Design and Implementation of an Enterprise Entity Extraction System Based on a Multi-Source Data Match Model

Kan Lin
Abstract

Design and Implementation of an Enterprise Entity Extraction System Based on a Multi-Source Data Match Model

Kan Lin

With the rapid development of the Internet and the increase in the number of online users, there is a growing interest in commerce for SMEs. However, it remains a concern as how to precisely describe the small to medium companies in the field of SME enterprises.

The goal of this thesis is to set a service-oriented matching model based on the study of existing enterprise matching models and data format. In this thesis, the author is going to address the model that is able to present E-commerce SMEs through three dimensions — reputation, geographic information and enterprise type. The model would enable the buyers to find better target enterprises from the range of SMEs and at the same time allow SMEs to present more reliable and complete information to the buyers.

The thesis is going to develop and evaluate the prototype based on the matching model.
Table of Contents

1 Introduction .............................................................................................................. 1
  1.1 Background........................................................................................................... 1
  1.2 Thesis research purpose and main content ......................................................... 2
    1.2.1 Research Purposes......................................................................................... 2
    1.2.2 Research Contents ...................................................................................... 3
  1.3 Thesis Structure .................................................................................................. 3

2 Relevant knowledge and theories ............................................................................ 5
  2.1 Data Type ............................................................................................................ 5
    2.1.1 Structured Data .............................................................................................. 5
    2.1.2 Unstructured Data ........................................................................................ 5
    2.1.3 Semi-Structured Data .................................................................................... 5
  2.2 Enterprise Matching Model .................................................................................. 6
    2.2.1 Zachman Framework .................................................................................... 6
    2.2.2 FEAF Framework ........................................................................................ 6
    2.2.3 RM-ODP ...................................................................................................... 6
  2.3 Introduction of Related Technology ...................................................................... 7
    2.3.1 Google API ................................................................................................... 7
    2.3.2 Zemanta API ................................................................................................ 8
    2.3.3 MVC Design Model ...................................................................................... 8
    2.3.4 AJAX ........................................................................................................... 9
    2.3.5 JSP .............................................................................................................. 10

3 Dimension Matching Model ..................................................................................... 11
  3.1 Definition ............................................................................................................ 11
    3.1.1 Reputation ................................................................................................... 11
    3.1.2 Geographic Information ............................................................................... 12
    3.1.3 Type ............................................................................................................ 13

4 Prototype Design and Implementation .................................................................... 17
  4.1 Architecture ...................................................................................................... 17
  4.2 Prototype Design and Implementation ................................................................ 18
1 Introduction

1.1 Background

In today’s world, with the rapid development of the Internet and the dramatic increase of the number of the Internet users, the various kinds of data which are available on the Internet are exponential increasing as well. Nowadays, the Internet has already been a world-wide magnanimity information releasing and sharing space and has become an indispensable and extremely important part of people’s daily life.

Meanwhile, the global business environment has experienced a great revolution. The traditional economic model has changed a lot. The globalization and networking lead to the arise of e-business world-wide. E-business can bring a lot of opportunities for the running and development of enterprises. The unlimited prospects of e-business has caught people’s eyes around the world and people are using the Internet to do various kinds of e-business. On the other hand, rapid, widely, and deeply developed e-business is challenging and also enriching traditional trading method. Part of traders and enterprises’ income are created by e-business while people are prevalently realizing the development of e-business is the great impulse power of the new century’s global economy.

As a very important representation of e-business users, small and medium-sized enterprises limited by its capital and human resource which make them differ from big enterprises and cannot have an integrated e-business plan and strategy. Their information on the Internet has a multi-source phenomenon. Because the Internet self is a highly distributed system, those small and medium-sized enterprises can release their information (name, address, phone, scope of business and so on) on many different websites and these information can be different as well.

Under normal circumstances, the SMEs in the different sites related to enterprise information has a certain correlation, which makes processing become possible to integrate these information.

Today, the Internet has developed into a global, mass distribution and shared information space. Although on the Internet, there are a large number of data sources involved in various professional fields, the vast majority of them do not provide direct access permissions to ordinary users of the network. Instead of, these data sources use a website will contain the information embedded in the webpage of related websites, appear in the form of an HTML document on the Web and to limit network users can only browse through the web browsers to view the information data in order to hide themselves in the website or spinal dorsal mirage from the user direct accessing. However, data shown on network in this way is lacking of description of the data itself. The semantic information of these data is relative obscurity and the mode is less clear as well which will
ultimately lead to some applications cannot be directly resolved and the usage of the vast amounts of information on the Web is also limited. This can be a waste of network resources.

In addition, most of the current network users of online information queries rely on Web search engines. And search engines can only use keyword matching to look up vast amounts of web-pages, then return the keyword contained web link to the user group as their searching results. However, most pages contain their subject information unrelated "noise", such as the navigation bar, advertising links, etc., so the accuracy of the search engine returns results may be disturbed. What's more, in order to obtain the required information, the user must individually artificial browse the results one by one to find the required data information which lead to a lot of time wasted.

With a growing number of SMEs begin to know or use the Internet, more and more information about SMEs will be appeared on the Internet. This may cause the Internet has different source of data which on the same business entity. The Internet self is a highly distributed system, enterprises can release their information (name, address, phone, scope of business and so on) on many different websites and these information can be different as well. However, on the essence, these information have a certain degree of correlation.

Generally, we cannot visually extract and filter these enterprise information, thus establish the intrinsic link between information. However, the Internet is not only a place for information releasing, it is a set of computing power and service capacity combination as well. People can use some web services provided by companies on the Internet (mainly in the enterprises’ opening API) to extract and filter the data on the Internet so as to integrate correlation information and to provide expected results to users.

1.2 Thesis research purpose and main content

In this section, my thesis research's aim, content and research significance will be elaborated in turn.

1.2.1 Research Purposes

This subject is based on the theory of the e-commerce field of small and medium-sized enterprise and oriented by corporate matching model to establish an effective reflecting dimension match model for SMEs. And a prototype model is also established to be used to verify.

If the dimension matching model is proven to be effective, there are several potential uses:

Firstly, the model can provide a new classification method to SMEs (three-dimensional way: reputation, location and type)

Secondly, the model can be a good guideline for SMEs' business development.
1.2.2 Research Contents

The main content of the thesis includes the following sections:

1. Research of data type
   The existing form of the data is divided into: unstructured data, semi-structured data and structured data.

2. Research of corporate matching model

3. How to be able to describe the small and medium-sized enterprises in the field of electronic commerce
   Traditional data matching model is difficult to accurately describe the e-commerce field of small and medium-sized enterprises, so we propose a matching model based on the dimensions of the data to more accurately describe small and medium-sized enterprises in the field of electronic commerce. The model can be described by the three-dimensional coordinates: reputation, position information and the type of business. Visible as shown below:

The part of content will be detailed introduced in the third section 'dimension matching model'.

1.3 Thesis Structure

This thesis attempts to establish a new service-oriented matching model that could better describe the information about small and medium sized enterprises from the field of electronic commerce. This model has a good prospect because it
is not only favorable for buyers to better seek small and medium sized enterprises in the field of electronic commerce, but also makes such enterprises available to numerous buyers in more real and complete forms. Based on the matching model, this thesis develops the prototype, verifies and evaluates the matching model that has been constructed.

The thesis is organized as follows:

Chapter 1, namely the Introduction, introduces the dissertation background and demonstrates the research purposes, significance and structure of this dissertation.

Chapter 2 describes the knowledge and theories related to this thesis, mainly including the form of data and corporate matching model.

Chapter 3 illustrates the dimensional data matching model proposed in this thesis, which can better depicts small and medium sized enterprises from the field of electronic commerce.

Chapter 4 demonstrates the design and realization of a prototype system based on dimensional matching model.

Chapter 5 primarily presents the test and analysis for the prototype system. The analysis results suggest that the dimensional data matching model brought forth in this thesis can better describe small and medium sized enterprises from the field of electronic commerce.

Chapter 6 makes a summary of this thesis.
2 Relevant knowledge and theories

In this chapter, the author introduces some theories related with this thesis including data type, enterprise matching model and so on.

2.1 Data Type

There are three main data types: structured data, unstructured data and semi-structured data. They will be illustrated separately in the following text.

2.1.1 Structured Data

The so-called structured data can be stored in the database, and it can use two-dimensional table structure to achieve logically expression. The most common form of structured data is the database like SQL or Access. For instance, SQL (Structured Query Language) allows you to choose particular pieces of information based on columns and rows in a field. You may look for all the rows containing a specific date or name or IP code -- this is structured data, searchable and organized by data type within the actual content.

2.1.2 Unstructured Data

Unstructured data doesn’t have any identifiable structure. Unstructured data normally contains bitmap images/text, objects and other data types which are not belonging to a database[9]. Most enterprises’ data today can actually be regarded as unstructured. An email may be regarded as unstructured data. Even though those email messages are organized in a database, such as Lotus Notes and Microsoft Exchange, the body of the message is actually freeform text and not having any structure at all – therefore the data is considered raw. Documents are another sample of unstructured data. In spite of the fact that a Word document has some formatting attached to it, the content of the document is totally a free form.

2.1.3 Semi-Structured Data

Semi-structured data refers to a form of structured data that isn’t consistent with the formal structure of tables and data models related to relational databases; however it contains tags or other markers to disconnect semantic elements and enforce hierarchies of records and fields within the data. Hence, it has also been known as schemaless or self-describing structure.

In the semi-structured data, the entities belongs to the same class might not have the same attribute even though they are classified together, and the order of attributes is not significant.
2.2 Enterprise Matching Model

During this section, I will introduce the mainstream enterprise matching model in detail.

2.2.1 Zachman Framework

The Zachman Framework provides a regular and highly structured means to observe and define an enterprise. It is an Enterprise Architecture framework for enterprise architecture. Based on the intersection of six communication questions (What, Where, When, Why, Who and How) with six rows, and according to reification transformations, it consists of a two dimensional classification matrix.

The Zachman Framework is not a methodology, because it lacks specific ways and processes for gathering, managing, or utilizing the information that it describes. It was John Zachman who first developed the concept of Framework in the 1980s at IBM. It was named after him, and since then it has been updated several times.

The Zachman "Framework" is taxonomy for organizing architectural artifacts, such as design documents, specifications, and models. It takes into account both whom the artifact aims at, for instance, business owner and builder and what particular issue is being handled, for instance, data and functionality.

2.2.2 FEAF Framework

The Federal Enterprise Architecture, short for FEA, is a set of interrelated service-oriented reference models which are designed to conduct cross-agency analysis and the identification of investments, gaps and opportunities for cooperation within and across America government agencies.

As the FEA reference models attach great importance to the provision of services, it is a very useful and practical resource for organizations to develop a Service-oriented Architecture and classifying and defining their own Business Services (by using the BRM), Application Services (by using the SRM) and Infrastructure Services (by using the TRM). The example content is obviously based on the government services provision to its citizens, but the example may quickly be adapted for other industry sectors or other organizations.

2.2.3 RM-ODP

Based on exact concepts derived from current distributed processing developments, RM-ODP is a reference model on the use of formal description techniques for the specification of the architecture. Many RM-ODP concepts, which may be under different names, have appeared for quite a long time and have been rigorously explained and described in precise philosophy (for instance, in the works of Mario Bunge) and in systems thinking (for instance, in the works of Friedrich Hayek). Some of these concepts, such as composition, emergence and abstraction, have lately been provided with a solid mathematical foundation in
category theory.

RM-ODP has four basic elements. The first one is an object modeling approach to system specification; the second element is the specification of a system in terms of separate but interrelated viewpoint specifications; the second one is the definition of a system infrastructure providing distribution transparencies for system applications; and the last element is a framework for evaluating the system conformance.

The RM-ODP family of recommendations and international standards describes a system of interrelated essential concepts necessary to specify open distributed processing systems and provides a well-developed architecture framework for enterprise, structuring the specifications for any large-scale systems including software systems.

### 2.3 Introduction of Related Technology

Related technology will be introduced in this section. It contains three parts:

1. The first part is the introduction of API. Because prototype adopts many third-parts API, it is necessary to illustrate them systematically.

2. The second part discusses MVC model. Prototype uses MVC model as the architecture. This will reduce the coupling between business logic and view part.

3. The third part is programming technology. Programming technology is important to implementation of this prototype.

#### 2.3.1 Google API

Google API is a set of application programming interfaces provided by Google. It is freely available to programmers, so that programmers can use Google services by customized API.

##### 2.3.1.1 Google Custom Search API

Google Custom Search creates a solution which lets users to search web contents in a special way. Users can customize the search results and present the results in your website using API which is provided by Google. Your custom search engine can restrict or prioritize the search results based on the websites you specify.

- **Google Maps API Family**

  Google Maps has a wide array of APIs that let you embed the robust functionality and everyday usefulness of Google Maps into your own website and applications, and overlay your own data on top of them.

  The API is a free service; it can be visited on any website by users for free. Please see the terms of service for more information.\[13\]

  Businesses that track assets, charge accessing fees or build internal applications have to use Google Maps API Premier, which will provide technical support, service-level agreement and enhanced features.
2.3.2 Zemanta API

As a content suggestion engine for bloggers and other content creators, Zemanta examines and explains user-generated content (e.g. a blog post) by using semantic search technology and natural language processing to suggest tags, pictures and links to related articles.

Zemanta suggests content from Wikipedia, Amazon.com, Youtube, IMDB, ITIS, Crunchbase, Flickr, Facebook, Myspace, Musicbrainz, Mybloglog, NCBI, Rottentomatoes, Twitter, Snooth and Wikinvest, as well as the blogs of other Zemanta users.

Zemanta’s understanding of the content (for example, whether ”Apple” refers to a company or a fruit) could be inserted into the content by using Common Tag for semantic tagging.

Originally released for use by bloggers, Zemanta is available as a Firefox and Internet Explorer extension and plugins for Wordpress, Blogger, TypePad, Ning, Myspace, LiveJournal, MovableType, Tumblr, Drupal and Joomla. It now also can be used with web-based email systems, such as Yahoo mail and Gmail, and a Microsoft Outlook add-in is in development.

Zemanta is software of server-based, so the server does all the tough work and the plugins just communicate with the server hence to retrieve the suggestions.

Many other companies have utilized Zemanta’s API to add content suggestion to their products, for instance RetailFans, Bukisa, hover.in, Faviki and Triond.

Zemanta used to be partly based on TextGarden content analysis software from Slovenia’s Jožef Stefan Institute of national research institute.

The API sketch map could be seen below:

2.3.3 MVC Design Model

Model-view-controller, short for MVC, refers to a software architecture, now being considered an architectural pattern applied in software engineering. The
pattern isolates "domain logic" (the application logic for the user) from the user interface (input and presentation), permitting independent testing, development and maintenance of each (separation of concerns).

In spite of the fact that MVC comes in different flavors, control flow is basically as follows:

The user communicates with the user interface in some way (for instance, by pressing a mouse button).

The controller handles the input event from the user interface, often through a callback or registered handler, and converts the event into a suitable and acceptable user action, which is understandable to the model.

The controller informs the model of the user action, possibly leading to a change in the model's condition. (For instance, the controller updates the user's shopping trolley.)

A view queries the model in order to produce a suitable user interface (for instance the view lists the shopping cart's contents). The view obtains its own data from the model. In some implementations, the controller could release a main instruction to the view to render itself. In others, the view is automatically informed by the model of changes in state (Observer) that require a screen update.

The user interface waits for further user interactions, hence to restart the control flow cycle.

Some implementations such as the W3C XForms also adopt the concept of a dependency graph to operate the updating of views when data in the model changes.

By decoupling models and views, MVC's goal is to decrease the complexity in architectural design and to promote code's flexibility and maintainability.

2.3.4 AJAX

Ajax is a series of connected web development methods applied in the client-side to produce interactive web applications. With Ajax, web applications can retrieve data from and send data to, a server asynchronously (in the background) without influencing the behavior and display of the current page. Data is often retrieved by using the XMLHttpRequest object. Without being affected by the name, the use of XML is not needed (instead JSON is usually applied), and the requests don't need to be asynchronous.

Like LAMP and DHTML, Ajax is not one technology, but a series of technologies. Ajax applies a combination of CSS and HTML to mark up and style information. The DOM is accessed with JavaScript to dynamically display, and to allow the user to interact with the presented information. JavaScript and the XMLHttpRequest object offer a method for exchanging data asynchronously between browser and server hence to avoid full page reloads.

The term Ajax has come to stand for a large group of web technologies that can be applied for implementing a web application that connects with a server in the background, without influencing with the existing condition of the page. In
the paper that coined the term Ajax, Jesse James Garrett explained that the following technologies are integrated:

- HTML or XHTML and CSS for presentation
- the Document Object Model (DOM) for dynamic display of and interaction with data
- XML for the interchange of data, and XSLT for its manipulation
- the XMLHttpRequest object for asynchronous communication
- JavaScript to bring these technologies together

From then on, however, there have been huge developments in the technologies applied into an Ajax application, and the meaning of the term Ajax. Particularly, it has been known that JavaScript is not the only client-side scripting language that can be adopted for performing an Ajax application; other languages for example VBScript can also have the required functionality. (But, JavaScript is the most popular and welcomed language for Ajax programming because of its inclusion in and compatibility with the greater number of modern web browsers.) And also, XML is not required for data exchange and therefore XSLT is not required for the manipulation of data. JavaScript Object Notation (JSON) is usually used as another available format for data exchange, despite other formats such as preformatted HTML or plain text can also be put into use.

2.3.5 JSP

JavaServer Pages (JSP) technology makes it possible for designers and Web developers to quickly develop and easily maintain, information-rich, dynamic Web pages that supplement the existing business systems. As part of the Java technology family, JSP technology realizes the rapid development of Web-based applications that are platform independent. JSP technology separates the user interface from content generation, enabling designers to change the overall page layout without altering the underlying dynamic content.
3 Dimension Matching Model

In Chapter 2, we have learned that traditional enterprise matching model could not reflect the SME in E-commerce. Therefore, in this chapter the author will illustrate a new matching model for SMEs.

3.1 Definition

“Dimension Matching Model” is a three-dimensional model which represents a SME properly and accurately. The architecture could be shown as follows:

In this chapter, the author explains three dimensions separately.

3.1.1 Reputation

Enterprise reputation theory constructs its basic theory from the public impression and public comment, and then combined with intangible assets, customer satisfaction and customer loyalty. There is no doubt that the enterprise reputation associates with these aspects, but also something else. The most convincing definition of enterprise reputation comes from Relationship School, which regards the enterprise reputation as comprehensive understanding, evaluation and emotional connection according to stakeholders’ own experience,
enterprise behavior and information from major competitors. Generally, the enterprise reputation includes the following three aspects:

Firstly, enterprise image refers to how other people evaluate the enterprise, which is the first concerning area in the enterprise reputation study, includes the awareness of enterprises, evaluation and emotional connection from the consumers, the partners, the government and public interest organizations, etc.

Secondly, self-identity, also called organizational identification, refers to how the enterprise comments itself. There are two basic questions to answer in self-identity, “who we are” and “how we regard ourselves”. Thus, the self-identity is the recognition on enterprise organization level, which includes the enterprise stakeholders, board of directors, employees’ enterprise cognitive, evaluation and emotional connection. At the same time, enterprise self-identity is overlapped with enterprise culture to a large extent, because self-identity relates to enterprise behavior level and relates to “how we handle things here”.

Thirdly, expected identity, also known as enterprise identity, refers to what the company wants other people to regard themselves. Expected identity includes visual content such as denomination, logo and symbol, also includes strategic content such as vision, mission and philosophy. Expected identity is quite different from self-identity. If self-identity is the most relevant to the case of organization behavior, expected identity is more relevant with the strategic vision and organizational expectations, so this kind of identity is more on the field of marketing.

3.1.2 Geographic Information

This part will introduce “Geographic Information”. The author first illustrates its concept. Later tries to explain its manifestation.

- **Definition**

Geographic information for an enterprise means where the location is. For companies which deal with business on Internet, users do not care its physical location information. This is decided by the attributes of these companies: these companies always provide virtual services to end users and users do not care the location information for these companies. But for SMEs, location information is very important: they produce real foods; End users or customers need to know these foods.

Furthermore, the location for a factory of a company is very important for users because they need to consider the transportation cost between the factory and market. For an E-commerce SME, customers should have a full understanding about enterprise’s location information. The geographic information could contain:

- Location where enterprise headquarters is located
- Location where enterprise branch is located
- Enterprise production location
- Enterprise storage location

Generally speaking, an enterprise’s geographic information is discussed
above. But there are some exceptions I need to mention here:

(1) Company location information from Internet may be incorrect or inaccurate.

(2) Sometimes there is more than one location for a company. So it is necessary to get the total geographic information. The solution is to obtain mass data in Internet.

- Manifestation
  For a SME, the manifestation could be seen below:

![Geographic Information Diagram]

Especially for Chinese companies, the zip code is composed by six Arabic numerals.

3.1.3 Type

Type is a very important attribute for a company. It consists of the production information, its business scope and so on.

There are many classifying methods:

(1) Industry classification
  For this classification method, all enterprises are divided into different industries according to their scope of business. The description could be seen below:
This classification method could describe the scope of business of SMEs, but it has some problems:

1. One enterprise could manage several scopes. Using this classification method could not affect this phenomenon.
2. The granularity of this classification method is not high enough. It is not useful for customers or buyers.

(2) **Area classification**

Like its name implies, area classification divides companies according to their region.

This classification method could describe the region of SMEs, but it exists some problems:

1. For a company, the location where corporate headquarters and enterprise branch is located, enterprise production location, enterprise storage location may not refer to the same place. So this classification method could not describe SMEs correctly.
2. It could not describe the scope of business for a company. So it is not useful to help customers or buyers.

(3) **Tagging Classification**

Because there are some limits about above classifications, so it provides a new classification-Tagging Classification to describe SMEs.

The usage of tagging is already a highlight for personal applications on Internet. GMAIL uses tagging to classify mail information. Users could add some tags to mark every mail. The effect could be seen below:
Delicious as one of the biggest sharing websites in the world, it uses tagging for bookmarks. The effect could be seen below:

In this study, the author tries to apply tagging to business area, to observe whether tagging takes effect or not. More details could be seen below:

- **Scope of Business**
  The scope of business of a company could reflect this enterprise’s type. Buyers could ensure whether this company could produce foods which they want to buy. So according to tagging, buyers could save more time.
- Awards
  Awards which company gains could be regarded as the success of a witness. Through the tagging to show awards could improve company’s popularity.

- Famous Person
  A famous person could also improve a company’s popularity. Steve Jobs has greatly enhanced the popularity of Apple, and thus indirectly improves Apple’s performance according to his personal charisma.

- Exhibition Participation
  In order to improve its popularity, company would always participate in a variety of exhibitions. Through participating in these exhibitions, companies could show their products. So it is necessary to list exhibition participated for a company.

  In summary, through four types of tagging which describe the types of enterprises, we could distinguish the type of business enterprise effectively and accurately.
4 Prototype Design and Implementation

In this chapter, the author illustrates the prototype based on the dimension matching model including the design solution and implementation.

4.1 Architecture

Prototype could be divided into four parts:

- **Data Extraction Module**
  
  Data extraction module is a module which extracting data from internet. This module adopts Google Custom Search API to crawl the web contents. We need to customize the “search engine” according to the characteristics of E-commerce. The details will be explained later.

- **Data Structuring Module**
  
  Data structuring module aims to transfer unstructured data, semi-structured data to structured data. We use Google Language API and Zemanta API to accomplish this work.

- **Data Processing Module**
  
  Data processing module aims to process the structured data. The final target is to generate the useful data.

- **Data Display Module**
  
  Data display module is responsible for displaying the useful data on the screen. The system flow chart could be seen below:
4.2 Prototype Design and Implementation

In this section, design and implementation of the prototype will be illustrated. It includes the design and implementation of four modules: data extraction module, data structuring module, data processing module, and data display module.

Below is the network architecture of the system. It includes two types of servers: network servers and database servers. The detail structure graph can be seen below:
Application Layer can be divided into MVC model. Details can be seen below:
Prototype system adopts following technologies:

- Develop Tools: Myeclipse 8.5, Tomcat 6.0
- Developing Programming Language: JAVA JDK 6.0, JSP
- Platform: Windows 7
- Database: Mysql

4.2.1 Data Extraction Module

Data extraction module is a module which means extracting data from internet. This module adopts Google Custom Search API to crawl the web contents. We need to customize the “search engine” according to the characteristics of E-commerce.

The contents of customization include the following parts.

1. Scrawl scope. How many websites should we crawl? Is it necessary to limit the number of websites? Google Custom Search API provides some mechanism to limit the scrawl scope.

   Through limiting the scrawl scope, we could reduce number of unrelated information. This could increase the quality of data.

2. Scrawl number. Is it necessary to limit the number of websites? We consider we have two reasons to limit the scrawl number:

   - Increase quality. According to statistics, 80% useful contents appeared in the first two result pages.
   - Reduce cost. Because we limit the scrawl number, the cost of processing data could be reduced
The flow chart can be seen below:

The implementation code can be seen below:

```java
siteSearch = new google.search.WebSearch();
siteSearch.setUserDefinedLabel("阿里巴巴");
    siteSearch.setUserDefinedClassSuffix("siteSearch");
siteSearch.setSiteRestriction("china.alibaba.com");
searchControl.addSearcher(siteSearch);
```

Alibaba is a leading B2B service provider, and it has mass data. Here we use Alibaba website to limit scrawl scope.

### 4.2.2 Data Structuring Module

We should finish transferring unstructured, semi-structured data to structured data in this module. Here we use semantic tagging to do this work.

- **Semantic Tagging**
  
  With the development of semantic net, semantic tagging becomes a very hot research object. Now there are three focus points about semantic tagging: Polysemy, Synonymy and “Different lexical forms”.

  - **Polysemy**
    
    Polysemy means that one word has several meaning. Generally speaking, simpler spell words with more meaning. For example, set has 126 meaning and 26 different phrases in Oxford English Dictionary.

  - **Synonymy**
    
    Synonymy means the words which presents similar concepts but has different
forms or spell. For example, “toil” and “lavatory” presents similar concepts.

Statement could also appear synonymy. For example:
This book is very interesting.
There is a very interesting book.
Above two statements presents similar concepts. But it has different forms.

- Different lexical forms
Here “Different lexical forms” means different forms of words, verb conjugation and so on.

Nowadays, semantic tagging is not only a research object but also a product or service for people: Faviki, Zigtag, Twine and Fav are open to end users. Below the author will try to illustrate how Faviki and Zigtag solve these three problems.

1) Faviki
Faviki is a social bookmarking tool that lets you use Wikipedia concepts as tags.

Faviki allows you to keep your own tags and connect them to common, universal concepts from the world’s largest collection of knowledge!

Below two pictures could help to explain the effects about Faviki:
The solution for Faviki is shown below:

- **Polysemy**
  Faviki uses extra information to solve polysemy. For example, word orange could be expressed as a color as well as a fruit. Faviki uses ‘Orange(colour)’ and ‘Orange(fruit)’ to separate them.

- **Synonymy**
  Faviki adopts redirecting way to solve synonymy problem. For example, “电影” could be described as “Movie” or “Film”. The solution for Faviki is when user pointed to movie, it redirected to the ‘film’.

Below is the process chart:

(2) **Zigtag**
Faviki is a social bookmarking tool that lets you use Wikipedia concepts as tags.

The user interface could be seen below:
Solution for Zigtag

- Polysemy
  Different from Faviki, Zigtag could “understand” the orange is pointed for a fruit other than a color. System judges it by reading the context.
- Synonymy
  Zigtag allows one tag has different meanings. Users could choose them manually to define the details.

- Implementation
  Data structuring module uses Google Translate API to translate Chinese to English. Then it uses Zemanta API to do semantic tagging. The part of translation code is shown below:

```javascript
if (siteSearch.results && siteSearch.results.length > 0) {
    for (var i=0; i<siteSearch.results.length; i++) {
        resultArray[i]=siteSearch.results[i].content;
        alert(resultArray[i]);
        google.language.translate(resultArray[0], 'zh-CN', 'en', function(result) {
            if (result.translation) {
                results+=result.translation;
            }
        });
    }
}
```

Method suggest() gets the translated text and returns a ZemantaResult set. It
will firstly send some requests. Codes could be seen below:

```
ZemantaResult res = new ZemantaResult();
    try {
        /* Make the request */
        Document doc = request{
            RequestType.SUGGEST, URL,
            inst.getApiKey(), text,
            useFreebase);
        Element root = doc.getDocumentElement();

        And then the analyzed text will be recommended. Its result type can be varied. For example, it can be as the following article:

```
NodeList list = root.getElementsByTagName("articles");
if (list.getLength() == 1) {
    Element n = (Element)list.item(0);
    NodeList nl = n.getElementsByTagName("article");
    for (int i = 0; i < nl.getLength(); i++) {
        Element articleNode = (Element)nl.item(i);
        res.articles.add(new Article(
            getElementText(articleNode, "url"),
            getElementText(articleNode, "title"),
            getElementText(articleNode, "published_datetime"),
            getElementText(articleNode, "confidence"),
            getElementText(articleNode, "zemified")))
    }
}
```

It can also be key words as follows:

```
list = root.getElementsByTagName("keywords");
if (list.getLength() == 1) {
    Element n = (Element)list.item(0);
    NodeList nl = n.getElementsByTagName("keyword");
    for (int i = 0; i < nl.getLength(); i++) {
        Element articleNode = (Element)nl.item(i);
        res.keywords.add(new Keyword(
            getElementText(articleNode, "name"),
            getElementText(articleNode, "confidence")))
    }
}
```
4.2.3 Data Processing Module

Data processing module aims to process the structured data. The final target is to generate the useful data.

The process could be seen below:

We designed three databases to meet the requirements about “Data Processing Module”:

- **Address information database**: This database is used for storing structured data about company address.
- **Reputation database**: This database is used for storing structured data about company reputation.
- **Basic information database**: This database is used for storing structured database about basic information of company.

**Database Designing Principles**

1. **Table Design Principle**
   
   The standardization for data could eliminate the existence of data redundancy. There are several ways, but “Third Normal Form” (3NF) is regarded as the best balance among performance, expansibility and data integrity.

2. **Field Design Principle**

   It is necessary to choose the type of number and text. If the data is too long, it can not conduct calculating operation. And to the greatest extent use the data types in common use.

**Database Design**

After fully understand the system requirements, it is time to design database. Database structure design is core for database design; the quality of design results will results in the coding efficiency and the cost of maintenance. With a good, suitable design, the program code will be relatively simple, and the system will be more easy to maintain.
All databases are listed:
Below is the table structure of T_LOCATION_INFORMATION. This table is used to store structured information about company address and related information. The main attributes include: company ID, detail location, location type and so on.

Below is the table structure of T_REPUTATION_INFORMATION. This table is used to store structured information about company reputation and related information. The main attributes include: company id, reputation information, reputation type and so on.

Below is the table structure of T_BASIC_INFORMATION. This table is used to store structured information about company information and related information. The main attributes include: company id, awards, famous person id, exhibition id, mobile and so on.
4.2.4 Data Display Module

Data display module aims to display the processed data to end users. How to describe a SME is a very concerned question.

The author thinks that the dimension matching model could describe a SME. So the reputation, geographic information and type would be regarded as results to be displayed on browser. The draft could be seen below:
What end users need to do is just to search company's name in search box and click the search button. All results about this company information would be displayed:

**Geographic Information:** This will be displayed in the left part. We use Google Maps to display the geographic information. It will bring better user experience.

**Basic Information:** Basic information includes the company’s location, contract information.

**Reputation:** Reputation means a company's qualification and certification.

**Type:** Type is one of the most important attributes. It includes the scope of business.

We use HTML+CSS to arrange page layout. Detail codes could be seen below:
User could input the company's name in the search box. When he or she clicks the search button, the search results would display in the left-bottom part. At the same time, "Translated Text" part would show the English texts which is translated from Chinese. And suggest tag would be displayed in “Suggest Tag” part.
5 Testing Analysis

This chapter will test the prototype. First we illustrate the testing samples and test cases. Later, analysis of the testing results will be carried out.

5.1 Testing Samples and Test Cases

We have designed two test cases. They are:

(1) “Common Web Pages Search” test cases. This test cases use Google to search web contents through inputting the enterprise name.

(2) “Web Pages Search used by dimension matching model” test cases. These test cases adopt the dimension matching model to transfer unstructured data, semi-structured data to structured data and combines private structured data to describe the company.

The results of test case could be shown below:

<table>
<thead>
<tr>
<th>Company Name</th>
<th>Num. of results</th>
<th>Keyword</th>
</tr>
</thead>
</table>

5.2 Results Analysis

In this section we analyze the testing results. Firstly we will give the analysis of the overview results. After that, we analyze some very specific examples.

5.2.1 Overview

For “Common Web Pages Search” test cases, there is total fifty five company information, 83.3%.

Among these companies, the results could be divided into the following parts:
Useless Information: forty four companies. Here useless information means the searching results are useless.

Wrong Information: one company. Here there are two same-name companies on internet. And PR value of one company in Nan Tong city is higher than that in Kun Shan city.

Useful Information: ten companies. Here useful information means users could find related information about reputation, type and so on.

Below chart shows these data:

For “Web Pages Search used by dimension matching model” test cases, the results are shown below:

Useless Information: thirty-four companies. Here useless information means the searching results are useless.

Wrong Translation: one company.

Useful Information: twenty companies. Here useful information means users could find related information about reputation, type and so on.

Below chart shows these data:
Above all, we found if we adopt dimension matching model, the useful information we got could be increased. So the model is efficient.

5.2.2 Successful Case Example

This section uses “台灣豪力辉工业股份有限公司” as successful case example to conduct analysis.

“台灣豪力辉工业股份有限公司” is one of the most important manufacturers in Taiwan for Accessories of Machine Center, Turning Center, Milling Machine, Grinding Machine with excellent technical productivity and more than 25 years experience in the world market. Their latest major products include: Angle Head (Angular Head, Angle Adjustable Angle Head, Aluninum Angle Head, Heavy Duty Angle Head), VDI / BMT Driven Tool Holder (Axial, Radial, Offset, Tapping, Angle Adjustable, Radial Swivel Gear Hobber) with different system, Hydraulic Vise, MC Power Vise, and so on. [27]

The search result page is shown below:

<table>
<thead>
<tr>
<th>Company Name</th>
<th>Result set No.</th>
<th>Keywords</th>
</tr>
</thead>
<tbody>
<tr>
<td>台湾豪力辉工业股份有限公司</td>
<td>20800</td>
<td>Taiwan, Machine tool, China, Milling machine, Machinery, Business, Numerical control, Industrial Goods and Services</td>
</tr>
</tbody>
</table>

Then we use our matching model to get the results:

From the results, we could find the following keywords:

- **Location**: Taiwan, China
- **Scope of business**: Machine tool, Milling machine, Numerical Control
If we could add additional structured data which is stored in our private database, such as contact information, reputation, contact address. It can be found that it could describe this enterprise more accurately.

The final result from the matching model is shown below:

Here we want to mention that ISO 9001 doesn't appear in Internet. This information is stored in our private database. And this information is very helpful for end users to learn about this enterprise.

5.2.3 Failure Case Example

This section uses “锦鸿塑胶五金模具配件有限公司” as failure case example to analysis.

“锦鸿塑胶五金模具配件有限公司” is a company which produces all kinds of pattern such as centre, sleeve, guide pillar and so on.

The search result page could be seen below:
From the results, we could extract the following information:

- **Location**: Dongguan, China
- **Scope**: Business
- **Unrelated Information**: Microsoft, Telephone

We can not learn about the enterprise type, scope. At the same time, the results include a number of unrelated information. The acceptable matching model is shown below:
We could find the expected result is different with real result. So we think it is a failure matching.

After analyzing the failure reason, the author summarizes the following factors:

(1). Source data exists “dirty” data. Because we extracted information from websites using Google, it exists huge “dirty” data. These “dirty” data affects the matching results.

(2). Enterprise type is not accurate.
- Enterprise information is incomplete on Internet. For 66 testing examples, there are some companies which we could not find the company information. This shows some enterprises do not realize the importance of E-business to their companies.

5.3 Problems and Improving Solution

According to the testing analysis, we could find there are following problems:

(1) Precision could be improved
   During the testing process, we found if we could improve the precision, the matching results could be better.

(2) Cannot recognize error information
   Error information here refers to the “dirty” data – These data is not useful for describing SMEs. It would affect the final results.
6 Summary

With a growing number of SMEs begin to know or use the Internet, more and more information about SMEs will be appeared on the Internet. This may cause the Internet has different source of data which on the same business entity. The Internet self is a highly distributed system, enterprises can release their information (name, address, phone, scope of business and so on) on many different websites and these information can be different as well. However, on the essence, these information have a certain degree of correlation.

Generally, we cannot visually extract and filter the small enterprise information, thus establish the intrinsic link between information. However, the Internet is not only a place for information releasing, it is a set of computing power and service capacity combination as well. People can use some web services provided by companies on the Internet (mainly in the enterprises’ opening API) to extract and filter the data on the Internet so as to integrate correlation information and to provide expected results to users.

The goal of this thesis is to set a service-oriented matching model based on the study of existing enterprise matching models and data format.

This thesis establishes a new service-oriented matching model that could better describe the information about small and medium sized enterprises from the field of electronic commerce. Based on the matching model, this thesis develops the prototype, verifies and evaluates the matching model that has been constructed.
References

[14] Börkur Sigurbjörnsson, Roelof van Zwol, "Flickr Tag Recommendation based on Collective Knowledge", the


[23] Semantic Web Use Cases and Case Studies,
http://www.w3.org/2001/sw/sweo/public/UseCases/Faviki [z] 2008.11.8

[24] Google AJAX Search API,

[25] Google AJAX Language API,

[26] Zemantan API,
http://developer.zemanta.com [z] 2009.3.27

[27] HoldWell