structure of resume highly varies. Even for a restricted domain, resumes are written in multitude of formats (e.g. structured tables or plain text), in different languages, different file types (e.g. Text, PDF, Word etc.) and are diversified. Thus automatically extracting structured information from resumes is a challenge [12].

(ii) As resumes contain free text written by humans it becomes very difficult to compare the text between two resumes. Even though resumes share similar information, variation in writing style makes it difficult to mine useful information from a set of resumes.

(iii) Thousands of resumes are received by big enterprises. Search services help them to refine from thousands to hundreds of resumes. Since these refined resumes are similar, user needs to browse each resume manually to select an appropriate candidate. Browsing through hundreds of resume manually is a tough job. We define this as 'Problem of Resume Selection'.

There has been some work on resume information extraction to extract structured information from resumes. Most of the work done on resumes focuses on information extraction of resume or building a classifier to extract the information from resume and storing it in structured manner.

4.2 Problem of Resume Selection

In the preceding section we discussed about some of the issues that are being faced in Resume Selection process. We have made an effort to investigate the problem of resume selection. The problem of resume selection is defined as follows: Big enterprises and corporations receive thousands of resumes every day. Currently available techniques or services employed by these enterprises help their HR managers to filter from thousands of resumes to hundred potential ones. Since these filtered resumes are similar to each other, they have to manually look through each resume to select the appropriate candidate. We have investigated the problem of resume selection from set of similar resumes and proposed an efficient framework to solve the same.

It can be observed that each resume contain several sections, and each section contains different kinds of text. For example, experience section contains long sentences, skills section contains skill

type (programming languages) and skill values (c++, java). So, development of an approach to process the resume dataset is a complex task, as several approaches have to be developed for dealing with each type of text.

We have proposed an approach by considering only skills related information of the resumes. We consider that there may exist special information in some resumes when compared to others. For example, a resume may contain specialty in education, specialty in experience, special skills or special achievements. Special information may exist in one or more sections of a resume. Thus identifying such special information and organizing them efficiently helps in improving the performance of resume selection process. We extended the notion of special features and proposed an approach to identify resumes with special skills and special skill values.

4.2.1 Motivation

It can be observed in a group of students from similar streams like computer science, electronics etc. there are some common skills which are possessed by all the students in the group and also each student possess some special skills that differentiate it from rest of the students in the group.

At the time of applying for a job each student post his resume to the selected companies of his choice. A resume is a document that contains a summary of relevant job experience and education. The resume is typically the first item that a potential employer encounters regarding the job seeker and is typically used to screen applicants, often followed by an interview, when seeking employment. Each student tries to reflect his/her special skills in the resume. Thus if we can extract such special skills for each student and organize it in efficient manner it would help the Human resource managers to select the appropriate resumes efficiently.

In the subsequent sections we model the structure of a resume and discuss the difficulties in applying the product selection framework to resume selection problem. Then we explain the proposed approach to provide a solution to resume selection process and present the experimental results to validate the proposed approach.

4.2.2 Modeling of Resumes

Resumes share document-level hierarchical contextual structure where the related information units usually occur in the same textual block and text blocks of different information categories usually occur in relatively fixed order [12]. Resume information is described as a hierarchical structure with several layers. The first layer consists of different sections such as education, experience, skills and experience. Second layer consists of text describing each section that acts as features for respective

Table 4.1: Sample Resume with corresponding sections and their respective features

Education
1. b.tech. (computer science & engineering) iiit, hyderabad (expected may, 2009) 6.66/10 cgpa.
2. senior secondary instrumental school, kota (cbse board 2004) 72%.
3. secondary st. sr. sec. school, ajmer (cbse board 2002) 83%.
Skills
1. programming languages: c, c++
2. operating systems: windows 98/2000/xp, gnu/linux
4. scripting languages: shell, python
5. web technologies: html, cgi, php
6. other tools: microsoft office, latex, gnu/gcc, visual studio 2005/08
7. database technologies: mysql
Experience
1. audio-video conferencing over ip networks:
2. duration: nov. 2007 nov. 2008 team size: 2. technical environment: c++ abstract: the objective of this project was to develop an audio/video conferencing system which enables multiple users to communicate with each other via a global server with improved efficiency in terms of voice clarity and low latency. the system is equipped with resources to facilitate text chat, voice chat and voice/video chat between multiple clients. this client server application was developed using c++ and .net framework in windows environment.
3. windows firewall
4. duration: july-nov 2007 team size: 1 technical environment: c abstract: packets from or to a network are analyzed and according to the users settings actions are taken on how the packets would be handled. various options are provided to the user in accordance to which action is taken ranging from what the packet contains to the source of the packet.
5. document request form automation
6. duration: sep-nov 2006 team size: 2 technical environment: php, mysql abstract: project developed for iiit hyderabad administration. this web-based tool automates the processing of the various documents.
7. implementation of outer loop join
8. duration: jan-march 2007 team size: 1 abstract: implementation of the above operation as a part of the database management systems course.
9. myshell
10. duration: aug-oct 2006 team size: 1 abstract: developed a program which acted as a shell, starting and running command line arguments as part of our operating systems course.
11. other studies and presentations
Achievements
1. secured 1573 air in all india engineering entrance examination, 2005.
2. secured 2216 air in iit-jee screening examination, 2005
3. cleared national talent search examination level 1 in 2002.
4. was among the finalists of the rajasthan state science talent search

sections separated by a delimiter.

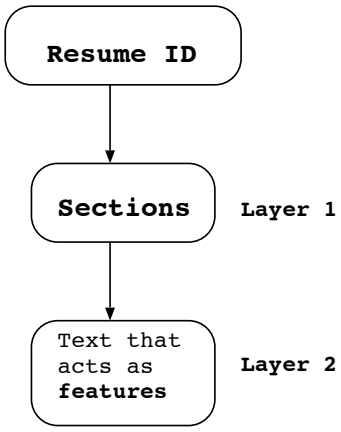

Figure 4.1: Hierarchical structure of Resume

Table 4.1 shows the sample resume and Figure 4.1 shows hierarchical structure for resume. Sections like education, experience, skills and achievements form the first layer of resume i.e 'Sections'. Text describing each section numbered from 1 to n is the 'features' for each section and forms the second layer of the resume.

We divide the resume into a hierarchical structure as resume is a multi-topic document where each section describes a different aspect of an individual. The special information identified on merging all the sections would be disorganized and difficult to understand for the HR managers. Thus we divide the resume into a hierarchical structure.

4.2.3 Issues faced in applying product selection framework

There are several issues being faced while applying the product selection framework in resume scenario.

1. There is no standard format for resume and thus structure of resume is highly varied where in case of products there is a definite structure of its specification.

2. The structure of resume is hierarchical and features are organized in several layers whereas in products features are organized in a single layer.

3. In products, features are standardized in the form of their specifications, whereas in resumes features consist of text written by humans. Thus it is easy to compare features between two similar products, whereas it becomes difficult to compute similarity between two features in case of resumes. Preprocessing of resumes becomes a complicated process as same text is written in different forms by different people.

The next section describes how the proposed approach tries to overcome some of the above issues to apply the product selection framework on resume selection process.

4.3 Proposed Approach

In this section we explain how we overcome the above issues to apply the product selection framework to solve the problem of resume selection.

The first and second issues are dealt during the preprocessing stage described in section 4.3.3 In order to extract structured skills information from the resume we have used rule based approaches and we have used canonical matching for standardization of skills in resumes.

In this thesis, we have narrowed down our problem to identifying 'special skills' in a resume to improve the process of resume selection. Given a set of similar resumes, we propose to identify special skills in each resume if it exists.

Table 4.2 shows the example of features for skill section. As in case of resumes it is difficult to directly compare two features we divide each skill feature into two parts i.e. skill type and skill value. For example programming languages would be skill type and c, c++, java would be skill value for this skill type. Thus in the proposed approach we divide each feature in skills section into two types of features also shown in Figure 4.2 :

1. Skill type
2. Skill value

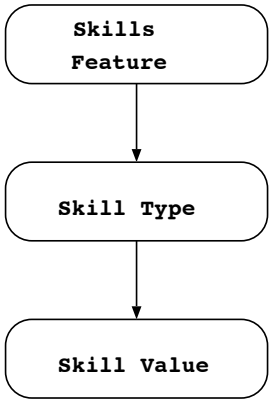

Figure 4.2: Hierarchical structure of Skills

The problem of resume selection is to identify the special skill type for each resume, then special skill value for each of the skill type and organize them efficiently. It would help the HR managers

to select the appropriate resumes with less effort. Hence in a resume skills data spreads over two levels while Product features data has only one level. Hence the algorithm has to be applied at two levels in Resume skills data i.e firstly identifying special skill type and then special skill value for respective skill type.

Before going into the problem we also define the term similar resumes. Similar resumes are set of resumes that HR managers get after filtering through their own resume management systems.

Table 4.2: Sample Features for Skill Tag

Skill Type features	**Skill Values features**
1. programming languages	c, c++,java.
2. programming languages	c, c++ java.
3. scripting languages	python, perl, shell
4. scripting languages	python perl shell
5. libraries	opengl, sdl
6. database technologies	mysql mssql
7. operating systems	linux, windows

4.3.1 Identifying Special Skill Type

In this section, we describe how the proposed framework can be applied to identify special skill type from given set of resumes.

Let R be a set of 'n' similar resumes, where resume $r_i \in$ R. Each resume r_i possess some set of features. Let f(r_i) be set of skill type features for resume r_i. Let F be set of all skill type features for all resumes such that F = $\cup_{i=1}^{n} f(r_i)$. Each feature F is denoted by f_j where $0 \leq j \leq |F|$ and n(f_j) denote the number of resumes to which feature f_j belongs. Note that, the set F may contain duplicate features. We consider these features as distinct features because they belong to distinct resumes.

First we assign degree of specialness (DS) value to all the features in set F using the notion of degree of specialness as described in Chapter 3 (Section 3.3.1). Then on the basis of DS values, we organize the skill type features in set F using three-level organization approach described in Chapter 3 (Section 3.3.2).

4.3.2 Identifying Special Skill Value

In this section, given a skill type and its corresponding skill values we identify the special skill value for respective skill type.

Let S be a set containing distinct skill type features from all the resumes and $s_j \in$ S denotes a

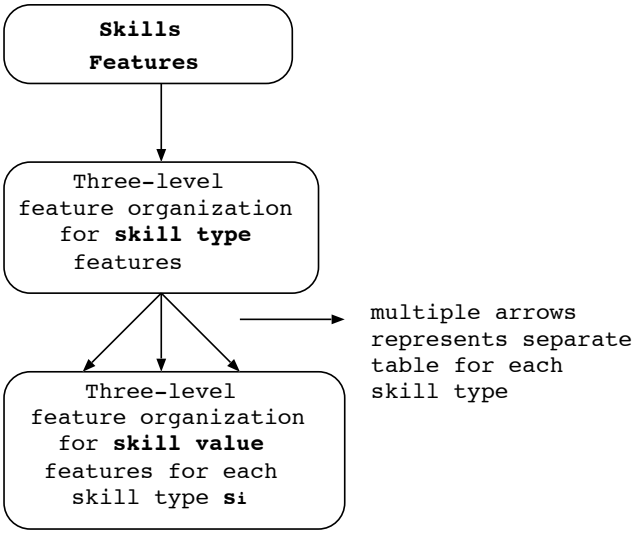

Figure 4.3: Hierarchical Feature Organization

particular skill type. All distinct skill type features are shown in Table 4.3. Let f(s_{ij}) denotes the skill value features for skill type $s_j \in$ S and resume $r_i \in$ R. For example programming language is an skill type s_j, resume ID 1 denotes r_i and c++ denotes one of the feature in set f(s_{ij}). Let F(s_j) be set of all skill value features for skill type s_j for all the resumes such that F(s_j) = $\cup_{i=1}^{n} f(s_{ij})$.

We apply the framework proposed in Chapter 3 for each skill type $s_j \in$ S consisting of feature set F(s_j) one by one to organize the skill value features for each skill type.

Thus the skill section features are organized into a hierarchical structure as shown in figure 4.3. The first level shows three-level organization of skill type features and second level shows the three-level organization of skill value for each distinct skill type.

4.3.3 Overall framework

The input to the proposed approach are the resumes stored as text documents where each text document contains different sections along with their features separated by a delimiter (Table 4.1). The steps of the proposed framework are discussed below (refer Figure 4.4).

1. **Preprocessing:**

   - Extracting structured skills information from resume: The skill section in resume is identified with the keyword 'skill' in the heading. This resolves the issue of uncertain structure of a resume because the skill section placed anywhere in the resume can be extracted. The skill type and its skill value(s) are identified and separately stored using

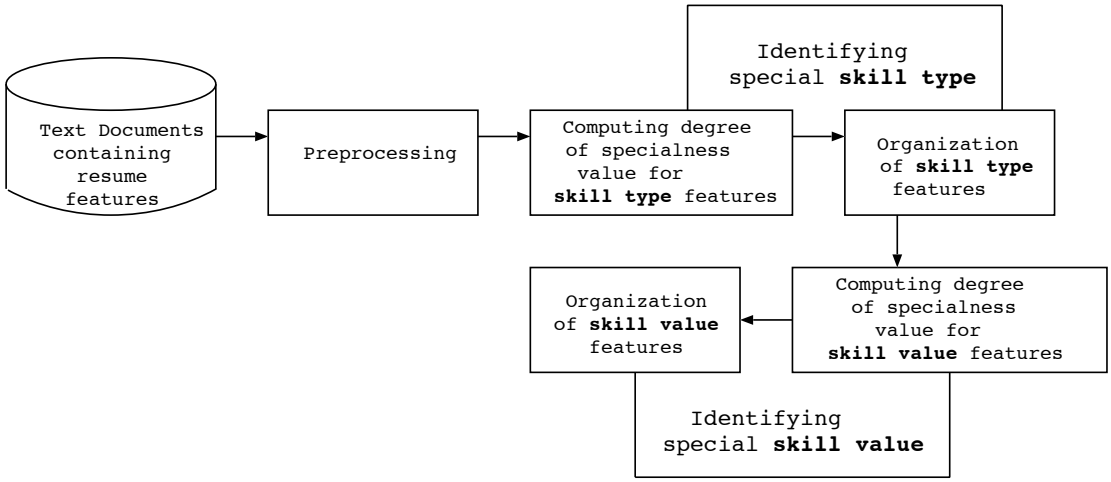

Figure 4.4: Flow diagram of the overall framework

a delimiter (: in our case).

- Entire input text is converted to lower case.
- Skill value(s) corresponding to each skill type are sorted lexically and separated by a comma (,). For a skill value having more than one word, the words are concatenated. It can be observed in Table 4.2 that skill values for the skill type programming languages (feature 1 and 2) are same except the comma and white space. Same is the case in skill values for the skill type scripting languages (feature 3 and 4). Thus we remove such special characters from skill values. For example, database technologies: ms sql, postgres sql, mysql.

  Corresponding skill value string would be: mssql, mysql, postgressql.
- Canonical Names: The issue of ambiguity in free-form text of resumes is dealt with by using the notion of Canonical Names. A skill value can have more than one name referring to it. For example mssql and microsoft sql refers to same skill. All the different names should be treated the same to avoid ambiguity. Hence we identify the various possibilities through data analysis and prepare a Hash Table with the Canonical Name (or Common Name) as the Hash Key and various other possible names as a list (Hash values) corresponding to the Canonical Name.

  We define a data structure called "Skill Values list". Each skill value in a skill type is checked against the list of possible skill values corresponding to that skill type. If a partial match is found, the skill value is replaced by the skill value from the list. If no match is found, the list is manually updated (need to be check before updating) with the skill value. We maintain a Hash list with each skill type as a Hash key. The possible skill values are extracted from the resume dataset.

This resolves the issue of ambiguity and redundancy due to lack of strict specifications.

- Functions are used to remove text which does not create visible output.
- Stop words [2] occurring in general purpose stop words list are removed.

2. **Identifying Special Skill Type:** Firstly DS value of skill type features are computed and based on DS value skill type features are organized.
   - **Computing DS value for Skill Type features:** We compute the DS value for skill type features.
   - **Organization of Skill Type features:** We organize the skill type features into three-level feature organization based on their specialness value.
3. **Identifying Special Skill Values:** For each skill type, the DS value for all its skill value are computed and based on the DS values skill value features are organized.
   - **Computing DS value of Skill value features:** We compute the DS value for skill value features for each skill type $s_i \in S$
   - **Organization of Skill Value features:** Skill value features for each skill type are organized using three-level feature organization. For each skill type, its skill value is organized in a table.

4.4 Experimental Results

To evaluate the performance, we have applied the proposed framework on real world data-set of resumes and reported the results. The details of the data-set are as follows:
Data-set contains 100 resumes from fourth year students of computer science department in a University. All the resumes are available in the same format as shown in Table 4.1. Total number of features in skills tag were 643. Table 4.3 shows distinct skill type present in the data-set.

Table 4.3: All Skill Type Features

Skill Type features	
1. programming languages	2. scripting languages
3. operating systems	4. web technologies
5. database technologies	6. libraries
7. other tools	8. compiler tools
9. mobile platforms	10. middleware technologies

[2]Stop words is the name given to words which are filtered out prior to, or after, processing of natural language data (text)

4.4.1 Evaluation Metric

We define the performance metric called 'reduction factor' (rf) to measure the performance improvement. The rf denotes the reduction in the number of features that the HR manager needs to browse to select a resume from set of 'n' similar resumes as compared to one-level approaches.

Let 'F' denote the total number of features for all the resumes, F(i) denote the number of features in 'i'-level and 'L' denotes the number of levels. The 'rf' is defined as,

$$rf = 1 - \frac{\sum_{i=1}^{L} F(i)}{F}$$

4.4.2 Preprocessing

As mentioned in section 4.3.3 preprocessing includes converting text into same case, extracting skill type and skill values, removing special characters, stopwords and resolving the issue of ambiguity using the notion of canonical names. The Table 4.4 mentions some of the examples from the data-set.

Table 4.4: Preprocessing

1. Removing unwanted characters: c,c++, python => c c++ python (separating commas) 2000/xp/vista => 200 xp vista (shell scripting), => shell scripting
2. Extracting skill types and skill values using skill tag and delimiters to separate skill values from skill types. < *skill* > programming languages: c, c++,java. scripting languages: python, perl, shell libraries: opengl, sdl < /*skill* >
3. Resolving canonical names: shell => shellscripting sql => mssql microsoft sql => mssql javascripting => javascript javascripts => javascript access => msaccess sql => mssql symbian(basic => symbian openc++(basic => openc++ fedora linux => linux gnu linux => linux

4.4.3 Experimental Results

In this section, we discuss the experimental results.

Results for Skill Type features

Feature statistics: Table 4.5 shows the feature statistics for skill type features. Feature statistics table indicate the statistics of data-set obtained after applying the proposed approach. It shows the number of common features, total number of clusters, number of features and number of products for each cluster, number of special features, total number of features present in the data-set and total features that are being displayed using the three-level organization approach. The total numbers of features present were 643 and numbers of features being display to user are only 73. There is such large reduction in number of features as number of features are present as common features so instead of displaying them for each resume, it's been displayed only once. Similarly the cluster features are displayed once for all the resumes present in a cluster instead of separately displaying for each one. These feature statistics are used to calculate the performance of the proposed approach.

Feature organization: Table 4.6 shows the organization of skill type features (set F) using three-level approach. I-level shows the common skill type, II-level shows the common skill type for each cluster of resumes and III-level shows the special skill type for each resume. The resumes can be classified based on their skill type in one click. Since number of resumes share same special feature we have mentioned them in same row separated by delimiter comma (,) for user convenience as well as to reduce space.

Performance: The reduction factor for skill type features comes out to be 88%. Thus there is 88% reduction in the effort of the HR managers in resume selection process.

Table 4.5: Skill Type Feature Statistics (III-level)

Number of Common Features = 4
Number of Clusters = 5
Features in Cluster 1 = 2, Resumes in Cluster 1 = 67
Features in Cluster 2 = 2, Resumes in Cluster 2 = 6
Features in Cluster 3 = 2, Resumes in Cluster 3 = 9
Features in Cluster 4 = 1, Resumes in Cluster 4 = 4
Features in Cluster 5 = 1, Resumes in Cluster 5 = 8
Total Common Cluster Features = 8
Special Features = 67
Total Features displayed = 4 (Common Features) +
8 (Common Cluster Features) + 67 (Special Features) = 79
F = (4 * 100) + (2 * 67 + 2 * 6 + 2 * 9 + 1 * 4 + 1 * 8) + 67 = 643

Table 4.6: Organization of features (skill type)

Common Features (I-level)		
programming languages operating systems web technologies database technologies		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
other tools scripting languages	78	1. Libraries 2. middleware technologies 3.mobile platforms
	83,13,91,114,112,52,67,54, 15,108,105,109,107	1.compilers 2. mobile platforms
	25,5	1.libraries 2. mobile platforms
	36,77,15	1.Compiler tools
	43,44,70,71,76,69,40,39,50, 57,62,59,101,95,65,68,84,90, 110,88,19,113,6,3,32,92,29, 104,93	1.libraries
	9,98,86,73,79,8,66,28,115, 103,106,11,37,24,53,55,47, 1,41	None
Libraries Other tools	63,35	1.Compiler tools
	89,94,17,49	None
scripting languages libraries	75,96,45,2,58,33,14,10,51	None
other tools	82,27,99,38	None
scripting languages	111,4,22,23,21,46,42,16	None

Results for Skill Value features

Feature Statistics: Table 4.7 shows the feature statistics for skill value features of skill type database technologies. Performance is calculated based on these feature statistics. Similar tables are formed for other skill value features and are shown in **Appendix**.

Feature Organization: Tables 4.8 shows III-level feature organization for database technologies skill values features. I-level shows the common skill values, II-level shows the common skill value for each cluster of resumes and III-level shows special skill values for each resume.
The results for other skill value features are mentioned in the **Appendix**.

Performance: Table 4.9 shows the reduction factor in the skill value features for each skill type. It can be observed that in most of the cases reduction factor is above 50% except in some cases where number of features is very less. Thus we can say that there is on average 50% reduction in the efforts of HR managers in resume selection process.

To summarize we can say that the resumes are organized based on skill type in first layer and second layer consists of tables for skill value of each skill type. Thus, the HR manager can select a resume based on the skill by browsing through only two tables, firstly skill type and then skill value for chosen skill type.

Table 4.7: Skill Value (database technologies) Feature Statistics (III-level)

Number of Common Features = 0
Number of Clusters = 3
Features in Cluster 1 = 2, Products in Cluster 1 = 58
Features in Cluster 2 = 1, Products in Cluster 2 = 29
Features in Cluster 3 = 1, Products in Cluster 3 = 3
Total Common Cluster Features = 4
Special Features = 6
Total Features displayed = 0 (Common Features) +
4 (Common Cluster Features) + 6 (Special Features) = 10
F = (0 * 100) + (2 * 58 + 1 * 29 + 1 * 3) + 6 = 154

Table 4.8: Organization of features (skill value :: database technologies) using three-level approach for Resume data-set

Common Features (I-level)		
None		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
mssql, mysql	10, 101, 105, 107, 114, 115, 14, 16, 19, 24, 25, 29, 3, 32, 33, 35, 36, 39, 4, 40, 41, 42, 46, 47, 5, 50, 53, 54, 55, 57, 58, 59, 6, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 73, 75, 76, 77, 78, 84, 86, 88, 90, 92, 93,98	none
	103, 112	postgresql
	51	oracle
mysql	1, 108, 109, 11, 110, 111, 15, 17, 2, 21, 22, 23, 26, 27, 28, 37, 43, 44, 45, 79, 8, 87, 89, 9, 91, 95, 96	none
	104	msaccess
	113	jdbc, postgresql
mssql	38, 49, 94	none

Table 4.9: Reduction Factor values for skill type and skill value

Feature Type	$\|F\|$	$\sum_{i=1}^{L} F(i)$	**rf**
database technologies	148	10	0.93
programming languages	283	38	0.86
scripting languages	150	66	0.56
compiler tools	41	7	0.83
mobile platforms	4	3	0.25
middleware technologies	3	3	0
libraries	170	91	0.56
web technologies	354	154	0.34
operating systems	271	16	0.94
other tools	302	208	0.31

4.5 Summary of the Chapter

Selecting appropriate resume from a group of similar resumes is one of the problems faced by HR managers in most of the companies. In this chapter, we have investigated resume selection issue and proposed an improved approach to help the HR managers in extracting the special skills for each resume from given set of resumes. We extended the product selection framework and proposed an approach to effectively extract special skills from given set of similar resumes. We divide the problem into two parts i.e identifying special skill type and identifying special skill value for each skill type. The proposed solution will help the HR managers to browse the special skills for each candidate by looking into the tables of skill type and skill value for chosen skill type.

We have conducted experiments on the real world data-set of resumes and the results indicate that the proposed approach has the potential to improve the process of resume selection.

As a part of future work, we plan to extend our idea for other sections like experience, education and so on. Also we are planning to carry out intensive experiments by considering different kinds of data sets to develop a generalized approach.

Chapter 5

Conclusion and Future Work

In this chapter we mention the summary, conclusion and future work.

5.1 Summary

We have identified two different issues and proposed an improved framework to solve the same. The first issue is related to the problem of product selection from similar products and second issue relates to the problem of resume selection.

Selecting appropriate product from a group of similar products is one of the problems faced by the customer in E-commerce environment. Companies offer several similar products with little deviation to provide more options for the customers. As a result, a customer faces difficulties while selecting a product in E-commerce environment due to the problem of information overload. For selecting a product from a set of similar products, a customer has to go through a large amount of information. It is very difficult and time consuming process to identify the specialty of each product manually.

We have investigated the product selection issue and proposed an improved approach to solve the same. We have introduced the notion of degree of specialness (ds) to identify special features of the products and calculated the 'ds' value for each feature. We also plotted a graph between feature rank and 'ds' values that helped us to come up with two different kinds of feature organization approaches. Using the 'ds' values we proposed a clustering algorithm to organize the features into Two-level and Three-level organization of features.

With the help of experimental results on 3 real-time data-sets we have shown that the notion of special feature can reduce the number of features browsed by the user by 50% to 80% and thus

reducing the effort of the user to select a product by same amount. The proposed approach helps the users in reducing the product selection time and also helps E-commerce companies by increasing the sales and user satisfaction.

The second issue deals with identifying special features to improve the process of resume selection. Selecting appropriate resume from a group of similar resumes is one of the problems faced by HR managers in most of the companies. We have extended the notion of special features to extract special skills from given set of similar resumes.

We have proposed an approach to identify resumes with special skills type and special skill values. The skill related information can be divided into two layers, skill types and skill values. It can be observed that, given the resume, the HR manager may want to identify the resumes with 'special skill values' or 'special skill value' of a given skill type. For example, HR manager may want the resumes which contain special skill type 'web technologies' or HR manager may want the resumes in which 'php' is a special skill value included in web technologies skill type. So, the proposed approach improves the performance of resume processing by extracting both special skills and special skill values. With the help of the experimental results we have shown that there is 50-88% reduction in the number of features that an HR manager needs to browse through to select appropriate resumes.

5.2 Conclusion

In the post period, customers would choose from whatever products were available. All the products manufactured had distinct features. The user has to browse through the features for number of products to select an appropriate product.

Today, mass production and mass customization has in many cases produced an over supply of very similar goods and, in particular, services. There are number of similar products manufactured but with very little deviation in their features. Each product has very few special features and shares number of common features with other similar products. Thus the notion of special features can be exploited to help the users in efficient product selection. The users need not browse through all the features for each of the product separately. We can use the notion of special features to provide the user with an efficient organization of features such that user can view the features for all the similar products as quickly as possible. From the experimental results we can conclude that the notion of special feature can help the buyers as well as sellers in e-commerce environment. The buyers can efficiently select an appropriate product from the set of similar products and sellers can increase their profit by satisying the user and increasing the sale of their products.

In the second part of the thesis we showed how the notion of special features can be extended to improve the process of resume selection system. In the past, resumes used to be no longer than two pages, as potential employers typically did not devote much time to reading resume details for each applicant. Since increasing numbers of job seekers and employers are using Internet-based job search engines to find and fill employment positions, longer resumes are needed for applicants to differentiate and distinguish themselves, and employers are becoming more accepting of resumes that are longer than two pages. Many professional resume writers and human resources professionals believe that a resume should be long enough so that it provides a concise, adequate, and accurate description of an applicant's employment education, experience, skills and achievements. Thus the process of resume selection has actually become a complex process as a resume has been divided into several sections such as education, experience, skills, achievements etc. Each section contains different type of text and thus different kind of techniques have to be used to process each section of the resume.

In this thesis we have exploited the skills section of the resume. We divide the problem into two phases' i.e. identifying special skill type and identifying special skill value for each skill type. In the first phase we identify the specialness value of skill type features and organize them using three-level approach and in the second phase we identify the specialness values of skill value features for each skill type and organize them using three-level approach. From the experimental results we can conclude that the proposed solution has a potential to help the HR managers in selecting appropriate resumes by browsing through the tables of skill type and skill value for chosen skill type.

5.3 Future Work

In case of product selection problem we are planning to carry out extensive experiments by considering different kinds of data sets to develop a generalized approach.

In the case of resume selection problem, we plan to extend our idea for other sections like experience, education and so on. It can also be observed from the results, that there is scope of classification of resumes based on skill type and skill value. We plan to explore the same idea and design an efficient system for classification of resumes.

Chapter 6

Appendix

The tables for feature statistics and feature organization for other skill value features are presented. Table 6.1 shows the naming for feature statistics and feature organization for different skill value features.

Table 6.1: Naming of Tables for Skill Value Features

Table No.	Table Name
Table 6.2	feature statistics of programming languages
Table 6.3	feature organization for programming languages
Table 6.4	feature statistics of compiler tools
Table 6.5	feature organization for compiler tools
Table 6.6	feature statistics of operating systems
Table 6.7	feature organization for operating systems
Table 6.8	feature statistics of mobile platforms
Table 6.9	feature organization for mobile platforms
Table 6.10	feature statistics of middleware technologies
Table 6.11	feature organization for middleware technologies
Table 6.12	feature statistics of libraries
Table 6.13	feature organization for libraries
Table 6.14	feature statistics of scripting languages
Table 6.15	feature statistics of web technologies
Table 6.16	feature organization for scripting languages
Table 6.17	feature organization for web technologies
Table 6.18	feature statistics of other tools
Table 6.19	feature organization for other tools

Table 6.2: Skill Value (programming languages) Feature Statistics (III-level)

Number of Common Features = 2
Number of Clusters = 13
Features in Cluster 1 = 2, Products in Cluster 1 = 3
Features in Cluster 2 = 2, Products in Cluster 2 = 3
Features in Cluster 3 = 2, Products in Cluster 3 = 2
Features in Cluster 4 = 2, Products in Cluster 4 = 2
Features in Cluster 5 = 1, Products in Cluster 5 = 6
Features in Cluster 6 = 1, Products in Cluster 6 = 24
Features in Cluster 7 = 1, Products in Cluster 7 = 3
Features in Cluster 8 = 1, Products in Cluster 8 = 3
Features in Cluster 9 = 1, Products in Cluster 9 = 4
Features in Cluster 10 = 0, Products in Cluster 10 = 1
Features in Cluster 11 = 0, Products in Cluster 11 = 1
Features in Cluster 12 = 0, Products in Cluster 12 = 1
Features in Cluster 13 = 0, Products in Cluster 13 = 1
Total Common Cluster Features = 13
Special Features = 23
Total Features displayed = 2 (Common Features) + 13 (Common Cluster Features) + 23 (Special Features) = 38
F = (2 * 100) + (2 * 3 + 2 * 3 + 2 * 2 + 2 * 2 + 1 * 6 + 1 * 24 + 1 * 3 + 1 * 3 + 1 * 4 +
0 * 1 + 0 * 1 + 0 * 1 + 0 * 1) + 23 = 283

Table 6.3: Organization of features (skill value :: programming languages) using three-level approach for Resume data-set

Common Features (I-level)		
c, c++		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
javascript, msil	101, 25	none
	112	j2ee
javascript, python	17	latex
	19	.net, vc++
	35	batchscripting, shellscripting
matlab, python	38	perl, php
	62	none
javascript, vb	47	none
	87	symbian
python	1, 27, 68, 82, 89	none
	63	shellscripting
javascript	10, 103, 104, 111, 113, 21, 24, 3, 33, 37, 39, 43, 49, 5, 51, 53, 6, 75, 78, 94, 98	none
	11	Perl
	54	j2me, socketprogramming
	99	Lisp, vhdl
Nasm	106, 108	none
	109	Msil, oz
matlab	14, 4, 90	none
Mips	69, 76, 86, 92	none
	36	Socketprogramming
	55	Actionscript, mxml
	57	openc++, symbian
	77	Prolog

Table 6.4: Skill Value (compiler tools) Feature Statistics (III-level)

Number of Common Features = 0
Number of Clusters = 2
Features in Cluster 1 = 2, Products in Cluster 1 = 18
Features in Cluster 2 = 0, Products in Cluster 2 = 1
Total Common Cluster Features = 2
Special Features = 5
Total Features displayed = 0 (Common Features) + 2 (Common Cluster Features) + 5 (Special Features) = 7
F = (0 * 100) + (2 * 18 + 1 * 0) + 5 = 41

Table 6.5: Organization of features (skill value :: compiler tools) using three-level approach for Resume data-set

Common Features (I-level)		
None		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
lex, yacc	105, 107, 108, 109, 112, 114, 13,	none
	15, 36, 52, 54, 64, 67, 77, 83	
	26, 91	phoenix
	35	phoenix, rdk
	63	phoenix

Table 6.6: Skill Value (operating systems) Feature Statistics (III-level)

Number of Common Features = 2
Number of Clusters = 6
Features in Cluster 1 = 2, Products in Cluster 1 = 3
Features in Cluster 2 = 2, Products in Cluster 2 = 2
Features in Cluster 3 = 1, Products in Cluster 3 = 36
Features in Cluster 4 = 1, Products in Cluster 4 = 14
Features in Cluster 5 = 1, Products in Cluster 5 = 4
Features in Cluster 6 = 0, Products in Cluster 6 = 1
Total Common Cluster Features = 7
Special Features = 7
Total Features displayed = 2 (Common Features) + 7 (Common Cluster Features) + 7 (Special Features) = 16
F = (2 * 100) + (2 * 3 + 2 * 2 + 1 * 36 + 1 * 14 + 1 * 4 + 0 * 1) + 7 = 271

Table 6.7: Organization of features (skill value :: operating systems) using three-level approach for Resume data-set

Common Features (I-level)		
linux, windowsxp		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
9x, vista	25, 54	none
	46	redhat, suse
vista, windows98	57, 59	none
vista	10, 13, 19, 21, 22, 27, 28, 29, 3, 33, 35, 37, 39, 40, 41, 42, 5, 51, 52, 53, 55, 6, 64, 65, 67, 75, 77, 78, 8, 82, 83, 84, 87, 84	none
	11	windowsnt
	32	ubuntu
9x	103, 104, 108, 110, 112, 114, 115, 2, 58, 70, 71, 93, 95, 96	none
	105	windowsnt
windowsnt	50, 73, 86	none
	109	redhat
	99	os161

Table 6.8: Skill Value (mobile platforms) Feature Statistics (III-level)

Number of Common Features = 0
Number of Clusters = 2
Features in Cluster 1 = 1, Products in Cluster 1 = 2
Features in Cluster 2 = 0, Products in Cluster 2 = 1
Total Common Cluster Features = 1
Special Features = 2
Total Features displayed = 0 (Common Features) + 1 (Common Cluster Features) + 2 (Special Features) = 3
F = (0 * 100) + (1 * 2 + 0 * 1) + 2 = 4

Table 6.9: Organization of features (skill value :: mobile platforms) using three-level approach for Resume data-set

Common Features (I-level)		
None		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
android	78	none
	5	symbian
	25	android1

Table 6.10: Skill Value (middleware technologies) Feature Statistics (III-level)

Number of Common Features = 0
Number of Clusters = 1
Features in Cluster 1 = 0, Products in Cluster 1 = 1
Total Common Cluster Features = 0
Special Features = 3
Total Features displayed = 0 (Common Features) + 0 (Common Cluster Features) + 3 (Special Features) = 3
F = (0 * 100) + (0 * 1) + 3 = 3

Table 6.11: Organization of features (skill value :: middleware technologies) using three-level approach for Resume data-set

Common Features (I-level)		
None		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
	78	Corba, ejb, j2ee

Table 6.12: Skill Value (libraries) Feature Statistics (III-level)

Number of Common Features = 0
Number of Clusters = 20
Features in Cluster 1 = 2, Products in Cluster 1 = 15
Features in Cluster 2 = 2, Products in Cluster 2 = 15
Features in Cluster 3 = 2, Products in Cluster 3 = 8
Features in Cluster 4 = 2, Products in Cluster 4 = 3
Features in Cluster 5 = 3, Products in Cluster 5 = 4
Features in Cluster 6 = 0, Products in Cluster 6 = 1
Features in Cluster 7 = 0, Products in Cluster 7 = 1
Features in Cluster 8 = 0, Products in Cluster 8 = 1
Features in Cluster 9 = 0, Products in Cluster 9 = 1
Features in Cluster 10 = 0, Products in Cluster 10 = 1
Features in Cluster 11 = 0, Products in Cluster 11 = 1
Features in Cluster 12 = 0, Products in Cluster 12 = 1
Features in Cluster 13 = 0, Products in Cluster 13 = 1
Features in Cluster 14 = 0, Products in Cluster 14 = 1
Features in Cluster 15 = 0, Products in Cluster 15 = 1
Features in Cluster 16 = 0, Products in Cluster 16 = 1
Features in Cluster 17 = 0, Products in Cluster 17 = 1
Features in Cluster 18 = 0, Products in Cluster 18 = 1
Features in Cluster 19 = 0, Products in Cluster 19 = 1
Features in Cluster 20 = 0, Products in Cluster 20 = 1
Total Common Cluster Features = 11
Special Features = 80
Total Features displayed = 0 (Common Features) + 11 (Common Cluster Features) + 80 (Special Features) = 91
F = (0 * 100) + (2 * 15 + 2 * 15 + 2 * 8 + 2 * 3 + 3 * 4 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1
+ 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1) + 80 = 170

Table 6.13: Organization of features (skill value :: libraries) using three-level approach for Resume data-set

Common Features (I-level)		
None		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
opengl, qt	113, 6, 70	none
	105	glut, sdl
	107	cg, glsl, sdl
	10	idevim, netbeans
	114	newmat, opencv , sdl
	2	sdl
	3	glut, gtk, sdl
	33, 35, 39	glut
	44	cuda, glut, latex, matlab, opencv, openscenegraph, sdl
	45	cuda, opencv, sdl
	58	multisim
opengl, sdl	112, 29, 40, 50, 51 67, 78, 96, 13	none
	109	ac3d
	110	opencv, qt3, qt4
	101	gtk
	19	net, openmp
	43, 88	glut
glut, opengl	25, 32, 49, 94, 95	none
	63	amigo, mpi, openmp
	65	cg, cuda, glsl, opencv
	108	sld
gl, qt	14, 68, 84	none
api, opengl, shakti	26, 91	mpi, openmp
	52, 83	sdl
	104	mfc, nokia, opengl, windowsmediaplayer
	17	matlab, opengl
	8	opengl, stl
	54	matlab, qt
	57	aria, opengl
	59	opengl
	62	opengl
	69	opengl
	89	opengl
	90	opengl
	19	opengl
	93	opengl
	71	api, qt, windowsdriverdevelopmentkit
	75	idevim, netbeans, opengl
	76	ffmpeg, opencv , opengl

Table 6.14: Skill Value (scripting languages) Feature Statistics (III-level)

Number of Common Features = 2
Number of Clusters = 13
Features in Cluster 1 = 2, Products in Cluster 1 = 17
Features in Cluster 2 = 2, Products in Cluster 2 = 51
Features in Cluster 3 = 2, Products in Cluster 3 = 3
Features in Cluster 4 = 2, Products in Cluster 4 = 4
Features in Cluster 5 = 0, Products in Cluster 5 = 1
Features in Cluster 6 = 0, Products in Cluster 6 = 1
Features in Cluster 7 = 0, Products in Cluster 7 = 1
Features in Cluster 8 = 0, Products in Cluster 8 = 1
Features in Cluster 9 = 0, Products in Cluster 9 = 1
Features in Cluster 10 = 0, Products in Cluster 10 = 1
Features in Cluster 11 = 0, Products in Cluster 11 = 1
Features in Cluster 12 = 0, Products in Cluster 12 = 1
Features in Cluster 13 = 0, Products in Cluster 13 = 1
Total Common Cluster Features = 8
Special Features = 58
Total Features displayed = 0 (Common Features) + 8 (Common Cluster Features) + 58 (Special Features) = 66
F = (0 * 100) + (2 * 17 + 2 * 51 + 2 * 3 + 2 * 4 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 +
0 * 1 + 0 * 1 + 0 * 1) + 58 = 150

Table 6.15: Skill Value (web technologies) Feature Statistics (III-level)

Number of Common Features = 1
Number of Clusters = 28
Features in Cluster 1 = 3, Products in Cluster 1 = 20
Features in Cluster 2 = 3, Products in Cluster 2 = 6
Features in Cluster 3 = 3, Products in Cluster 3 = 3
Features in Cluster 4 = 3, Products in Cluster 4 = 4
Features in Cluster 5 = 3, Products in Cluster 5 = 5
Features in Cluster 6 = 3, Products in Cluster 6 = 3
Features in Cluster 7 = 3, Products in Cluster 7 = 2
Features in Cluster 8 = 3, Products in Cluster 8 = 3
Features in Cluster 9 = 3, Products in Cluster 9 = 4
Features in Cluster 10 = 2, Products in Cluster 10 = 12
Features in Cluster 11 = 2, Products in Cluster 11 = 2
Features in Cluster 12 = 2, Products in Cluster 12 = 3
Features in Cluster 13 = 2, Products in Cluster 13 = 3
Features in Cluster 14 = 2, Products in Cluster 14 = 2
Features in Cluster 15 = 2, Products in Cluster 15 = 2
Features in Cluster 16 = 2, Products in Cluster 16 = 2
Features in Cluster 17 = 0, Products in Cluster 17 = 1
Features in Cluster 18 = 0, Products in Cluster 18 = 7
Features in Cluster 19 = 0, Products in Cluster 19 = 1
Features in Cluster 20 = 0, Products in Cluster 20 = 1
Features in Cluster 21 = 0, Products in Cluster 21 = 1
Features in Cluster 22 = 0, Products in Cluster 22 = 1
Features in Cluster 23 = 0, Products in Cluster 23 = 1
Features in Cluster 24 = 0, Products in Cluster 24 = 1
Features in Cluster 25 = 0, Products in Cluster 25 = 1
Features in Cluster 26 = 0, Products in Cluster 26 = 1
Features in Cluster 27 = 0, Products in Cluster 27 = 1
Features in Cluster 28 = 0, Products in Cluster 28 = 1
Total Common Cluster Features = 41
Special Features = 112
Total Features displayed = 1 (Common Features) + 41 (Common Cluster Features) + 112 (Special Features) = 154
F = (1 * 100) + (3 * 20 + 3 * 6 + 3 * 3 + 3 * 4 + 3 * 5 + 3 * 3 + 3 * 2 + 3 * 3 + 3 * 4
+ 2 * 12 + 2 * 2 + 2 * 3 + 2 * 3 + 2 * 2 + 2 * 2 + 2 * 2 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1
+ 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1) + 112 = 354

Table 6.16: Organization of features (skill value :: scripting languages) using three-level approach for Resume data-set

Common Features (I-level)		
None		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
php, python	101, 50, 55	none
	10, 75, 86	cgi, modpython, shellscripting
	104	javascript, shellscripting
	106	cgi, oz
	110, 66, 88	shellscripting
	113	cgi, shellscripting
	15, 19	Bash
	24	asp, jsp, .net ,shellscripting
	47	j2ee, jsp, servlets
	8	cgi, jsp, servlets
python, shellscripting	105, 109, 11, 111, 112, 114, 14, 16, 2, 21, 22, 23, 28, 3, 32, 33, 36, 37, 39, 4, 40, 46, 5, 51, 53, 57, 59, 62, 65, 69, 70, 71, 77, 78, 79, 83, 9, 90, 95, 96, 99	none
	103	perl
	108	awk, perl, sed
	115, 41	javascript
	26, 54, 84, 91	perl
	6	powershell
	92	cgi, modpython
Perl, python	107	none
	67	bash
	68	cgi
Bash, python	76	none
	29	javascript
	43	javascript, php, server
	44	cgi, modpython
shellscripting	1, 25	none
	38	bash
	42	python
	45	actionscript, cgi, perl ,php
	64	python
	73	python
	99	bash
	58	lisp, php, python, shellscripting

Table 6.17: Organization of features (skill value :: web technologies) using three-level approach for Resume data-set

Common Features (I-level)		
html		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
cgi, php, xml	105, 107, 109, 114, 3, 75	none
	103	ajax, css, j2ee, javascript, jsp, servlets
	10	jsp
	112	css, javascript
	15, 2, 40	css
	25	xhtml
	35, 39	css, javascript, jsp
	37	css, javascript
	41	ajax, css, jsp, servlets
	51	css, ibmmq
	77	css, javascript, modpython, python
	79	ajax, css, python
cgi, css, javascript	104	ajax, apache, iis, perl, python, xml
	110	php, xhtml
	32	jsp, modpython, python, servlets, xml
	57, 59	modpython, php
	76	modpython, xml
cgi, jsp, servlets	113	ejb, xml
	17	ajax, css, javascript, php, xml
	19	php
cgi, modpython, php	26, 65, 91	none
	27	latex, python
cgi, css, php	28, 46, 62, 66, 70	none
cgi, php, python	29	xhtml
	67	css, xhtml
	93	jsp
cgi, modpython, python	36, 69	none
ajax, css, xml	43	javascript, jsp, servlets, xhtml
	5	cgi, javascript, modpython, php
	53	asp.net, php
css, javascript, php	71	ajax, asp.net, xhtml
	82	actionscript, flex, xhtml
	87	asp.net, jsp, servlets, xml
	99	ajax, jsp, servlets
cgi, php	101, 108, 111, 16, 21, 50, 73, 84, 9, 90	none
	23	javascript
	96	svg
css, php	11	none
	89	xml
cgi, javascript	13, 63, 83	none
cgi, xml	55, 78	none
	33,	css
css, javascript	45	flex
	8	xml
php,xml	47	none
	88	javascript
css, xml	6	none
	86	xhtml
	1	php
	14	cgi
	22	cgi
	4	cgi
	52	cgi
	54	cgi
	92	cgi
	98	cgi
	24	ajax, javascript
	42	javascript
	64	cgi, python
	95	php

Table 6.18: Skill Value (other tools) Feature Statistics (III-level)

Number of Common Features = 0
Number of Clusters = 37
Features in Cluster 1 = 3, Products in Cluster 1 = 5
Features in Cluster 2 = 3, Products in Cluster 2 = 5
Features in Cluster 3 = 3, Products in Cluster 3 = 3
Features in Cluster 4 = 3, Products in Cluster 4 = 2
Features in Cluster 5 = 3, Products in Cluster 5 = 2
Features in Cluster 6 = 3, Products in Cluster 6 = 3
Features in Cluster 7 = 4, Products in Cluster 7 = 2
Features in Cluster 8 = 4, Products in Cluster 8 = 4
Features in Cluster 9 = 3, Products in Cluster 9 = 3
Features in Cluster 10 = 3, Products in Cluster 10 = 4
Features in Cluster 11 = 3, Products in Cluster 11 = 2
Features in Cluster 12 = 3, Products in Cluster 12 = 2
Features in Cluster 13 = 3, Products in Cluster 13 = 2
Features in Cluster 14 = 2, Products in Cluster 14 = 3
Features in Cluster 15 = 2, Products in Cluster 15 = 4
Features in Cluster 16 = 2, Products in Cluster 16 = 2
Features in Cluster 17 = 2, Products in Cluster 17 = 2
Features in Cluster 18 = 2, Products in Cluster 18 = 4
Features in Cluster 19 = 0, Products in Cluster 19 = 1
Features in Cluster 20 = 0, Products in Cluster 20 = 1
Features in Cluster 21 = 0, Products in Cluster 21 = 1
Features in Cluster 22 = 0, Products in Cluster 22 = 1
Features in Cluster 23 = 0, Products in Cluster 23 = 1
Features in Cluster 24 = 0, Products in Cluster 24 = 1
Features in Cluster 25 = 0, Products in Cluster 25 = 1
Features in Cluster 26 = 0, Products in Cluster 26 = 1
Features in Cluster 27 = 0, Products in Cluster 27 = 1
Features in Cluster 28 = 0, Products in Cluster 28 = 1
Features in Cluster 29 = 0, Products in Cluster 29 = 1
Features in Cluster 30 = 0, Products in Cluster 30 = 1
Features in Cluster 31 = 0, Products in Cluster 31 = 1
Features in Cluster 32 = 0, Products in Cluster 32 = 1
Features in Cluster 33 = 0, Products in Cluster 33 = 1
Features in Cluster 34 = 0, Products in Cluster 34 = 1
Features in Cluster 35 = 0, Products in Cluster 35 = 1
Features in Cluster 36 = 0, Products in Cluster 36 = 1
Features in Cluster 37 = 0, Products in Cluster 37 = 1
Features in Cluster 38 = 0, Products in Cluster 38 = 1
Features in Cluster 39 = 0, Products in Cluster 39 = 1
Total Common Cluster Features = 51
Special Features = 157
Total Features displayed = 0 (Common Features) + 51 (Common Cluster Features) + 157 (Special Features) = 208
F = F = (0 * 100) + (3 * 5 + 3 * 5 + 3 * 3 + 3 * 2 + 3 * 2 + 3 * 3 + 4 * 2 + 4 * 4 + 3 * 3
+ 3 * 4 + 3 * 2 + 3 * 2 + 3 * 2 + 2 * 2 + 2 * 4 + 2 * 2 + 2 * 2 + 2 * 4 + 0 * 1 + 0 * 1 +
0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1
+ 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1 + 0 * 1) + 157 = 302

Table 6.19: Organization of features (skill value :: other tools) using three-level approach for Resume data-set

Common Features (I-level)		
None		
Common cluster features (II-level)	**ResumeId**	**Special Features (III-level)**
gcc, matlab, microsoftoffice	1, 93	none
	55	adobeflexbuilder, adobephotoshop cs3, mssql, visualstudio
	63	visualstudio
	70	adobephotoshop, latex
adobephotoshop, gimp, microsoftoffice	36, 37	none
	105	latex, matlab
	17	idevim, netbeans
	103	dreamweaver
gcc, latex, matlab	107	none
	39	adobephotoshop, netbeans
	86	cuda, gpu
latex, matlab, netbeans	11	idevim
	112	gnuplot
eclipse, netbeans, qt	110	j2ee, j2me, jdbc, jsp, matlab, quanta, servlets
	43	idevim
idevim, latex, matlab	26, 91, 92	none
	13	hadoop, lucene, microsoftoffice
	57	gimp, microsoftoffice, mobilesim
	59	gimp, microsoftoffice
adobephotoshop, dreamweaver, matlab, microsoftoffice	25	none
	71	adobefireworks, gcc, gimp latex, visualstudio
adobephotoshop, gimp, idevim, latex	29	none
	66	matlab, microsoftoffice, nettbeans
	69	matlab, microsoftoffice, multisim
	89	gcc, microsoftoffice
eclipse, idevim, netbeans	41	adobephotoshop, dreamweaver, msoffice
	82	3dsmax, adobefireworks, adobeflexbuilder, adobeillustrator, adobephotoshop, matlab, adobephotoshoppremiere, dreamweaver
	99	awk, batchscripting, javamail jdbc, latex , mssql
idevim, matlab, microsoftoffice	83	none
	52	eclipse, kile
	65	eclispse, gimp
	44	3dsmax, itk
adobephotoshop, gcc, microsoftoffice	49, 94	none
frontpage, microsoftoffice, picasa	62	none
	90	latex, visualstudio
gcc, latex, microsoftoffice	76	idapro
	9	visualstudio
gcc, matlab	101	none
	109	qt
	35	visualstudio
gcc, visualstudio	40	none
	19	cygwin
	104	codewarrior, eclipse, symbian
	53	neatbeans
matlab, visualstudio	108	qt
	3	adobephotoshop, idevim
microsoftoffice, netbeans	113	emacs, idevim, visio
	32	gimp, latex
latex, matlab	29, 6, 64, 67	none

Common cluster features (II-level)	ResumeId	Special Features (III-level)
	106	microsoftoffice
	15	gcc, glomosimnetworksimulator
	24	visualstudio
	47	visualstudio
	27	eclipse, idevim
	38	gcc, gdb, latex, msoffice, ssh
	5	manager, microsofthpcserver2008, microsofthyper-v, microsoftoperationmanager, microsoftsystemcentervirtualmachinemanager, virtualizationtechandtools
	50	adobephotoshop, gcc
	54	idevim, lucene, microsoftoffice rainbowclassifier, visualstudio, weka
	68	adobephotoshop, microsoftoffice, nlptools
	73	adobephotoshop, opengl, sdl
	77	carbide, eclipse
	78	eclipse, netbeans
	79	apache, cms, dovecot, drupal, eclipse gcc, gimp, git, tcpdump, wireshark, wordpress
	8	netbeans
	84	latex, microsoftoffice
	88	apache, microsoftoffice
	95	gimp, microsoftoffice, qt
	98	gcc, hadoop, idevim
	114	gcc, gnu, gpu, latex, matlab, msil

Publications

1. Sumit Maheshwari and P.Krishna Reddy, Discovering Special Product Features for Improving the Process of Product Selection in E-commerce Environment, in the proceedings of the 11th International Conference on Electronic commerce (ICEC 2009), August, 11-15, 2009, Taipai, Taiwan, Published by ACM.

2. Sumit Maheshwari, Abhishek Sainani, and P.Krishna Reddy An Approach to Extract Special Skills to Improve the Performance of Resume Selection 6th International Workshop on Databases in Networked Information Systems (DNIS 2010), The University of Aizu, JAPAN, Mar 29-31, 2010, Lecture Notes in Computer Science, vol. 5999, Springer-Verlag, 2010.

Bibliography

[1] http://www.shopping.hp.com. 20 June 2008.

[2] http://www.mobile.am/. 24 July 2008.

[3] http://www.sonystyle.com/. 28 September 2008.

[4] Charu C. Aggarwal and Philip S. Yu. Outlier detection for high dimensional data. In *SIGMOD '01: Proceedings of the 2001 ACM SIGMOD international conference on Management of data*, pages 37–46, New York, NY, USA, 2001. ACM.

[5] Malik Agyemang, Ken Barker, and Rada S. Alhajj. Mining web content outliers using structure oriented weighting techniques and n-grams. In *SAC '05: Proceedings of the 2005 ACM symposium on Applied computing*, pages 482–487, New York, NY, USA, 2005. ACM.

[6] Malik Agyemang, Ken Barker, and Rada S. Alhajj. Wcond-mine: Algorithm for detecting web content outliers from web documents. *Computers and Communications, IEEE Symposium on*, 0:885–890, 2005.

[7] Malik Agyemang, Ken Barker, and Reda Alhajj. Framework for mining web content outliers. In *SAC '04: Proceedings of the 2004 ACM symposium on Applied computing*, pages 590–594, New York, NY, USA, 2004. ACM.

[8] Fabrizio Angiulli, Fabio Fassetti, and Luigi Palopoli. Detecting outlying properties of exceptional objects. *ACM Trans. Database Syst.*, 34(1):1–62, 2009.

[9] V. Barnett and T Lewis, editors. *Outliers in statistical data*. John Wiley and Sons, Chichester, 1994.

[10] Vinayak Borkar, Kaustubh Deshmukh, and Sunita Sarawagi. Automatic segmentation of text into structured records. *SIGMOD Rec.*, 30(2):175–186, 2001.

[11] Markus M. Breunig, Hans-Peter Kriegel, Raymond T. Ng, and Jörg Sander. Lof: identifying density-based local outliers. *SIGMOD Rec.*, 29(2):93–104, 2000.

[12] Soumen Chakrabarti, Martin Van Den Berg, and Byron Dom. Focused crawling: a new approach to topic-specific web resource discovery. In *Computer Networks*, pages 1623–1640, 1999.

[13] F. Ciravegna and A. Lavelli. Learningpinocchio: adaptive information extraction for real world applications. *Nat. Lang. Eng.*, 10(2):145–165, 2004.

[14] Hawkins D, editor. *Identification of Outliers*. Chapman and Hall, 1980.

[15] Martin Ester, Hans-Peter Kriegel, and Matthias Schubert. Web site mining: a new way to spot competitors, customers and suppliers in the world wide web. In *KDD '02: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 249–258, New York, NY, USA, 2002. ACM.

[16] A. Finn and N. Kushmerick. Multi-level boundary classification for information extraction. In *In Proceedings of ECML04*, 2004.

[17] Alfonso Valdes Harold S. Javitz. The sri ides statistical anomaly detector. In *Proceedings, 1991 IEEE Symposium on Security and Privacy, Oakland CA, May 1991*, May 1991.

[18] P. Helman and J Bhangoo. A statistically based system for prioritizing information exploration under uncertainty. In *In IEEE International Conference on Systems, Man, and Cybernetics. Vol. 27*, pages 449–466, 1997.

[19] B. Joseph Pine II, editor. *Mass Customization*. Harvard Business School Press, 1993.

[20] M Kaplan, A.M; Haenlein. Toward a parsimonious definition of traditional and electronic mass customization. In *Journal of product innovation management*, 2006.

[21] Vorapranee Khu-smith, Chris J. Mitchell, and Royal Holloway. Enhancing e-commerce security using gsm authentication. In *In the Proceedings of the EC-Web 2003, 4th International Conference on Electronic Commerce and Web Technologies*, 2002.

[22] Willi Klösgen. Explora: a multipattern and multistrategy discovery assistant. pages 249–271, 1996.

[23] Edwin M. Knorr and Raymond T. Ng. Algorithms for mining distance-based outliers in large datasets. In *VLDB '98: Proceedings of the 24rd International Conference on Very Large Data Bases*, pages 392–403, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[24] Edwin M. Knorr and Raymond T. Ng. Finding intensional knowledge of distance-based outliers. In *VLDB '99: Proceedings of the 25th International Conference on Very Large Data Bases*, pages 211–222, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc.

[25] Bing Liu, Yiming Ma, and Philip S. Yu. Discovering unexpected information from your competitors' web sites. In *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 144–153, New York, NY, USA, 2001. ACM.

[26] Bing Liu, Kaidi Zhao, and Lan Yi. Visualizing web site comparisons. In *WWW '02: Proceedings of the 11th international conference on World Wide Web*, pages 693–703, New York, NY, USA, 2002. ACM.

[27] E. Johnston N. Kushmerick and S. McGuinness. Information extraction by text classification. In *In IJCAI01 Workshop on Adaptive Text Extraction and Mining.*, 2001.

[28] Rohit Paravastu, Hanuma Kumar, and Vikram Pudi. Uniqueness mining. In *DASFAA*, pages 84–94, 2008.

[29] Emanuel Parzen. On estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.

[30] F. Peng and A. McCallum. Accurate information extraction from research papers using conditional random fields. In *In Proceedings of HLT/NAACL-2004*, pages 329–336, 2004.

[31] L. Peshkin and A. Pfeffer. Bayesian information extraction network. In *In Proceedings of IJCAI03*, pages 421–426, 2003.

[32] J. Ben Schafer, Joseph Konstan, and John Riedi. Recommender systems in e-commerce. In *EC '99: Proceedings of the 1st ACM conference on Electronic commerce*, pages 158–166, New York, NY, USA, 1999. ACM.

[33] A. D. Sitter and W. Daelemans. Information extraction via double classification. In *In Proceedings of ATEM03*, 2003.

[34] Jian-Tao Sun, Xuanhui Wang, Dou Shen, Hua-Jun Zeng, and Zheng Chen. Cws: a comparative web search system. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 467–476, New York, NY, USA, 2006. ACM Press.

[35] J Tseng, M.M.; Jiao. Mass customization, in: Handbook of industrial engineering, technology and operation management. In *NY: Wiley*, 2001.

[36] etal Wei L, Qian W. Hot: Hypergraph-based outlier test for categorical data. In *In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, pages 399–410, 2003.

[37] Wikipedia. Masscustomization.

[38] Weng-Keen Wong, Andrew Moore, Gregory Cooper, and Michael Wagner. Bayesian network anomaly pattern detection for disease outbreaks. In *In Proceedings of the Twentieth International Conference on Machine Learning*, pages 808–815. AAAI Press, 2003.

[39] Kun Yu, Gang Guan, and Ming Zhou. Resume information extraction with cascaded hybrid model. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 499–506, Morristown, NJ, USA, 2005. Association for Computational Linguistics.

[40] Ji Zhang, Qigang Gao, and Hai Wang. A novel method for detecting outlying subspaces in high-dimensional databases using genetic algorithm. In *ICDM '06: Proceedings of the Sixth International Conference on Data Mining*, pages 731–740, Washington, DC, USA, 2006. IEEE Computer Society.

[41] Ji Zhang and Hai Wang. Detecting outlying subspaces for high-dimensional data: the new task, algorithms, and performance. *Knowl. Inf. Syst.*, 10(3):333–355, 2006.

[42] Cui Zhu, Hiroyuki Kitagawa, and Christos Faloutsos. Example-based robust outlier detection in high dimensional datasets. In *ICDM '05: Proceedings of the Fifth IEEE International Conference on Data Mining*, pages 829–832, Washington, DC, USA, 2005. IEEE Computer Society.