The Hong Kong Polytechnic University

Department of Computing

Relation Extraction for Ontology Extension Using Integrated Evidences

CHEN Yirong

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

August, 2010

CERTIFICATE OF ORIGINALITY

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

___________________________________(Signed)

__________CHEN Yirong________________(Name of student)

Abstract

Ontology is a valuable resource for many domain specific applications where domain knowledge is needed. With the rapid development in science and technology, new terminology and associated concepts must also be updated in the ontology to suit for real time applications. Current methods of manual construction of ontology is too time consuming and difficult to update. Thus automatic extension of ontology is especially needed.

In this study, investigation to terminology extraction is first carried out. In addition to unit-hood measurement, this work further studies how to take domain specific knowledge to further measure term-hood to improved terminology extraction algorithms. After a thorough review of existing ontology resources, this study further investigates how to map the extracted terminology to a domain specific ontology. Given a core ontology, the key issue is how to find the relationships of the new terms to the concepts of the ontology. The investigation focuses on the extraction of *kind-of* relations. The work is divided into three steps: (1) To design effective algorithms to extract terms from domain corpus with good accuracy; (2) To investigate effective techniques to extract relations between concepts especially kind-of relations; (3) To link obtained ontology to upper ontology.

The contributions of this work are three folds: (1) An effective term extraction algorithm is proposed based on the measures of both linguistic unit and domain specificity; (2) Algorithm of relation extraction is designed to construct domain ontology using multiple evidences; and (3) the construction and mapping of a core ontology to the upper ontology to ensure interoperability with other domain ontologies.

Publications Arising From the Thesis

Papers as the main contributor

[1] **Yirong Chen**, Qin Lu, Wenjie Li, Zhifang Sui, Luning Ji. 2006. A Study on Terminology Extraction based on Classified Corpora. In Proceedings of the International Conference on Language Resources and Evaluation (LREC 2006), pp. 2383-2386, Genoa, Italy, May 24-26, 2006.

[2] **Yirong Chen**, Qin Lu, Wenjie Li, Gaoying Cui. 2007. Discovering Kind-of Relation for Ontology Construction. In Chinese Lexical Semantics Workshop (CLSW2007), Hong Kong, May 21-23, 2007.

[3] **Yirong Chen**, Qin Lu, Wenjie Li, Wanyin Li, Luning Ji, Gaoying Cui. 2007. Automatic Construction of a Chinese Core Ontology from an English-Chinese Term Bank. in Proceeding of ISWC2007 Workshop OntoLex07 - From Text to Knowledge: The Lexicon/Ontology Interface, pp. 78-87, Busan, Korea, November 11, 2007.

[4] **Yirong Chen**, Qin Lu, Wenjie Li, Wanyin Li, Gaoying Cui. 2008. Chinese Core Ontology Construction from a Bilingual Term Bank. In the Sixth International Conference on Language Resources and Evaluation (LREC 2008), pp. 2344-2351 Marrakech, Morocco, May 28-30, 2008.

[5] **Yirong Chen**, Qin Lu, Wenjie Li, Gaoying Cui. 2009. A Novel Method to Improve Chinese Core Ontology Construction with Terms Sharing Suffixes. In the proceedings of the 10th Chinese National Conference on Computational Linguistics (CNCCL 2009), pp. 370-375, Yantai, China, July 24-26, 2009.

[6] **Yirong Chen**, Qin Lu, Wenjie Li, Gaoying Cui. 2010. A Novel Method for Chinese Core Ontology Construction, In Journal of Chinese Information Processing, vol. 24, no. 1, pp. 48-53.

Papers as a minor contributor

[7] Gaoying Cui, Qin Lu, Wenjie Li, and **Yirong Chen**. 2007. Attributes Selection in Ontology Taxonomies Acquisition with FCA. In proceedings of Chinese Lexical Semantics Workshop (CLSW2007), Hong Kong, May 21-23, 2007.

[8] Gaoying Cui, Qin Lu, Wenjie Li, and **Yirong Chen**. 2008. Corpus Exploitation from Wikipedia for Ontology Construction. In proceedings of the International Conference on Language Resources and Evaluation (LREC 2008), pp. 2125-2132, Marrakech, Morocco, May 28-30, 2008.

[9] Gaoying Cui, Qin Lu, Wenjie Li, and **Yirong Chen**. 2008. Attributes Selection in Chinese Ontology Acquisition with FCA. In International Journal of Computer Processing of Oriental Languages (IJCPOL), vol. 21, no 1, pages 77-95.

[10] Gaoying Cui, Qin Lu, Wenjie Li, and **Yirong Chen**. 2009. Automatic Acquisition of Attributes for Ontology Construction. In the proceedings of the 22nd International Conference on the Computer Processing of Oriental Languages (ICCPOL2009), pp. 248-259, Hong Kong, March 26-27, 2009..

[11] Luning Ji, Mantai Sum, Qin Lu, Wenjie Li, **Yirong Chen**. 2007. Chinese Terminology Extraction using Window-Based Contextual Information. In the proceedings of the Conference on Intelligent Text Processing and Computational Linguistics (CICLing), pp. 62-74, Mexico city, Mexico, February 18-24, 2007.

[12] Luning Ji, Qin Lu, Wenjie Li, **Yirong Chen**. 2007. Automatic Construction of a Core Lexicon for Specific Domain. In the proceedings of the Sixth International Conference of Advanced Language Processing and Web Information Technology (ALPIT 2007), pp 183-188, Luoyang, China, August 22-27, 2007.

[13] Gaoying Cui, Qin Lu, Wenjie Li, and **Yirong Chen**. 2009. Mining Concepts from Wikipedia for Ontology Construction. In the workshop NLPOE for 2009 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Milan, Italy, Sep.15-18, 2009.

Acknowledgments

This thesis could not have been done without the help and cooperation of many people, and it is my great pleasure to take this opportunity to thank them.

First and foremost, I would like to express my deepest gratitude to my chief supervisor Prof. Qin Lu for being a consistent source of support and encouragement. Without her knowledge and perceptiveness, I would never have finished my Ph.D. study. I gratefully acknowledge her for her instruction, help and encouragement. I would also like to thank my co-supervisor Dr. Wenjie Li for her care, advice and cheers.

I would like to express my deep gratitude to Dr. Ruifeng Xu and Ms. Liao Li for giving me continuous support and kind help. I appreciate very much the research discussions and sharing of ideas with Ms. Gaoying Cui who works with me in the same research area. Thanks a lot to them and many others. Your company and help made my four years of life in Hong Kong a pleasant experience.

At last, I would like express my deepest appreciation to my father Zhikun Chen and my mother Maoxiu Tao, for their endless love and unwavering support. Although they cannot read it, this thesis is dedicated to them.

Table of Contents

Abstract i

Publications Arising From the Thesis ii

Acknowledgments iv

Table of Contents v

List of Figures viii

List of Tables x

Chapter 1 Introduction 1

1.1 Background and Motivation 1

1.2 Objectives and Scopes 3

1.3 Methodologies 4

1.4 Contributions 6

1.5 Organization 6

Chapter 2 Literature Review 7

2.1 Basic Concepts and Terms 7

2.2 Overview of Ontologies 11

2.2.1 Ontology Classification and Cases 12

2.2.2 Ontology Engineering Processes 17

2.2.3 Ontology Learning from Text 18

2.3 Review of Terminology Extraction 19

2.3.1 Unithood 19

2.3.2 Termhood 20

2.4 Review of Kind-of Relation Extraction 21

2.5 Review of Core Ontology Construction 24

Chapter 3 Terminology Extraction using Unithood and Termhood Measures 27

3.1 Unithood 27

3.1.1 Definitions 28

3.1.2 Baseline methods 29

3.1.3 Multiple-Features Based Unithood Algorithm 31

3.1.4 Performance Evaluation 33
3.1.5 Conclusion 40
3.2 Termhood 41
3.2.1 Algorithm Description 41
3.2.2 Preparation for Termhood Evaluation 43
3.2.3 Evaluation 44
3.2.4 Conclusion 46
Chapter 4 Kind-Of Relation Extraction using FCA and Internal Components 48
4.1 Formal Concept Analysis using Contextual Collocation 48
4.1.1 Design of Selection Algorithms 49
4.1.2 Evaluations 51
4.1.3 Conclusion 56
4.2 Observations of Suffix Component Based Kind-Of Relation Extraction 56
4.2.1 The evaluation of coverage of suffix components 57
4.2.2 The evaluation of precision of suffix components as hypernyms 58
4.2.3 Conclusion 59
Chapter 5 Chinese Core Ontology Construction 60
5.1 Acquisition of Core Terms 60
5.2 Design Principles of COCA 62
5.3 Statistical Translation Module 63
5.4 Sense Disambiguation Module 64
5.5 Concept Selection Module 65
5.5.1 Union Probability of Independent Events 66
5.5.2 Multiple English Translation Feature 66
5.5.3 Hyponym Feature 67
5.5.4 Feature of Terms Sharing Suffixes 68
5.5.5 Part-of-Speech Feature 70
5.5.6 Feature Integration for Concept Selection 71
5.6 Evaluation 71
5.7 Conclusion 74

Chapter 6 Improving Chinese Core Ontology Construction with Bilingual WordNet 76

6.1 Direct Use of Sinica BOW 76

6.1.1 Algorithm Design 77

6.1.2 Preprocessing of SBOW 78

6.1.3 Performance Evaluation of DMM and MATB 79

6.1.4 Analysis of Matching Types 80

6.2 Improved COCA Using SBOW 83

6.2.1 Design of the Improved COCA 84

6.2.2 Performance of the Improved COCA with SBOW 85

6.3 Conclusion 87

Chapter 7 Conclusions and Future Work 88

Bibliography 91

Appendix 1 The 49 Chinese Terms used in Evaluation of FCA 102

Appendix 2 Mappings of 1500 Chinese IT Core Terms to Synsets and SUMO 103

Appendix 3 Internet Lexicon Ontology with IT Core Ontology by COCA 233

List of Figures

Figure 2.1 Three Level Model of Ontology 12

Figure 2.2 Demonstrations of WordNet Relations 17

Figure 3.3 Precision in Character-based Bigram Unithood Computation 34

Figure 3.4 RCP of Character-based Bigram Unithood Measures 36

Figure 3.5 Precision of Character-based Trigram Unithood Measures 36

Figure 3.6 *RCP* in Character-based Trigram Unithood Computation 37

Figure 3.7 Precision on Character-based 4-gram Unithood Measures 38

Figure 3.8 *RCP* in Character-based 4-gram Unithood Measures 38

Figure 3.9 Precision of *UREval* on Word Based Extraction against Top N Candidate 40

Figure 3.10 Precisions using noisy term set 45

Figure 3.11 Precisions using correct unit set 45

Figure 4.1 F-measure on Three Attribute Types 54

Figure 4.2 *INT* Performance 54

Figure 4.3 *DNT* Performance 55

Figure 4.4 *DET* Performance 55

Figure 4.5 Coverage of Suffix Components 58

Figure 5.1 the Framework of COCA 63

Figure 5.2 Multiple English Translation Feature 66

Figure 5.3 Using Hyponym in Domain 68

Figure 5.4 Using Part-of-Speech 70

Figure 5.5 Performances in Bar Chart for Different Combination of Features 73

Figure 5.6 A Fragment of Automatically Constructed Chinese Core Ontology 74

Figure 6.1 Four Types of Matching cases between *SBOW* and 400 answers 81

Figure 6.1 Framework o Improving *COCA* with Bilingual WordNet.................................... 84

List of Tables

Table 3.1 Percentage improvement of *UREval* in different gram unit computation............. 39

Table 4.2 Performance of Different Object-attribute Features 52

Table 6.1 Precision of *DMM*, *MATB* and *COCA* 80

Table 6.1 Four Types of Matching cases between *SBOW* and 400 answers.......................... 81

Table 6.2 *NoMatch* Cases in Matching between *SBOW* and 400 answers 82

Table 6.3 "*Partially Matched Without Correct Answer"* Cases in Matching........................ 83

Table 6.4 Notations of Configurable Parameters.. 86

Table 6.5 Three Groups of Evaluation Result on Three Parameters Respectively 86

Table 6.6 Ratio of Matching Type against the Length.. 87

Chapter 1

Introduction

1.1 Background and Motivation

With the abundance of information available over the Internet, there is an increasing need for building and updating domain knowledge base quickly so as to facilitate applications in areas such as information retrieval, information extraction, machine translation, summarization and question-answering systems.

Terms, as the carriers of domain concepts, are the most fundamental units to represent and encapsulate concepts in any specific domain. However terms alone are not enough to represent domain knowledge. The concepts behind terms and the relationship among different concepts form the essential structured knowledge base, the so called **domain ontology** [Navigli, 2004], which contains all domain relevant terms, concepts, mappings from terms to concepts and relations between concepts. The discovery of new terms is useful only if it can be used to extend the knowledge of a domain. This requires the finding of the association of these new terms with existing concepts and mapping these new terms into existing domain ontology.

Traditional terminology extraction and ontology construction are conducted by domain experts manually which are difficult to process and time-consuming. The availability of large scale online text and the fact that they are being updated constantly require fast and efficient automatic terminology extraction and ontology construction methods which **can** not only process millions of words in minutes, but also provide better coverage, consensus and accessibility.

Term [Sui, 2002] is the lexical unit to represent domain knowledge or concepts. Automatic term extraction measures the suitability of a string in two aspects, namely, unithood and termhood. **Unithood** [Kageura, 1996] measures the suitability of a string candidate being an independent linguistic unit so as to serve as a term. Many research works have been done on it. However most of the methods are sensitive to frequency and the performance still needs improvement. **Termhood** [Kageura, 1996] measures the domain specificity of a string. Comparing to unithood, less works are reported on termhood measure.

To create ontology from terms, various relations should be extracted. A critical relation between concepts is the kind-of relation which can be used to form a concept hierarchy for different concepts. Based on such a concept hierarchy, axioms and reasoning systems can be used. Although numerous researchers [Navigli, 2004; Hearst, 1992; Cimiano, 2004; Caraballo, 2001; Li, 2005; Hindle, 1990; Bisson, 2000] have been working on this area, the reported performances of the kind-of relation extraction are still far from satisfactory.

Ontology can be divided into three levels [Navigli, 2004]: (1) upper ontology, (2) mid-level ontology, and (3) specific domain ontology. An **upper ontology** is a general ontology to ensure commonality and reusability across different domains. A **domain ontology** is an ontology to conceptualize a specific domain. A **mid-level ontology** is a bridge ontology to help automatically mapping between some specific domain ontology to an upper ontology. In other words, it captures the concepts which are considered to be the most fundamental to a domain and can be directly mapped from upper ontology.

In addition to extractions of certain types of relations for ontology construction, domain terms should be mapped to some upper ontology so as to form a unified conceptualization of the world because an upper ontology describes general concepts that are the same across all domains. Without such mapping, domain specific ontology is isolated knowledge bases without commonalities being recognized.

It is not easy to map a domain ontology to an upper level ontology directly because the diversified lexical expressions make it difficult to find commonality between them. From ontology engineering perspective, it makes more sense to first construct a mid-level ontology (namely core ontology) to model the fundamental domain knowledge so that it can bridge the gap between an upper ontology and a domain ontology. Core ontology construction should start by identifying **core terms** [Ji, 2007a] in a domain which identifies the fundamental concepts. Identifying domain fundamental concept is a process of disambiguation based on domain information. The process can also discover some synonym relations among domain terms.

1.2 Objectives and Scopes

The main objective of this work is to explore methods to construct domain ontology automatically or semi-automatically. In order for ontology construction methods to be feasible for different domains, this work focuses on the investigation of three issues: (1) to design effective algorithms to extract terms from domain corpus with good accuracy, (2) to investigate effective techniques to extract relations between concepts especially kind-of relations, and (3) to link obtained ontology to upper ontology.

This study focuses on Chinese ontology construction. However the methods developed should not be language dependent except in the preprocessing part. The acquisition of relation information is done through the use of a number of resources including Chinese domain corpora, high-quality English ontological data such as Wikipedia, WordNet [Fellbaum, 1998], an upper ontology such as SUMO, and other available internet resources. The use of Chinese domain corpora is for domain term extraction. Relation extraction explores evidences from sources of all kinds. Multiple evidences are investigated including internal evidences such as component information of terms and external evidences such as the contextual collocations. Relation extraction is mainly focused on the kind-of relation. Then, the extracted relations and evidences are integrated into a decision system to construct the ontology using automatic or semi-automatic method.

1.3 Methodologies

To reach the objectives listed in previous section, our fundamental idea is to integrate all obtained evidences in both term extraction and relation extraction to construct ontology. With reference to the objectives given in the previous section, detailed methodology used for each objective is elaborated below.

(1) To design effective algorithms to extract terms from domain corpus with good accuracy − In terminology extraction, investigations are focused on both stability of term candidates in unithood measures and domain specificity in termhood measure. Stability deals with the tightness of characters used as a string and the independence of the whole string. So stability is measured both in terms of boundary information as well as inner relations of substrings within a term candidate. In addition to unithood, this study also investigates methods to measure domain specificity which is not widely studied so far. In this study, classified corpora are used to provide both domain specific information as well as information common to different domains which can help provides cues to the nature of data and can be used in termhood calculation. Three algorithms are designed and evaluated to investigate termhood based on classified corpora. Design considerations are based on lexicon set, term frequency, document frequency, and strength of relations between a term and its document class, respectively.

(2) To investigate effective techniques to extract relations between concepts especially kind-of relations - To create ontology, various relations should be extracted to form concept hierarchies and other structures. Among all relations, kind-of relations is the most vital one to the construction of concept hierarchies. In this work, a corpus-based approached is used to identify kind-of relations. The mathematic model of **F**ormal **C**oncept **A**nalysis (FCA) which is effective in identifying partial ordering relations is used to mine kind-of relations. The method selects stable attributes for each concept in sentence context from a large scale domain corpus. Then based on FCA, kind-of relation between two concepts can be induced if there is a sub-set relation between their attribute sets respectively. In selecting attributes from a corpus, previous works were short on providing stable, frequent and accurate pairs. In our research, attributes are taken from more kinds of words besides verbs, and the position information and path information after parsing between the attribute and the concept are also adopted to help identify potential attributes.

(3) To link obtained ontology to upper ontology − In order to obtain a unified conceptualization, linking previously extracted domain concepts to upper ontology becomes necessary. However, directly linking the extracted relations from a domain corpus can be difficult because they are extracted from a corpus which is unlikely to have good coverage and comprehension. As core ontology contains all the fundamental concepts in a domain, it is much easier to map concepts and relations identified in (2) to the concepts in a core ontology than to map them indirectly to an upper ontology. In this research, our approach is to first obtain a core ontology (namely, mid-level ontology) using some domain lexicon as the main domain resource. Then, the core ontology is mapped to an upper ontology. Due to the lack of ontological resources in Chinese, this work takes the bilingual approach by using available English resources to assist in the mapping of Chinese core ontology to SUMO [Niles, 2001], the upper ontology for English. The resources used include bilingual lexicon, and other English language resources such as WordNet, and Wikipedia. A graph based **C**ore **O**ntology **C**onstruction **A**lgorithm (COCA) [Chen, 2008b] is designed with consideration of different resources and features. The features include multiple translation feature between Chinese core terms and WordNet, extended string feature and Part-of-Speech feature. Additional resources such as bilingual WordNet and Wikipedia are integrated into COCA at the end to improve performance.

1.4 Contributions

The contributions of this study are listed as follows. (1) An effective term extraction algorithm is proposed based on the measures of both linguistic unit and domain specificity. (2) Algorithm of relation extraction is designed to construct domain ontology using multiple evidences. (3) The proposed method maps the created ontology to a standard upper ontology through the construction and mapping of a core ontology to the upper ontology to ensure interoperability with other ontologies.

1.5 Organization

The thesis is organized as follows. Chapter 2 gives details of literature review. Chapter 3 presents terminology extraction based on statistical measures for unithood and termhood. Chapter 4 describes kind-of relation extraction algorithm. Chapter 5 proposes the algorithm of Chinese core ontology construction from bilingual term bank. Chapter 6 further investigate bilingual resources, WordNet and Wikipedia, respectively to help the construction of Chinese core ontology. Chapter 7 concludes this study.

Chapter 2

Literature Review

The main objective of this work is to explore methods to construct domain ontology automatically or semi-automatically. Aiming to make clear about the domain ontology construction and its context, basic concepts, terms and an overview of ontology is given first. Then the related works are further elaborated in three issues of ontology learning from text: terminology extraction, kind-of relation extraction and core ontology construction.

2.1 Basic Concepts and Terms

The definitions of several basic concepts such as concepts, terms, relations, and ontology are given below.

Concept [WordNet, 2007]:

A concept is an abstract or general idea inferred or derived from specific instances. It is similar to words "Class", "Category" and "Type". For example, "computer" is a concept while "my notebook" is not because "my notebook" is an instance existed in specified time and space. More specifically, concepts are characterized by three parts [Buitelaar, 2005], namely

1). A set of linguistic realization, i.e. (multilingual) terms for this concept.

2) An intentional definition of the concept.

3) A set of concept instances, i.e. its extension

Term [WordNet, 2007]:

Term is a word or expression used for some particular thing. Terms serves as the lexical representation of concepts. For example, the word "Computer" is the term representing concept computer. However, each term may be associated with a number of concepts and thus, disambiguation is needed to identify the specific concept that a term represents in a particular context.

Relation:

Relation[1] is a logical or natural association between two or more entities; relevance of one to another. For example, “notebook” in IT domain is a kind of “computer”. Thus the relation between them is kind-of relation. In mathematics, an *n*-place relation[2] is defined on a Cartesian product of n sets, and is represented by a set of ordered *n*-tuples. Because binary relations are used very often, the term relation without reference to its order commonly refers to a binary relation. A relation can have associated properties, relation type, and attributes. A relation can either be between concepts or between a concept and its instance. In this research a relation generally refers to the former without qualification.

Kind-of Relation:

If A is a kind of B, then <A, B> is an element of kind-of relation. Kind-of relation is often referred to as the “*is-a”* relation. It is also a kind of *hypernym* relation. Once <A, B> is a kind-of relation, then A usually is replaceable by B in its context.

Synonym Relation / Synonym / Synset :

Synonymy relation is the semantic relation that holds between two terms that can (in a given context) express the same meaning. Two words *A* and *B* can be interchanged in a context, then *A* is the synonym of *B* and *B* is the synonym of *A*. For example, the word ”disk” and ”disc” can be interchanged in a context. Consequently, the word ”disk” are synonym of ”disc” and vice versa. They have sysnonym relations. The two words together having the same meaning can form a synset. Synset is a set of one or more synonyms defined in WordNet [WordNet, 2007], a lexical ontology to indicate relatios of lexical terms . As a

[1] From The American Heritage® Dictionary
[2] From PRIME mathematics encyclopedia

term represents a different concept can appear in a different synset, each synset has a definition sentence to serve as descriptive data.

Originally Ontology is a philosophy concept. It is later introduced into the domain of information technology and has its meaning changed. The following two definitions clarify the two meanings of “ontology”.

Ontology:

In philosophy, the uncountable word **O**ntology (with capital “O”) refers to the metaphysical study of the nature of being and existence [WordNet, 2007].

ontology:

In computer science, the countable word **o**ntology (with lower case “o”) refers to a **formal**, **explicit specification** of a **shared conceptualization** [Studer, 1998]. In the rest of the thesis, ontology only denotes the computer science ontology if not specified.

This definition is developed based on the most cited definition of Gruber (ontology is “an explicit specification of conceptualization”) [Gruber, 1993] and Borst’s refined version (ontology is “a formal specification of shared conceptualization”) [Borst, 1997].

In the definition, “conceptualization” can be understood as an abstract model using concepts and relations of concepts to represent a piece of reality and communicate meanings among agents [Chen, 2008a]. “Shared conceptualization” requires a common understanding between agents. As a basic communication tools, natural language can be understood between agents and can help to form a shared conceptualization. “Formal, explicit specification” requires the ontology to be machine readable and explicitly stated. To satisfy the requirement, disambiguation is widely introduced in the mapping between natural language and ontology since natural language is ambiguous.

In practice, a formal definition of ontology structure is:

$$O := (C, \leq c, R, \sigma R, \leq_R)$$

where there are two disjoint sets C and R whose elements are called *C*oncept identifiers, and *R*elation identifiers. The upper semi-lattice [Hazewinkel, 2002] $\leq_C$ on C with a top element $root_C$, is called the **concept hierarchy** or **taxonomy** where any two concepts in set C must have the least upper bound. Namely, any two concepts in set C must share at least one same upper concept. The function $\sigma_R : R \rightarrow C^+$ is called the relation signature and means that a relation is between one, two or more concepts. The partial order $\leq_R$ on R is called the **relation hierarchy**. For two relations r_1 and r_2, r1 $\leq_R$ r_2 implies that $|\sigma_R(r_1)| = |\sigma_R(r_2)|$ and $\pi_i(\sigma_R(r_1)) \leq_R \pi_i(\sigma_R(r_2))$, for each $1 \leq i \leq |\sigma_R(r_1)|$ where function $\pi_i(t)$ returns the *i-th* components of tuple *t*.

A simple ontology example is demonstrated as following. An ontology *MINI_COMPUTER_ONOTLOGY* has the concepts (C) including computer, notebook and disk; relation types (R) including kind-of relation and part-of relation; concept hierarchy $\leq_C$ including the hierarchy root “computer” and the leaf node “notebook”; relation signature σ_R including the kind-of relation between computer and notebook, and the part-of relation between computer and disk.

The formal definition above of ontology is modified from the one of Cimiano [2006] with attribute related components deleted. Attribute can be considered as a special type of relation which is named and is the relation between two concepts. So the deletion of attribute related components is reasonable which makes the formal definition more concise. Consequently, the coming definition of “Lexicon Ontology is also with attribute related component deleted.

Lexicon Ontology (modified from [Cimiano, 2006]):

A *lexicon ontology* is a pair *(O, Lex)* where O is an ontology and *Lex* is a lexicon for O. The lexicon *Lex* for the ontology $O := (C, \leq_C, R, \sigma_R, \leq_R)$ is the structure:

$$Lex := (S_C, S_R, Ref_C, Ref_R)$$

which consists of sets S_C and S_R whose elements are called signs (commonly are terms) for concepts and relations respectively; a relation $Ref_C \subseteq S_C \times C$ called lexical reference for concepts; a relation $Ref_R \subseteq S_R \times R$ called lexical reference for relations. Hereby the Cartesian product [Hazewinkel, 2002] operator × constructs a new set $X \times Y$ which consists of all ordered pair $<x, y>$ with x belonging to the set X and y belonging to the set Y. Against the definition of Cartesian product, it can be seen that the above expressions for Ref_C and Ref_R means that concept or relation can has zero, one or more terms, and vice versa.

The example provided for ontology is further used here. A lexicon *COMPUTER_LEX* will be provided for ontology *MINI_COMPUTER_ONOTLOGY* and finally form the lexicon ontology *(COMPUTER_LEX, MINI_COMPUTER_ONOTLOGY)*. The lexicon *COMPUTER_LEX* has concept word set S_C with words "disk", "disc", "notebook" and "computer"; the relation set Ref_C with mappings between concept words and concepts. For instance, both the word "disk" and "disc" denote the concept "disk, disc, magnetic disk" in relation set Ref_C. The other two elements R and Ref_R in lexicon are very similar to C and Ref_C.

A typical lexicon ontology is WordNet which is further introduced in later chapters.

2.2 Overview of Ontologies

In the previous section, definitions and formal representation of ontology are given. In this section, an overview of ontology classification, ontology engineering processes and ontology learning is discussed. Ontology languages (such as RDF [Klyne, 2004], OWL [McGuinness, 2004] in XML[Bray, 2000] format and KIF [Genesereth, 1992]) and representations are not included because our leaned ontology is a light-weight lexicon ontology which includes only terms, concepts, relation types, relations between concepts, relations between concepts and terms. The acquired ontology can be easily stored in database or any other formats and standards.

2.2.1 Ontology Classification and Cases

To do ontology extension, what kind of ontology resources is available is very important. To better use the existing ontology, an ontology classification model discussed below is widely used.

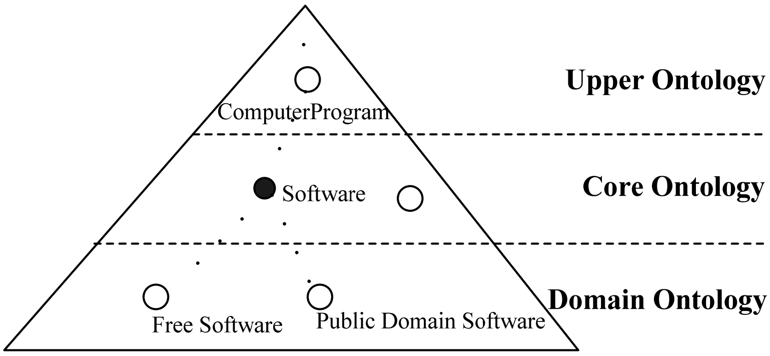


Figure 2.1 Three Level Model of Ontology

As shown in the above figure, Ontology is often divided into several levels: domain ontology and upper ontology; sometimes Mid-Level Ontology (MILO) is used.

An **upper ontology [**Ontology, 2007; Navigli, 2004**]** (or foundation ontology) models the common objects that are generally applicable across a wide range of domain ontology. It contains a core glossary in whose terms objects in a set of domains can be described.

A **domain ontology [**Ontology, 2007**]** (or domain-specific ontology) models a specific domain, or part of the world. It represents the particular meanings of terms as they apply to that domain. For example the word "card", has many different meanings. It can mean the card used in games in general domain as well as circuit board in electronics domain.

A **mid-level ontology** [Semy, 2004; Navigli, 2004], namely core ontology, serves as a bridge between abstract concepts defined in the upper ontology and the low-level domain specific concepts specified in the domain ontology. Mid-level ontologies may provide the concepts which are more concrete than those defined in upper ontologies and more abstract than those defined in domain ontologies, so as to make the mapping of concepts across domains easier.

In the following cases, an upper ontology and a widely used general domain lexicon ontology is introduced.

2.2.1.1 Suggested Upper Merged Ontology

SUMO (Suggested Upper Merged Ontology) [SUMO, 2002; Pease, 2002] is an upper ontology which provides definitions for general-purpose terms and acts as foundation for more specific domain ontologies [Niles, 2001]. It is created as part of the IEEE (Institute of Electrical and Electronics Engineers) Standard Upper Ontology Working Group. The goal of this working group is to develop a standard upper ontology which promotes data interoperability, information search and retrieval, automated inference, and natural language processing.

The structure of SUMO consists of concepts, instances, relations and axioms. SUMO is an ideal upper ontology for domain ontology construction with the following features:

1) Easy access. SUMO is free.
2) Authority. SUMO is owned by the IEEE[3].
3) Applicability. SUMO is one of the few formal ontologies that have been mapped to WordNet [Miller, 1993; Niles, 2003] lexicon. All synsets in WordNet were mapped to SUMO.
4) Multi-language support. It provides language generation templates for Hindi, Chinese, Italian, German, Czech and English.
5) Professional tool support. SUMO has good tool support for browsing and editing. An ontology visual editing and displaying system is available which is particularly fine tuned for SUMO.
6) Large scale. SUMO is the largest free, formal ontology available, with 20,000 terms and 60,000 axioms when all domain ontologies are combined. These consist of SUMO itself (the official latest version on the IEEE web site is here), the Mid-Level

[3] http://www.ontologyportal.org/

Ontology (MILO), and ontologies of Communications, Countries and Regions, distributed computing, Economy, Finance, engineering components, Geography, Government, Military, North American Industrial Classification System, People, physical elements, Transnational issues, Transportation, Viruses, World Airports A-K, World Airports L-Z. Additional ontologies of terrorism and weapons of mass destruction are available on request.

7) Axiom support. SUMO is richly axiomatized, not just taxonomy. All terms are formally defined. Meanings are not dependent on a particular inference implementation. An inference and ontology management system is provided.

2.2.1.2 WordNet: A General Domain Lexicon Ontology

WordNet [Miller, 1993] is an online lexical reference system whose design is inspired by current psycholinguistic theories of human lexical memory. English nouns, verbs, adjectives and adverbs are organized into synonym (equivalent word) sets, each representing one underlying lexical concept. Different relations link the synonym sets.

WordNet was developed by the Cognitive Science Laboratory at Princeton University under the direction of Professor George A. Miller (Principal Investigator).

The WordNet contains

Nouns: organized as topic hierarchies

Verbs: with entailment relations

Adjective: in N-dimensional hyperspace

Adverb: in N-dimensional hyperspace

Function words: probably stored separately as part of the syntactic component of language.

As of 2005, the database contains about 150,000 words organized in over 115,000 synsets for a total of 203,000 word-sense pairs; in compressed form, it is about 12 megabytes in size.

WordNet distinguishes between nouns, verbs, adjectives and adverbs because they follow different grammatical rules. Every synset contains a group of synonymous words or collocations (a collocation is a sequence of words that go together to form a specific meaning, such as "car pool"); different senses of a word are in different synsets. The meaning of the synsets is further clarified with short defining glosses. A typical example synset with gloss is:

Good, right, ripe -- (most suitable or right for a particular purpose; "a good time to plant tomatoes"; "the right time to act"; "the time is ripe for great sociological changes")

Most synsets are connected to other synsets via a number of semantic relations. For example, antonymy and synonymy are two common relations in WordNet.

These relations vary based on the type of word, and include:

Nouns

Hypernym: Y is a hypernym of X if every X is a (kind of) Y

e.g., edible fruit is the hypernym of apple.

Hyponym: Y is a hyponym of X if every Y is a (kind of) X

e.g., apple is the hyponym of edible fruit.

Coordinate terms: Y is a coordinate term of X if X and Y share a hypernym

e.g., pear is the coordinate term of apple because they are all edible fruits.

Holonym: Y is a holonym of X if X is a part of Y

e.g., car is a holonym of automobile engine

Meronym: Y is a meronym of X if Y is a part of X

e.g.: automobile engine is the meronym of car.

Verbs

Hypernym: the verb Y is a hypernym of the verb X if the activity X is a (kind of) Y (e.g., movement is the hypernym of travel)

Troponym: the verb Y is a troponym of the verb X if the activity Y is doing X in some manner (e.g., lisp is the troponym of talk)

Entailment: the verb Y is entailed by X if by doing X you must be doing Y (e.g., snoring is entailed by sleeping)

Coordinate terms: those verbs sharing a common hypernym

Adjectives

Related nouns (e.g., beauty is a related noun of "beautiful")

Participle of verb (e.g., exciting)

Adverbs

Root adjectives (e.g., smooth is the root adjective of "smoothly")

While semantic relations apply to all members of a synset because they share a meaning and are all mutually synonyms, words can also be connected to other words by lexical relations, including antonyms (opposites of each other) and derivationally related words.

WordNet also provides the polysemy count of a word: the number of synsets that contain the word. If a word participates in several synsets (i.e. has several senses), then typically some senses are much more common than others. WordNet quantifies this by the frequency score: in several sample texts all words were semantically tagged with the corresponding synset, and then it was counted how often a word appeared in a specific sense.

The morphology functions of the software distributed with the database try to deduce the lemma or root form of a word from the user's input; only the root form is stored in the database unless it has irregular inflected forms.

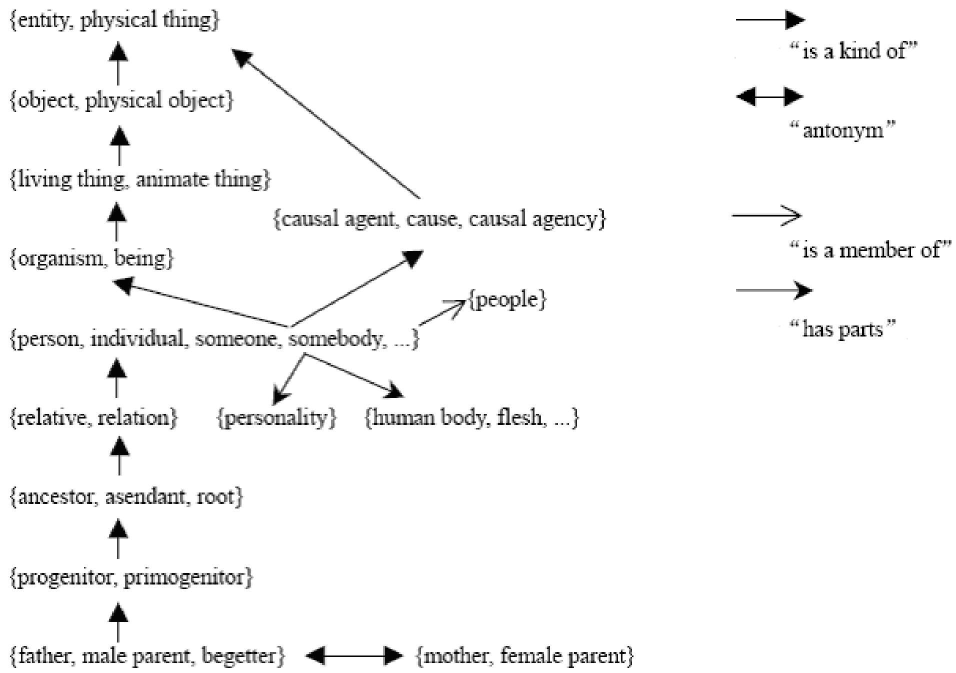


Figure 2.2 Demonstrations of WordNet Relations

As WordNet also contains many domain specific terms, concepts and relations between concepts, importing the domain specific entries from WordNet is very useful. The imported entries can be used to prepare training set and testing set for relation extraction in evaluation. It is can also be used as the base to extend the domain core ontology.

2.2.2 Ontology Engineering Processes

A set of processes is included in modeling a particular domain and is referred as "ontology lifecycle". [Pinto, 2004] proposed the following ontology building sub-processes which is similar to software engineering.

Specification identifies the purpose and scope of the ontology

Conceptualization describe the ontology in a conceptual model according to specifications

Formalization transforms the conceptual description into a formal model**.**

Implementation implements the formalized ontology in a knowledge representation language.

Maintenance updates and corrects the implemented ontology.

Knowledge acquisition acquires knowledge about the subject either by using elicitation techniques on domain experts or by referring to relevant bibliography.

Evaluation technically judges the quality of the ontology.

Documentation report what has done, how it was done and why it is done. It is of especial importance that the concept terms and relations should be clarified in documents.

Since manual ontology building is time-consuming and costly, ontology learning is taking into focus. **Ontology learning** develops methods and tools to reduce the manual effort for engineering and managing ontologies [Maedche, 2001]. Ontology learning includes the following sub-processes:

Importing reuses and import existing ontologies.

Extraction models the major parts of the target ontology. The input of extraction may be based on the imported ontology.

Pruning prunes the extracted ontology to better satisfy the purpose and specification of the ontology learning task.

Refinement finely tunes the target ontology and support the evolution of the ontology.

2.2.3 Ontology Learning from Text

In this research, text is used as the sources to learn ontology. Researches on ontology learning from text can learn **term**, **concept**, **relation** and **axiom**. Among the four targets to be learned, term is the bases for concept. Relation is the most critical one to form a knowledge structure among terms and concepts. Concept acquisition is often related to synonym relation extraction, instance-concept relation extraction and so on. In practical, axioms often are inherited from existing ontology. A typical ontology learning task starts from terminology extraction, then import known upper ontology knowledge tailored for the target ontology, and extract kind-of relation from text to link more concepts into concept hierarchy, and finally form an automatically built ontology to be further revised by domain experts. In this way, the target automatically built ontology is an ontology aligned with upper ontology, with term realization and reasonable structure.

2.3 Review of Terminology Extraction

Term is the lexical unit for effectively representing domain knowledge. Algorithms for automatic term extraction compute at least a domain index and a unit index. So Unithood and termhood are introduced by [Kageura, 1996]. Unithood refers to the degree of strength or stability of syntagmatic combination or collocation. Termhood refers to the degree that a linguistic unit is related to domain-specific concepts.

2.3.1 Unithood

Unithood can be measured using supervised and unsupervised method. Many machine learning methods has been tried, such as Supporting Vector Machine [Li, 2004], Decision Tree C4.5 [Sornlertlamvanich, 2000], and etc. However, supervised method needs manual annotation of training set which is time and money consuming. Supervised methods also suffer from domain dependence that the model trained in one domain may have its performance dropped significantly when applied into another domain.

To study domain independent internal principles of how term unit is formed, it is more desirable to explore unsupervised methods. In general, there are two kinds of unsupervised statistic-based measures for estimating the unithood of a candidate term [Luo, 2003]. The first kind uses internal measures to estimates the soundness through internal associative strength between constituents of a candidate. Nine widely adopted internal measures are listed in [Schone, 2001], including frequency [Guiliano, 1964], mutual information [Church, 1990; Schone, 2001], symmetric conditional probability [Ferreira, 1999], dice formula [Dice, 1945], *log*-likelihood [Dunning, 1993; Daille, 1994], *chi*-squared [Church, 1991], *z*-score [Smadja, 1993; Fontenelle, 1994], student’s *t*-score [Church, 1990]. The second kind uses contextual measures to estimates the soundness through dependency measures of the candidate on its context such as the left/right entropy [Sornlertlamvanich, 2000], and the left/right context dependency [Chien, 1999]. There is not much comprehensive study of unithood on integrating contextual measures with internal measures to get better performance.

2.3.2 Termhood

Many research works have been done on term extraction in a specific domain (terminology extraction). Various methods for measuring the domain specificity of a word have been proposed in term extraction.

[Nakagawa, 2002] made an assumption for automatic recognition of domain specific terms, that is, "terms having complex structure are to be made of existing simple terms". So he only focused on the relation between single-noun and compound noun. The compound nouns can be determined as domain specific by measuring and scoring each single-noun as their part in a given document or corpus. However this method cannot deal with non-compound terminology.

The most commonly used measurement for termhood measurement is Term Frequency Inverse Document Frequency (TF-IDF). It calculates the termhood by combining word frequency with a document and word occurrence within a set of documents.

[Frank, 1999] focused on domain-specific key phrase extraction. He considered only two attributes for discriminating between key phrases and non-key phrases—the TF-IDF score of a phrase, and the position of the phrase's first appearance in the whole document. However, classical measures such as TF-IDF are so sensitive to term frequencies that they fail to avoid very frequent non-informative words. And since Frank mainly focused on key phrases of a document, the second feature may not help much in extracting terminology.

[Hisamitsu, 2002] used the baseline method for defining the representativeness of a term. The document set which contains all the documents is labeled as D_0. Documents that contains the term *T* is labeled as *D(T)*. If a term is topic specific, all the terms in *D(T)* should probably have different distribution in D_0. They developed a method called the baseline method to compute the difference between the two distributions. Baseline method cannot handle some "background noise"—words which are irrelevant to term *T* and simply happen to occur in *D(T)* (documents containing *T*). This is the part to be offset by the baseline function. Based on the idea of the baseline method, [Hisamitsu, 2002] used another approach to measure the bias of word occurrences. The number of words with saliency over a threshold value is taken as the degree of bias of word occurrences in *D(T)*. The algorithm has good performance, but its running time is quite long.

[Chang, 2005] proposed a statistical model for finding domain specific words (DSW). He defined Inter-Domain Entropy (IDE) by acquiring normalized relative frequencies of occurrence of terms in various domains. Terms whose IDE are above a threshold are unlikely to be associated with any certain domain. Then the top-k% candidates can be determined as the domain specific words of the domain.

2.4 Review of Kind-of Relation Extraction

Researches on Kind-of relation extraction can be classified into internal component based methods, linguistic pattern based methods, clustering based methods, and formal concept analysis (abbreviated as FCA) based methods.

Internal component based methods uses internal information to discover the relations. [Navigli, 2004] introduced a method "Semantic Interpretation" which is the process of determining the right concept (sense) for each component of a complex term (this is known as sense disambiguation) and then identifying the semantic relations holding among the concept components, in order to build a complex concept. For example, "bus service" can be semantically interpreted as bus#1 and service#1 taken from WordNet. The relation between the two components is INSTR (instrument, which means the service#1 is operated through the instrument bus#1). Although the method using internal information seems simple and effective, but there are still many kind-of relations have their two concept terms one of which is not the component of another.

Linguistic pattern based methods exploits linguistic patterns which represent kind-of relations in corpus, web document and so on. [Hearst, 1992] use several simple patterns such as "NP_0 *such as* {NP_1, NP_2, ... , (*and* | *or*)} NP_n" in a large corpus to find the hyponym relations. [Cimiano, 2004] proposed a method to use patterns in Google to collect evidence from the Web for the different concepts a given instance could belong to. Linguistic pattern based methods commonly achieve high precision, but low recall.

Clustering based methods extract kind-of relation by comparing vectors of contextual information of different concepts. [Caraballo, 2001] scans the corpus to build term vector for each term with co-occurring terms within a special context, then select or define a similarity algorithm and a hierarchical clustering method to compute the clustered node. By default the clustered nodes or the middle node of the tree have not a label. To assigns hypernyms for the middle node, Caraballo first use patterns to extract hypernym for every word according to Hearst's work which is referred to in previous chapter. For each internal node of the tree, they assign the node a hypernym if the hypernym has maximum number of direct related child node of the current node. Clustering method does not need too much rules and simple input "corpus". It can identify many implicit concepts without any linguistic realization, and can form a concept hierarchy in one time. However the hierarchy suffers from weakness in labeling clustered nodes, abstraction level control and rationality to human beings.

The last one is FCA based methods. FCA is a formal technique for data analysis and knowledge representation [Li, 2005]. It can be used to construct formal concepts as a lattice automatically for a given context. This can replace the time-consuming manual work of building domain ontology. FCA takes two sets of data to represent concept, one is called the object *set* which contains all the possible instances denoted by the abstract concept. The other one is called the *attribute set* which contains all the attributes owned by the concept. Consequently FCA finds a binary relationship between the data of the two sets, and further constructs a so-called formal concept lattice with a concept inclusion ordering according to a *formal context*.

In this work, ontology is defined below as formal ontology according to [Sowa, 2000].

Definition 1: An ***ontology*** in FCA, denoted by *O*, is defined by a quadruplet, *O = (L, D, C, R)*, where *L* is a specific language, *D* is a specific domain, *C* is the set of concepts and *R* is the set of relations between concepts.

Definition 2: A ***formal context*** is a triple *(G, M, I)* where G is a set of objects, M is a set of attributes, and I is the relation on $G \times M$.

Definition 3: A ***formal concept*** of the context *(G,M,I)* is a pair *(A,B)* where $A \subseteq G$, $B \subseteq M$, $A^{'} = B$ and $B^{'} = A$, Where $A' := \{m \in M \mid (g, \mathrm{m}) \in I, \forall\ g \in A\}$ and $B' := \{g \in G \mid (g, \mathrm{m}) \in I, \forall m \in B\}$.

For a formal concept *(A, B), A* is called the *extent* and B is the *intent* of the formal concept. Formal concepts satisfy the ***partial ordering relationship***, denoted by $\leqslant$, with regard to inclusion of their extents or inverse inclusion of their intents, formalized by:

$$(A_1, B_1) \le (A_2, B_2) \Leftrightarrow A_1 \subseteq A_2 \ and \ B_2 \subseteq B_1$$

The whole formal concept lattice satisfies the partial ordering relations. When applying FCA to ontology construction, each term used in a specific domain can be mapped into an object in FCA. Thus, a term along with its set of attributes forms a node as a formal concept in the FCA lattice. Along the partial ordering relationships built based on the definition given earlier, relationships among different terms can be found., A formal concept in FCA corresponds to a concept in the concept set *C* for a specific language *L* and a specific domain *D* and the partial ordering relationship in FCA corresponds to *R*.. How to make use of the partial-ordering relation to get relations between terms is the subject of an application making use of the FCA model.

FCA can use co-occurring words in some context as attributes to extract kind-of relations. The extraction of kind-of relations from text by comparing their contexts holds the *distributional hypothesis* [Harris, 1968] that terms are similar to some extent to which they share linguistic contexts. Many researches have been done based on the hypothesis. Among all these researches, clustering using the similarity of pairs is mostly tackled. In clustering, a vector space model is adopted and terms are represented as a vector containing features or attributes. After defining a certain similarity measure, clustering method can compute final clusters. Typical works by clustering from text include Hindle [1990], Caraballo [1999] and Bisson [2000].

FCA is an effective technique for the construction of formal ontology. In general, most works focus on the selection of formal objects and attributes. In [Haav, 2003], a text describing a certain entity is seen as an object and thus an object used in FCA can be any domain-specific text that use domain-specific vocabulary and describe domain-specific entities. Attributes of an object are noun-phrases that are present in the domain-specific text. An ontology used in the real estate domain was then constructed. In [Cimiano, 2005a], verb-object dependencies are extracted from text where the headwords of objects are considered FCA objects and the corresponding verbs together with the postfix "able" are used as attributes.

2.5 Review of Core Ontology Construction

A core ontology is to bridge the gap between an application oriented domain ontology to an upper ontology, it must maintain certain properties. The following gives the required properties which are considered fundamental in Chinese core ontology construction.

Firstly, the concepts represented in a core ontology must be **widely accepted and commonly referenced**. Since core ontology contains fundamental concepts in a specified domain, obviously a core concept should be widely used and commonly referenced in a consistent way in the domain. In case of polysemy, it is important to identify the correct sense. For example, the term “system” (系统) is widely used in the IT domain. However, one of its senses with the definition “*a group of physiologically or anatomically related organs or parts*” is obviously not a core concept in IT domain. So it is important to identify core terms and the appropriate sense(s) used in the domain.

Secondly, its representative core terms must be **highly used and should be productive to compose longer terms.** It is understood that a concept must presented by some lexical terms. Core terms represent the most fundamental concepts in a domain. As the realizations of the core concepts, core terms must be commonly used in a domain (e.g.: “software” in IT domain). Core terms should also have strong ability to form longer terms used in a domain ontology. In another words, core terms should have strong ability form longer terms so that the core ontology can link more domain specific concepts (often represented by longer domain terms) to upper ontology. An observation made by the study shows that the top 1,500 most productive core terms extracted can serve as suffixes to form more than 50% of the terms in a domain specific term bank (The term bank contains about 130K entries of IT terms). And since in most cases (93%), a Chinese term act as the suffix of a longer term is the head words of the longer term [Cui, 2008], the core ontology constructed using core terms can directly map most of the domain specific concepts to the upper ontology in theory. From the definition, core terms should be nouns in a specified domain. However, many morphemes serving as components in the domain can be also productive and have stable meaning in composing longer terms. Thus, based on the functional definition, these morphemes are also included as “core terms” in this thesis.

Thirdly, the concepts/terms can be **mapped to upper ontology**. This ensures that the core ontology is not a dangling concept which does not have any relation to the upper ontology. It also ensures that all concepts in the core ontology can inherit the attributes provided by upper ontology. Upper ontologies are relatively small in size and are more carefully designed with additional more information such as axioms. Furthermore, the mapping to upper ontology can help the domain ontology to merge or interoperate with other domain specific ontology.

There are not many directly related works on automatic core ontology construction. Some ontology related researches build up their concept nodes using a core lexicon [Hirst, 2004]. However, many of the ontologies including Chinese ontology are manually built [Huang, 2004; Tang, 2005] and are often for a special application [Doerr, 2003]. Few works are reported on automatic construction of core lexicon and core ontology. It is more difficult to automatically construct Chinese core ontology because there are relatively limited natural language resources compared to that of English. For example, HowNet, the most commonly used semantic resource for Chinese, contains about 20K number of concept nodes in its 2002 version [Dong, 2006] whereas WordNet [Fellbaum, 1998] contains close to 130K concept nodes in its version 1.6 database.

Besides core ontology construction, upper ontologies are mainly manually constructed. A widely known upper ontology is the Suggested Upper Merged Ontology (SUMO) which is a part of the IEEE Standard of the Upper Ontology Working Group by merging public available ontological content into a single, comprehensive and cohesive structure [Pease, 2002; Niles, 2001]. A widely known English lexicon ontology is WordNet [Miller, 1990]. An important notion in WordNet is "Synset" which is a set of synonym terms representing one unique concept. Different senses of a word are included in different synsets. From the perspective of an ontology, a synset is equivalent to a concept in an ontology. Another two famous upper level ontology works based on lexicon are CoreLex [Buitelaar, 1998] and the base synsets of EuroWordnet [Rodríguez, 1998].

Chapter 3

Terminology Extraction using Unithood and Termhood Measures

Automatic terminology exaction is a major topic in natural language processing and has a wide variety of applications such as dictionary generation, keyword extraction for information retrieval and ontology construction. As mentioned in the literature review, terminology extraction includes both unithood and termhood measures. Most researches mainly focused on unithood measures to identify potential terms first. To improve the performance of term extraction from domain corpus, both unithood and termhood are considered in this study. Internal information and neighbouring information of a term candidate is integrated to compute the unithood in an unsupervised style towards better extraction result. Several termhood algorithms which identify domain specific terms from unithood computation are evaluated to find out how the performance of terminology extraction can be designed with unithood and termhood working together to improve performance.

3.1 Unithood

Unithood [Kageura, 1996] measures the suitability of a string candidate being an independent linguistic unit so as to serve as a term. As unithood measures often exploit statistical information of strings in a corpus, definitions on corpus statistics are given first. After the description of several existing algorithms, a proposed new algorithm, *UREval,* is presented.

3.1.1 Definitions

General definitions on string computation are firstly introduced to facilitate the formal description of different algorithms. Many of the definitions come from Lv [2007].

Given a corpus *C* which is formed by a sequence of characters, a **string** is simply a sequence of characters in *C*. Generally speaking, *C* can be considered a string by itself. A **word** is a string which is considered as a lexical unit having independent meaning and can be used to form sentences. A string can contain a number of words, or some part of a word. The **length of a string** is measured either by the number of characters or the number of words in the string depending on the statistical units of interest. A **statistical unit** is the unit (as a string) used for the collection of statistics in *C.* a Statistical unit can be a character, a word, or a substring of a word, depending on whichever unit of study is needed. Other symbols such as punctuation marks, can also serve as statistical units if so desires. As a result, Corpus *C* can also be considered as a sequence of statistical units. For simplicity, the terms “statistical unit” and “unit” can be used interchangeably.

For a string *S*, $S=s_1s_2...s_n$ with *n* number of statistical units, s_i is called a **component** of *S*, and the sub-sequence $s_i ... s_j$ where $i=1...n$, and $j = 1 ...n$ ($i<j$ but not $i=1$ & $j=n$) is called the **substring** of *S* and *S* is called the **superstring** of $s_i ... s_j$. Further more, if there exists a $s_0s_1...s_n$, s_0 is called the **left adjacent unit** of *S*; and if there exists $s_1...s_ns_{n+1}$, s_{n+1} is called the **right adjacent unit** of *S.*

For $S=s_1s_2...s_n$, a sequence of point $p_0p_1...p_n$ can be inserted into *S* to form the sequence $p_0\ s_1p_1\ s_2...s_n\ p_n$ where p_i ($0\leqslant i\leqslant n$) is called the **insertion point** at position *i* indicating the boundary position of the *i*th statistical unit. An insertion point p_j , where $1\leqslant j\leqslant n-1$, is called the **internal insertion point** of *S*. The insertion point p_0 is called the **left boundary insertion point** and p_n is called the **right boundary insertion point**.

3.1.2 Baseline methods

As introduced in literature review chapter, there are many algorithms to calculate the association of the statistical units. Three of them are selected here as baseline algorithms: (1) mutual information, (2) log-likelihood, and (3) left/right entropy. The first two are the two most commonly used algorithms to calculate pair-wise associations which are considered internal measures. The third one gives a boundary measure.

Point-Wise Mutual Information [Church, 1990; Schone, 2001]:

Mutual information is a measure of association. For two statistical units, *x* and *y* having probabilities *P(x)* and *P(y)*, function *P(x, y)* is the joint probability of *x* and *y.* Their mutual information *PMI(x, y)* and a simplified version *MI(x, y)* are defined as

$$PMI(x,y) = \log\left(\frac{P(x,y)}{P(x)P(y)}\right) \qquad \text{(E3.1)}$$

$$MI(x,y) = \frac{P(x,y)}{P(x)P(y)} \qquad \text{(E3.2)}$$

Mutual information compares the joint probability (the real probability of observing *x* and *y together*) with the estimated probabilities of *x* and *y* under the *independence* assumption that the corresponding random variables *X* and *Y* are independent. If a strong association exists between *x* and *y*, the joint probability *P(x,y)* is larger than *P(x) P(y),* thus *PMI(x,y) >> 0*. If no strong relationship exists between *x* and *y, P(x,y)≈ P(x) P(y),* and *PMI(x,y) ≈ 0.* If *x* and *y* are complementary, *P(x,y) is* much less than *P(x) P(y),* making *PMI(x,y) << 0*.

Log-likelihood [Dunning, 1993; Daille, 1994]

Log-likelihood can also measure association between statistical units. For two statistical units *x* and *y* having probabilities *P(x), P(y),* and *P(x, y),* let $a = N^*P(x,y)$, $b = N^*P(\bar{x},y)$, c= $N^*P(x,\bar{y})$, $d = N^*P(\bar{x},\bar{y})$ where *N* represents the total number of occurrences of all statistical unit, $\bar{x}$ represents all none-x statistical units and $\bar{y}$ represents all none-y statistical units. The log-likelihood formula gives a score to the association between *x* and *y* as,

$$LL(a,b,c,d) = a \log a + b \log b + c \log c + d \log d$$

$$- (a+b) \log (a+b) - (a+c) \log (a+c) - (b+d) \log (b+d) - (c+d) \log (c+d)$$

$$+ (a+b+c+d) \log (a+b+c+d) \qquad \text{(E3.3)}$$

The larger the value of LL, the stronger the association between x and y is.

Left/Right Entropy [Sornlertlamvanich, 2000]

Entropy measures information uncertainty of a variable. Those term candidates which have a high information uncertainty in both its left boundary and right boundary are more likely independent terms. For a term candidate S composed of n statistical units, $S = s_1, s_2, \ldots, s_n$, suppose, s_0 is a left adjacent unit of S, and $s_0 \in S_0$ where S_0 is the set of all left adjacent units, and s_{n+1} is a right adjacent unit of S, $s_{n+1} \in S_{n+1}$ where S_{n+1} is the set of all right adjacent units. The left entropy, $LE(S)$, and the right entropy, $RE(S)$, are defined as follows:

$$LE(S) = -\sum p(s_0 S \mid S) \log_2 p(s_0 S \mid S) \qquad \text{(E3.4)}$$

$$RE(S) = -\sum p(S s_{n+1} \mid S) \log_2 p(S s_{n+1} \mid S) \qquad \text{(E3.5)}$$

where function $p(x|y)$ is the conditional probability of x given y.

For term extraction, both the left entropy and right entropy should be considered. So, to take into consideration of both the left entropy and the right entropy, [Sornlertlamvanich, 2000] defined the **Left/Right Entropy**, denoted as $LRE(S)$, as given below

$$LRE(S) = Min(LE(S), RE(S)) \qquad \text{(E3.6)}$$

which takes the minimum of $LE(S)$ and $RE(S)$ because the independence of S is determined by the weakest boundary insertion point between the left one and the right one.

The more statistical units there are on the two adjacent points of S, the larger the values of LRE are. Furthermore, the less difference between the left entropy and the right entropy, the larger the value of LRE is. Consequently, the larger the value of LRE is, the more independent S is.

3.1.3 Multiple-Features Based Unithood Algorithm

The baseline algorithms introduced above make use of occurrence probability of either the components of term candidates or their boundaries to compute the association strength of candidates. Many term candidates are a part of a longer term. Without full consideration of longer terms, these algorithms tend to obtain shorter terms because they tend to have higher statistical significance although they may actually be components of some longer terms. In this study, more contextual information of term candidate is considered besides internal measures to give a more comprehensive measure on unithood.

For a given a term candidate composed of *n* statistical units $S = s_1, s_2, \ldots, s_n$, it is very important for unithood to measure weather *S* is a relatively independence lexical unit or a part of a larger lexical unit. This requires a closer look at the boundary units. Each candidate has two boundaries, s_0 on the left and s_{n+1} on the right. Suppose *LAU(S)* and *RAU(S)* denote the set of all left adjacent units of *S* and all right adjacent units of *S,* respectively. Two contextual measures, the left dependent ratio *LDR(S)* and the right dependent ratio *RDR(S)* can then be defined as:

$$LDR(S) = \frac{\max_{s_0 \in LAU(S)} f(s_0 S)}{f(S)} \qquad \text{(E3.7)}$$

$$RDR(S) = \frac{\max_{s_{n+1} \in RAU(S)} f(S s_{n+1})}{f(S)} \qquad \text{(E3.8)}$$

where the function *f(x)* returns the frequency of *x*. The higher *LDR(S)* is, the more likely that the left insertion point of *S* is not a left boundary insertion point because there is a high probability that *S* is the component of $s_0 S$. By the same token, the higher *RDR(S)* is, the less likely that the right insertion point of *S* is the right boundary point.

Consequently, the two boundary insertion points of the term candidate should be independent for the candidate to qualify as a term. The *i*n*d*ependence *r*atio of a candidate *S*, labeled as *IDR(S)*, should take into consideration of both its frequency and the boundary dependency (both *LDR* and *RDR*) as defined below

$$IDR(S) = (1 - 1/f(S)) \times \sqrt{(1 - LDR(S)) \times (1 - RDR(S))} \quad \text{(E3.9)}$$

Basically, the higher *f(S)* is, the more independent *S* is. To normalize it, $(1 - 1/f(S))$ is used in E3.9. As *LDR(S)* and *RDR(S)* compute the dependency of boundary insertion points, $(1-LDR(S))$ and $(1-RDR(S))$ indicate boundary independence of the left and the right boundary insertion points, respectively. Therefore, *IDR(S)* is a more comprehensive measure than the Left Entropy and Right Entropy used in [Sornlertlamvanich, 2000].

Furthermore, all the internal insertion points of a term candidate *S* should not be independent for *S* to be considered as an indivisible unit, and thus, a lexical term. In other words, to qualify as a term from the micro level, all its internal insertion points should have low independence ratio. For each insertion point p_i between s_i and s_{i+1}, the Non-In**d**ependence Ratio, $NIDR(s_i s_{i+1})$, is defined below as the ratio of two statistical units within a string in terms of non-independence:

$$NIDR(s_i s_{i+1}) = (1 - 1/MI(s_i, s_{i+1})) \times \sqrt{RDR(s_i) \times LDR(s_{i+1})} \quad \text{(E3.10)}$$

where $NIDR(s_i s_{i+1})$ is defined by considering two parts. The first part, $(1- 1/MI((s_i s_{i+1}))$, is derived from the mutual information algorithm *MI* where function $MI(s_i, s_{i+1})$ is given in (E3.2). The higher the value of $(1- 1/MI((s_i s_{i+1}))$, the more non-independent the internal insertion point is between s_i *and* s_{i+1}. The second part, $\sqrt{RDR(s_i) \times LDR(s_{i+1})}$, is defined based on (E3.8) at the micro level for both the left dependency and right dependency. In order for *S* to qualify as a term, it is important to examine every insertion point and if the weakest insertion point is still quite non-independent, *S* should qualify as a term. Based on this argument, the overall measure for all internal insertion points on non-independence, *AllNIDR(S)*, is defined as

$$AllNIDR(S) = \min_{1 \le i \le n-1} (NIDR(s_i s_{i+1})) \quad \text{(E3.11)}$$

To take into consideration of both the boundary insertion points and the internal insertion points, the proposed algorithm *UREval,* measures unithood as

$$UREval(S) = IDR(S) \times AllNIDR(S) \qquad \text{(E3.12)}$$

UREval takes values in the range of 0 to 1. The higher the *UREval* value is, the more likely *S* as a term. Since *UREval* is a ranking algorithm, a threshold value needs to be specified which should be determined experimentally.

3.1.4 Performance Evaluation

In the evaluation, *UREval* is compared to the baseline methods including mutual information, *log*-likelihood, and left/right entropy (abbreviated as *MI*, *LL* and *LRE* respectively). It should be pointed out that all baseline algorithms and *UREval* can take either characters or words as statistical units. *UREval* and *LRE* can be computed on any *N*-gram where *N* is a positive integer. 2-grams, 3-grams and 4-grams are the most commonly used units to form a large majority as indicated by analysis to an IT term bank [Chen, 2007b].

The evaluation is first made on character based bigram unithood computation to prove the ability of *UREval* to extract longer terms especially those mostly used in 2/3/4-grams, *UREval* and *LRE* are compared in character based 3-gram, and 4-gram unithood computation. Then, *UREval* is applied in word based bigram unithood computation to two corpora to show its performance at the word level and its stability across different corpora.

1. Evaluation of the character based unithood measure

In the character based unithood measure, the test data is the raw corpus which contains the complete text of *the Journal of Software* published in 1998. The reason to use this corpus is that software is a typical sub-domain of IT which keeps developing with new terms carrying new concepts.

Each of the four algorithms assigns every term candidate a weight to rank within the candidate set. Consequently, a threshold *N* is needed to determine the number of top candidates which should be kept in the result. From the *UREval* formula, it can be seen that only the term candidates with at least two different left neighboring grams and two right ones would have *UREval* greater than 0. Thus satisfying the two conditions is the minimum requirement for picking the top *N* candidates in *UREval.* To make comparison fair, all the other baseline algorithms also use the same *N*.

The performance is manually examined through sampling one out of every 10 consecutive candidates in the top *N* candidate list. The evaluation criterion is whether the candidate is an independent linguistic unit. For example, "数据"(data) and "数据仓库" (data warehouse) are real units because they are both syntactically sound and can be used without dependence on other words. "模型具有" (model has) contains two syntactic components and is thus not a unit.

The value of *N* for the bigram set according to the minimum requirement of *UREval* is 14,150.

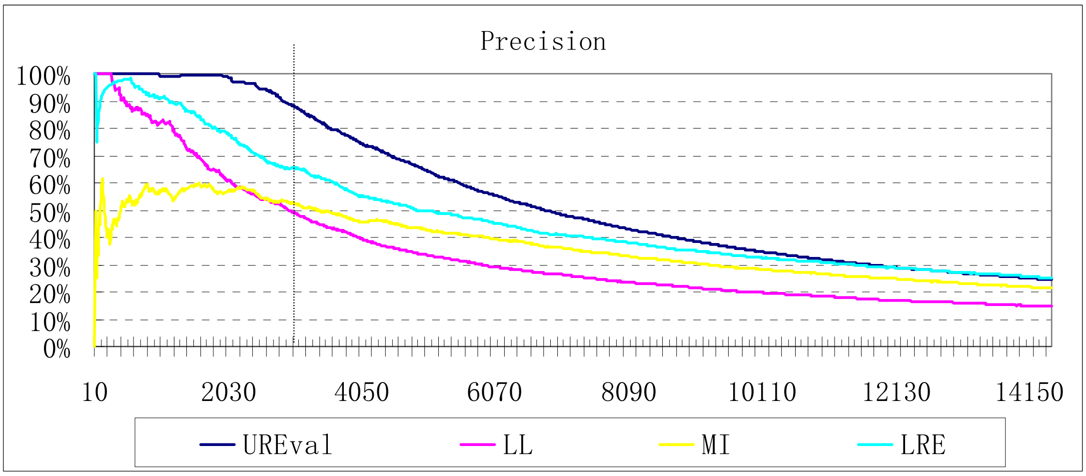


Figure 3.3 Precision in Character-based Bigram Unithood Computation

Figure 3.3 shows the performance of the four algorithms against the top *i* ranked candidates (the x-axis) using the character-based bigram unithood measures. It can be shown that *UREval* is significantly better than all the other algorithms. Its precision is 100% up to the top 980 candidates, 99% up to the top 2,070 candidates, 95% up to the top 2,470 candidates, 90% up to the top 2,890 candidates. It is interesting to note that the *Log-Likelihood* method achieves the second best performance at the beginning when *N* is very small, yet it soon drops to the worst one. A possible reason is that the *Log-Likelihood* model is not linearly correlated to the real data. On the other the other hand, both *Mutual Information* and *Left/Right Entropy* have similar performance curves indicating that different contextual measures can all be effective as the internal measures. However, as a single feature, neither the internal measure nor the external measure is enough to give good performance. The significant performance improvement by *UREval* proves that contextual measure and internal measure are complementary to each other and the combined use of them proposed in *UREval* is more effective.

For unithood identification, it is particularly more important to identify all correct units so that no recall is lost in termhood identification. Thus, a better recall of units (through unithood) is indeed a prerequisite for better termhood performance. Although true recall of unithood is not possible given such a large corpus, yet, given the same number of candidates, the higher the precision is, the higher recall of units there are. Thus, Figure 3.3 actually indirectly proves that the proposed algorithm has better unithood recall.

As an additional measure, it is also interesting to know the rate of change of precision. In principle, the smaller the rate of change (either positive or negative), it is for an algorithm, the more stable the algorithm is. A peak analysis would give an indication of where a good selection of N is or not as the drop from the peak indicate the most significant drop in Precision, and thus it would be a good cut-off point for N. By definition, *RCP* is equal to *Precision (i)-Precision (i-1)*, but range size 1 is too small and makes *RCP* mostly be 0 and always trembling between 0 and other value. Thus range size 100 is selected and *RCP* is set to be (*Precision (i)-Precision (i-100))/100.* Figure 3.4 shows the (*RCP*) against top *i* candidates.

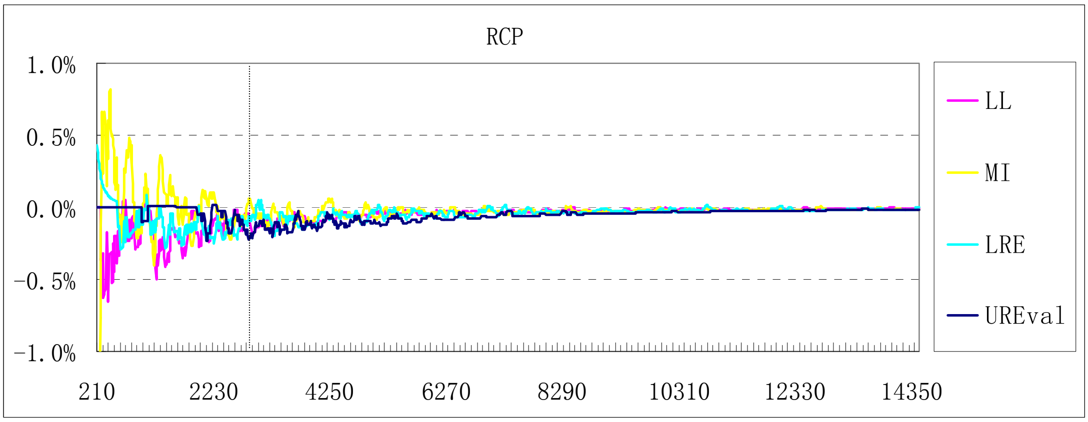


Figure 3.4 RCP of Character-based Bigram Unithood Measures

Figure 3.4 shows that the *RCP* of *UREval* is almost all 0 before the top 2,000, and is the closest to 0 which means that it is the most stable one among all the reference algorithms. This proves the significant performance improvement of *UREval* over other baseline method. The peak value of *UREval* is achieved at about the position 3,000 which is marked out by a dotted line. This indicates that selecting 3000 as a threshold *N* achieves a kind of best combined performance between precision and recall.

The second set of experiments evaluates trigram performance of the character-based algorithms as real term units. The size *N* of the trigram set according to the minimum requirement of *UREval* is 13,451. As both *LL* and *MI* works only on pair-wise bigrams, *UREval* is compared to *LRE* only. The performance on precision of the trigram unithood measures are shown in Figure 3.5.

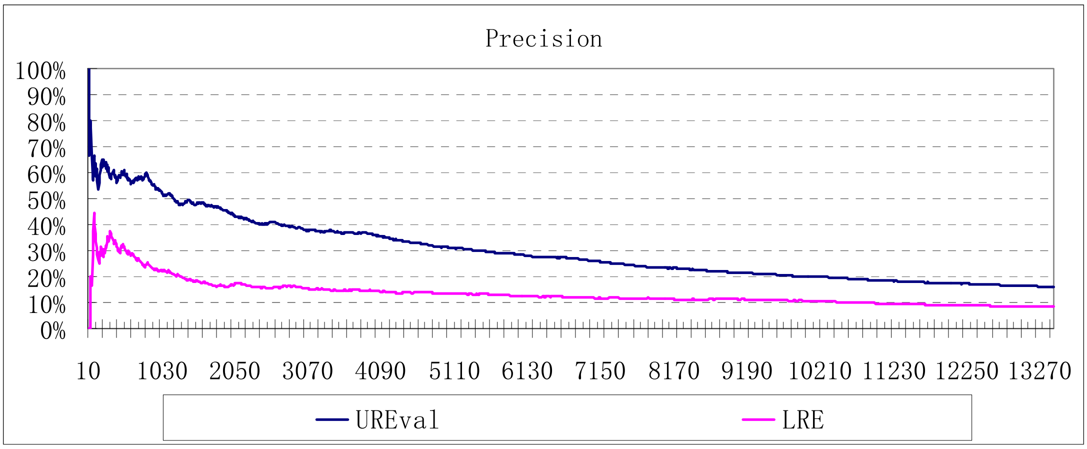


Figure 3.5 Precision of Character-based Trigram Unithood Measures

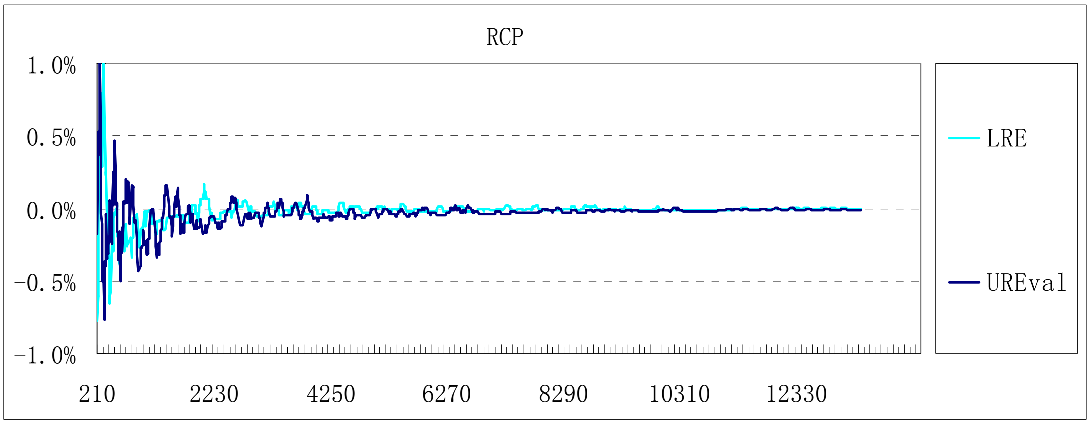


Figure 3.6 *RCP* in Character-based Trigram Unithood Computation

It can be seen in Figure 3.5 that *UREval* outperforms *LRE* significantly in precision (and thus in recall) over the complete range of the evaluation. The best performance can be about 100% better than that of LRE. However, the precision of trigram is obviously much worse than that of the bigram. A possible reason is that the testing data used in this experiment is not large enough. Thus, longer terms do not have enough statistical information as backing.

Figure 3.6 shows the *RCP* for tri-grams. Results show that both algorithm displays similar stability in terms *RCP*. Thus, there is no advantage for *UREval* in this measure. As the corpus size may be the main issue, lack of statistical information makes the result less thrust worthy too.

The 3rd set of experiment is on character-based 4-gram unithood computation. The size *N* of the 4-gram result set according to the minimum requirement of *UREval* is 5,606. The precision and *RCP* on 4-gram unithood computation is shown in Figure 3.7 and Figure 3.8, respectively.

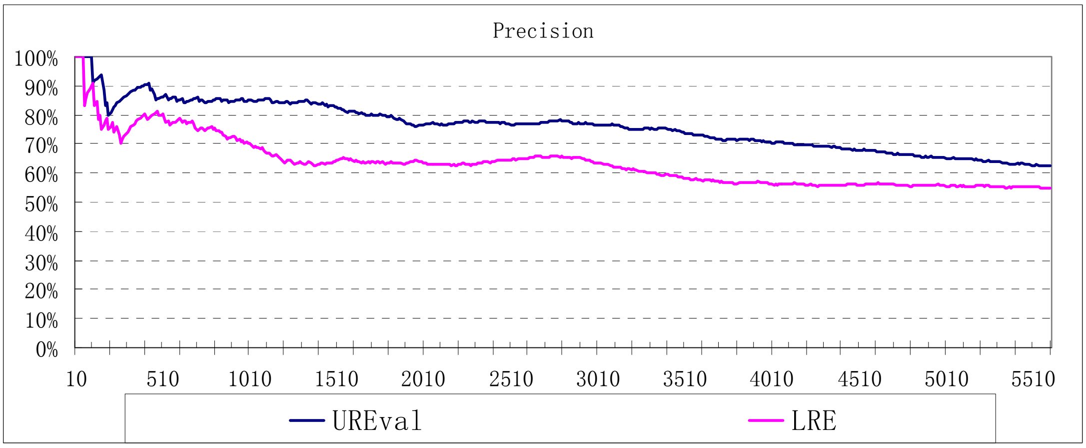


Figure 3.7 Precision on Character-based 4-gram Unithood Measures

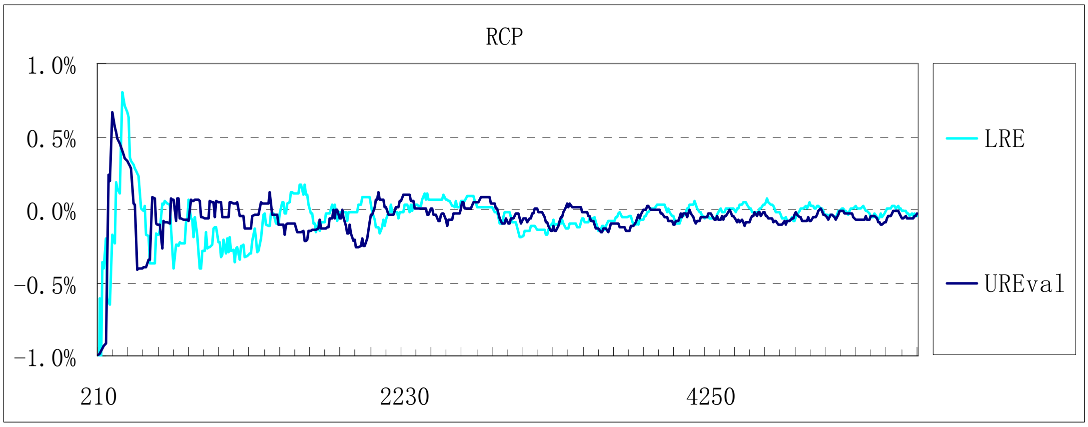


Figure 3.8 *RCP* in Character-based 4-gram Unithood Measures

Both Figure 3.7 and Figure 3.8 show the superiority of *UREval* over *LRE* using 4-gram unithood measures. In 4-gram unit extraction, the scarcity of statistical information is more severe that the size of the result set (5,606) of 4-gram term units is smaller than the size of the result set (13,451) of trigram term units while in fact the number of all 4-grams is much more than the number of trigrams. It is also found that the difference between the best precision of *UREval* and *LRE* on 4-gram is smaller than that of bigram and trigram. The observation shows that *UREval* is obviously good at utilizing the statistical information of bigrams while it is still better complete to *LRE*, yet a much bigger corpus must be used to really evaluate the usefulness of the proposed algorithm. Figure 3.8 again shows that the proposed algorithm has very similar performance on *RCP* and thus, *UREval* has no advantage in this measure.

It is, however, interesting to see that 4-gram algorithm performs better than the trigram algorithm even though less statistical data is available. Table 3.1 shows the improvement of *UREval* compared to the baseline algorithms for different length units, referred to as, GramLen.

GramLen	***NRUnit***	***NCand***	***N***	***URatio***	***BP***	***IMP***
2	3,950	35,657	14,150	11.1%	**77%**	65.9%
3	2,750	56,398	13,451	4.9%	*43%*	**38.1%**
4	7,296	39,765	5,606	18.3%	*54%*	**35.7%**

Table 3.1 Percentage improvement of *UREval* in different gram unit computation

In Table 3.1, *NRUnit, NCand, N, URatio, BP, IMP* denote "number of real term unit", "number of all candidates", "maximum number of top candidates that satisfy the minimal requirement for applying *UREval*", "the ratio of real term unit among all candidate", "Best Precision", and "Percentage Improved", respectively. Here *IMP=BP-URatio.* It is obvious that 4-gram algorithm performs better than the trigram algorithm because the performance start point (*URatio*) of 4-gram is much higher than that of trigram algorithm. The index *IMP* also clearly showed that the performance percentage improved is decreased when less statistical data is available.

In summary, the evaluation of character based unithood computation proves that the proposed *UREval* outperforms all the baseline methods as it integrates both contextual measure and internal measure and can consistently make a significant performance improvement. The improvement to bigram is most significant. This reflects the language phenomenon in Chinese where more terms are binary combinations.

2. Evaluation of the word based unithood computation

Word based extraction is conducted because many new terms are indeed compound words with more characters. The termhood evaluation in this section is based on segmented data. The same corpus *journal of software* 1998 is chosen to keep consistence. This corpus serves as the domain specific corpus labeled as *SO*.

The whole evaluation process is a little bit more complex than character based one. First all the bigram words are collected from segmented *SO* corpus with the word segmentation and POS (abbreviation of part-of-speech) tagging tool developed at the lab in PolyU [Lu, 2003]. Only bigram words satisfying a specified POS tag patterns are preserved. The specified patterns include *a*+*n*, *v*+*n*, *n*+*n* and *v*+*v* where *a* denotes adjective, *v* denotes verb, and *n* denotes noun. The pattern is selected because most bigram terms are composed according to these pattern rules by observation. In these experiments 5300 is selected as the threshold *N* to prove that the *UREval* value is greater than or equal to 0.

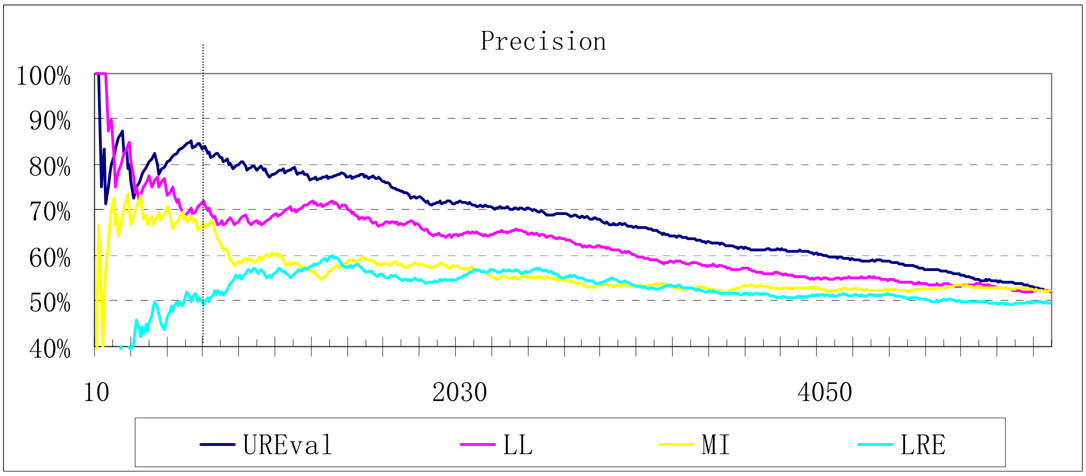


Figure 3.9 Precision of *UREval* on Word Based Extraction against Top N Candidate

The performance of bigram word extraction using *UREval* and other baseline methods is shown in Figure 3.9. Obviously, the algorithm achieved high performance in the top 100 candidates. But the drop off rate is also quick high. One reason is that the bi-word is longer than character based bigrams which makes the performance suffers from data sparseness problem. This part also shows that it works well for bigram words. Thus the results are complementing covering data of different granularity.

3.1.5 Conclusion

From the two evaluations it can be seen that the performance of *UREval* is the best comparing to all well-known baseline algorithms in unithood computation. The combination of internal measure and contextual measure is a practical way to improve the measure of unithood. However, *UREval* is still sensitive to frequency, which means that the parameters needs to be adjusted based on the corpus used. In the future, neighboring function words can be exploited to help identifying the boundary of terms. Component measures of term candidates may be given higher weights even if the frequency of some candidates is low.

3.2 Termhood

Termhood refers to the degree of a linguistic unit representing a domain-specific concept. Generally speaking, domain specificity can be identified by two types of verification. The first type of verification is through inter-domain validation. Since a term is used to represent a domain concept, it is obvious that a term is most likely to occur in a specific domain and not so often in a general domain. Thus with the availability of classified corpora, domain specific terms can be identified through cross domain validation using various features such as simple set difference or more comprehensive statistical distributions. The second type of verification is through intra-domain validation. Even within a specific domain, a term is more likely to occur many times in only certain documents where the related subjects are discussed whereas general terms are distributed more evenly in documents.

Past studies focused either on inter-domain measures or intra-domain measures and performance evaluation are conducted on different sets of data. Hence it is difficult to evaluate their relative effectiveness. This work [Chen, 2006] aims to provide a comparative study of both inter-domain algorithms and intra-domain algorithms under the same testing data set to investigate their effectiveness as termhood measures.

3.2.1 Algorithm Description

This work considered two inter-domain algorithms and one intra-domain algorithm. The first inter-domain algorithm is the inter-domain lexicon validation algorithm and the second is the inter-domain occurrence validation algorithm. Each of the three algorithms is designed based on one of the following three observations, respectively.

First, since a term is used to represent a domain concept, it is obvious that a term is more likely to occur in a specific domain and not so often in a general domain. Thus with the availability of classified corpora, term candidates can be extracted from different classes corpora. By examining the intersections and differences of the extracted lexicons of these corpora, domain specific terms can be identified. From a lexical perspective, suppose a general corpus in Chinese, labeled as *G,* is available with the associated lexicon set LEX_G, and a domain specific corpus, labeled as *S,* is also available with a lexicon set LEX_S, the lexicon set obtained from the set difference below

$$LEX_S - LEX_G \qquad \text{(E3.13)}$$

should contain the terms used in the specific domain. This algorithm based on the difference between domain lexicon sets is called the *Lexicon Set Difference* algorithm (*LSD*).

Second, even within a specific domain, the distribution of a term in the document set contains the relativity information about the terms within its domain. A typical phenomenon is that a term is more likely to occur many times in just a few documents where the related subjects are discussed whereas general words are likely to be distributed more evenly in the whole document set. The second algorithm, referred to as the *Document Crossing Algorithm* (*DCA*), is based on term frequency and document frequency in a domain corpus with a collection of documents often used in information retrieval systems [Joachims, 1997]. It uses the so called TF-IDF (term frequency inverse document frequency) method. Term frequency (TF) is the frequency of terms appeared in the corpus. Document frequency (DF) represents the number of documents in which the term occurs. Since a terminology is domain specific, it is more likely to occur many times in just some specific documents. So the value of TF divided by DF (thus called TFIDF), should be related to termhood closely. Based on the TDIDF formula used in information retrieval, below is the formula used for termhood estimation:

$$Termhood(w) = 1 - \frac{df(w)}{f(w)} \qquad \text{(E3.14)}$$

where *w* is a term, *f(w)* refers to the frequency of *w*, *df(w)* refers to the document frequency of *w*, and *Termhood(w)* refers to the termhood estimation value of *w*. *Termhood(w)* is a value between 0 to 1 and the larger the value is, the more likely a term is a terminology.

Third, the distribution of a term in different domains can also indicate the relationship between the term and its domain. Naturally, a term candidate is more likely to be a terminology in one domain if it occurs much more frequent than in other domains. The third algorithm, referred to as the *Domain Relativity Algorithm* (*DRA*), is based on the strength measure of a term with a document class. Supposedly, if a term belongs to a domain, more people in this domain use it than in other domains. In other words, they should appear more frequent in the documents of certain domain/class. Consequently, the frequency of a term appearing in different documents of a particular domain would be a good measure. Based on this observation, the strength of a term *w* with a specific domain *d*, denoted by *Association(d, w)*, is defined below as the termhood measure:

$$Association(d,w) = \frac{p(d,w) - p(d)p(w)}{p(d,w)} \qquad \text{(E3.15)}$$

where *p(d,w)* is the probability of a term *w* belongs to domain *d*, *p(d)* and *p(w)* are the independent probabilities of domain *d* and word string *w*. It should be noted that *p(w)* equals to *DF(w)* divided by the total number of documents in the document set of the corpora and *p(d)* equals to the number of documents belong to domain *d* divided by the total number of documents. *Association(d, w)* is a value between 0 and 1 and the larger the value is, the more likely a term is a terminology.

3.2.2 Preparation for Termhood Evaluation

The evaluation of termhood measures using the proposed algorithms requires two term lists with frequency information in two different domains. The preparation of the lists includes the following steps:

1. Classified Corpora Preparation: prepare a general domain corpus and a domain specified corpus.
2. Word Segmentation: the segmentation tool developed at the lab in PolyU [Lu, 2003] is applied to segment the two corpora.
3. Collecting frequency tuples: Count all the unigram and bigram in the segmented corpora to get tuples <*w, f(w, D), df(w,D), f(w,d)*> where *w* denotes a word n-gram identified by the segmentation tool in Step 2, *D* denotes a domain, *d* denotes a document. *f(w,D)* gives the number of gram *w* in domain *D*, *f(w)* represents the number of n-gram words *w* in document *d*, and *df(w,D)* represents the number of documents which belong to domain *D* containing *w.*
4. Unithood calculation: Compute unithood based on *UREval*.

The general domain corpus *G* is from the People's Daily (in January 1998) with 3,147 documents in the size of about 1.8M characters. The domain specific corpus S_{JOS} is taken from 116 documents from the Chinese *Journal of Software* in the year of 1998 with a total of 1.5M characters in size.

3.2.3 Evaluation

Each of the measures given in Section 3.2.1 is evaluated in terms of precision. As introduced in the section of word based *UREval* computation, 530 term candidates with 277 real units among them are evenly sampled from the output list with 5,300 candidates of the *UREval* algorithm. Among the 277 units, 228 candidates are indeed domain specific terminologies. To evaluate the performance of the three algorithms, two sets of experiments are designed. The first set takes the output from the *UREval* algorithm. As the list of 530 term candidates from the *UREval* algorithm contains noise, it is named the noisy term set. The second set takes the 277 real term units as the input of the three algorithms. This set is called the correct unit set. The two experiments make it possible to see the effect of noise to the performance of terminology extraction algorithms.

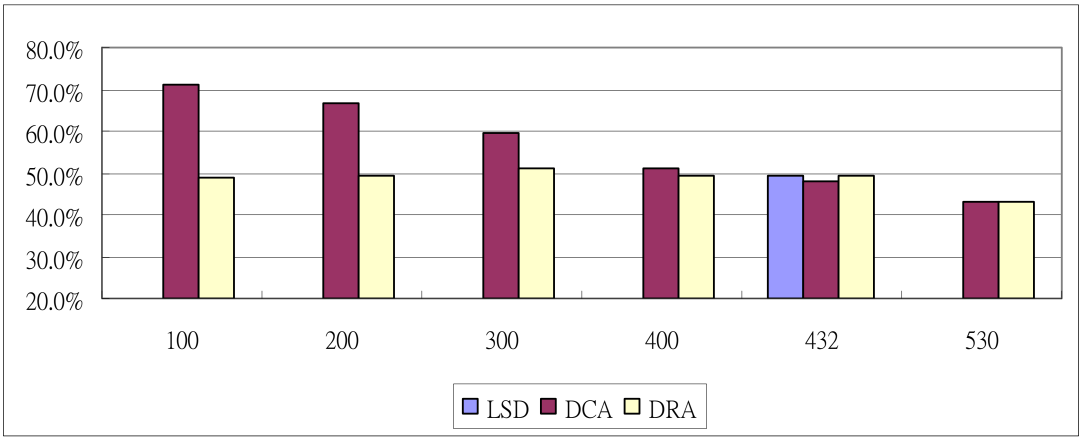


Figure 3.10 Precisions using noisy term set

Figure 3.10 shows the precisions of three algorithms on the noisy term set for the top *i* number of ranked terms. As the set difference algorithm *LSD* returns a set of 432 terms without ranking, there is only one precision for *LSD*. It can be seen that the inter-domain based *DCA* performs better than the intra-domain based *DRA* on the top 300 or so ranked results. However after 392, its performance is worse than the performance of both *LSD* and *DRA*. Some items near 432 of the ranked list generated by *DCA* show that their rank is 0. This is caused by the low frequency of these items. This means that *DCA* is very sensitive to frequency statistics. However, *DCA* is more effective when adequate occurrences are available.

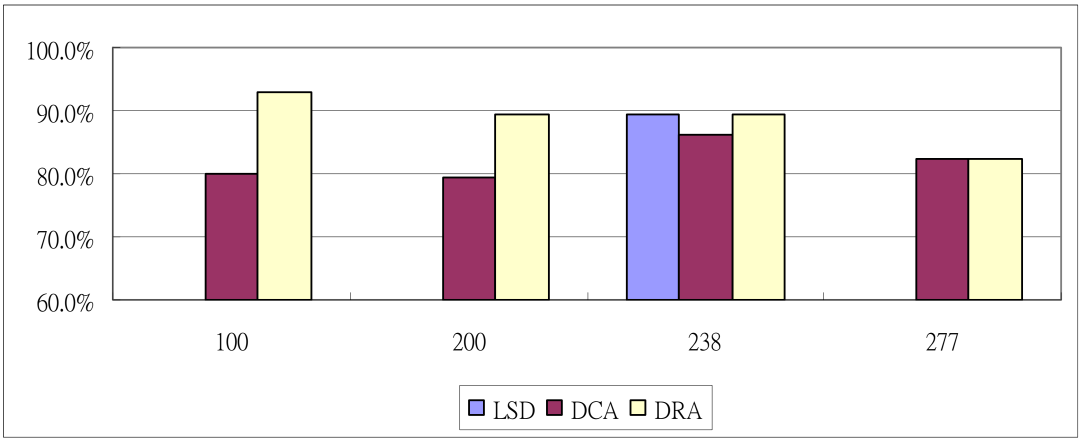


Figure 3.11 Precisions using correct unit set

Figure 3.11 shows the performance of the three algorithms using the correct lexical units as input. The relative performances of these three algorithms are almost the same as those using the noisy term set. Comparing the data in the noisy set, however, it can be seen that every algorithm in the correct unit set enjoys about 20% better performance than the corresponding algorithm in the noisy term set. This 20% increase can be seen as the direct removal of the 20% noise in the data set. This shows that the performance of Unithood calculation is extreme important for terminology extraction.

From the evaluations of both the noisy term set and the correct unit set, the performance of *DRA* are the most stable over the spectrum of data and yet at a lower level than the other two algorithms. This is because these top items get the same ranks. Given that a term occurs in the corpus S_{JOS}, *p(d,w)* is equal to *p(w),* the formula for *DRA* can be transformed into:

$$Association\ (d,w)=1-p(d) \qquad \text{(E3.16)}$$

Since *p(d)* is a constant in the evaluation, the result is a constant, too. So *DRA* lacks the ability to distinguish the relativity when the terms only occur in one domain. And since most of the terms do not occur in the general corpus, the poor performance is reasonable. Better algorithms needs to be developed to compute the relativity between a term and the domain in the future.

By examining the three algorithms in terms of complexity, it can be seen that the simplest approach in *LSD* achieves the best precision. However, the relatively poor performance of *DCA* and *DRA* may well be due to the data sparseness problem caused by the relatively small size of the domain corpus used in the experiment and the longer length of the term candidates.

3.2.4 Conclusion

In this study, three terminology extraction algorithms are selected and evaluated under the same testing set as the post-processing of unithood computation. Without noisy data, the precision of the algorithms can reach upper 80% to over 90% and degrade when unithood algorithm introduces noises. The result also shows that the simple lexicon set algorithm can be very effectively in filtering out terms of general use.

In the future, combinations of these algorithms will be investigated so that different features can be used in the same algorithm can improve the performance. Better unithood identification algorithms should be explored to minimize the noise to termhood identification. Experiments with larger domain data should also be investigated to reduce frequency sensitive of these algorithms.

Chapter 4

Kind-Of Relation Extraction using FCA and Internal Components

FCA is a formal technique for data analysis and knowledge representation [Li, 2005; Chen, 2007a]. It can be used to construct formal concepts as a lattice automatically for a given context which can satisfy the partial-ordering relationships. This can replace the time-consuming manual work of building domain ontology. As kind-of relationship naturally fit into the partial ordering relationship, so it is used in the Kind-of relation extraction in our research.

4.1 Formal Concept Analysis using Contextual Collocation

FCA is proposed to mine kind-of relations from free text corpus in this chapter. FCA is a method based on the order theory and often used in finding inherent relations between objects described with a set of attributes and attributes themselves since attributes can also be described using a set of objects [Ganter, 1999]. Our method uses co-occurred word as attributes to identify the partial-ordering relationship of concepts. If the attribute set of a concept term *A* contains all the elements of the attribute set of concept term *B*, *A* is probably a kind of *B*.

For a given set of concepts represented by a set of domain specific terms, the main steps using FCA to extract kind-of relations include: (1) select attributes and identify the object-attribute pairs as candidates from a chosen corpus, (2) collect statistics and other information from the corpus as features, to select the target object-attribute pairs from the candidates, (3) group the target object-attribute pairs by objects to generate attribute sets of concepts, then mine the kind-of relations between concepts using their attribute sets, and finally form a FCA lattice with concepts and relations mined.

4.1.1 Design of Selection Algorithms

For a given concept term, *T*, attribute selection deals with the selection of its context term, W_i in the corpus. In this work, the context words are selected as attributes in three different ways for FCA construction.

1. Three attribute types

The first attribute type considers only the immediate neighbor of *T* to extract the object-attribute pair (T, W_i) and is thus called the *Immediate Neighbor Type(INT)*. As an example, for a given term 计算机 as a concept, 计算机/组装(computer/assemble) can be considered as a pair (计算机, 组装).

The second algorithm, referred to as the *Distant Neighbor Type (DNT)***,** considers distant neighbors W_i of *T* with an additional $Offset_i$ parameter as (T , W_i, $Offset_i$). As an example, for a given term 计算机 as a concept, 组装/计算机 (assemble a computer) forms an object-attribute pair (计算机,组装, -1) where the offset is to the left by one in offset .

The third algorithm, which identifies the dependency relation of a context word W_i, denoted by DR_i, is called the *Dependency Extraction Type* (*DET*) as (T, W_i, *DRi)*. This algorithm relies on a dependency parser to identify the dependency relation DR_i . Suppose that A_{DR} is dependency relation, the Inverted dependency relation of A_{DR} is denoted as p_A_{DR}. For example, given the context 组装/计算机 (assemble a computer), the extracted pair should be (计算机, 组装, *p_VOB*) . Here VOB represents the verb-object relation.

2. Object-Attribute Features

With the chosen attributes, there are still different features that can be extracted from the corpus for identifying the object-attribute pairs. Five different features are used for the extraction of the object-attribute pairs.

a) Strong Frequency (SF)

If an attribute value A_j, co-occurs with an object O_i, at least N_1 times, where N_1 is a threshold value, the pair (Oi, A_j) is a target object-attribute pair.

b) Weak Frequency (WF)

If an attribute value A_j, co-occurs at least N_2 times with at least one object O_i in a object set O, where N_2 is a threshold value, all pairs of (O_k, A_j) which appear in the corpus with O_k belongs to O are target object-attribute pairs.

c) Conditional Probability (CP)

For all pairs of (O_i, A_j) such that the conditional probability $P(O_i \mid A_j) \geqslant P_3$ where P_3 is a threshold value and $P(O_i \mid A_j)$ is the conditional probability of O_i given A_j . The pair (O_i, A_j) is a target object-attribute pair.

d) Frequent and Dependent Index (FDI)

For all pairs of (Oi, A_j) such that

$$f(O_i, A_j)*P(O_i \mid A_j) \geqslant I_4.$$

Where I_4 is a threshold value, $f(O_i,A_i)$ represents the co-occurrence frequency between an object O_i and an attribute A_i, $P(O_i \mid A_j)$ is the conditional probability of an object O_i given an attribute A_j. The pair (O_i, A_j) is a target object-attribute pair.

e) Frequent and Relative Index (FRI)

For all pairs of (O_i, A_j) such that

$$f(O_i, A_j)*log(P(O_i, A_j)/(P(O_i)*P(A_j))) \geqslant I_5.$$

where I_5 is a threshold value, $f(O_i,A_i)$ represents the co-occurrence frequency between an object O_i and an attribute A_i , $P(O_i)$ is the probability of O_i and $P(A_j)$ is the probability of A_j. The pair (O_i, A_j) is a target object-attribute pair. The formula $log(P(O_i, A_j)/(P(O_i)*P(O_j)))$ is called the point mutual information formula which is often used to compute the relation strength between two elements.

4.1.2 Evaluations

The evaluation is based on two different resources. The first one is a corpus of 1G People's Daily Newspaper as the general corpus [4](C_1 for abbreviation). The second one is a term bank containing 130K IT Domain terms[5] (C_2 for abbreviation). This IT term bank is selected because this is an independently acquired term bank which is quite comprehensive with large coverage of the terms used in the Chinese IT domain and most of terms used in this evaluation are short and occurred as the main components of longer terms of the term bank. Also, using the term bank, it is easier to observe common modifier components around the object terms than simply use of neighboring words in sentences in a corpus. Corresponding to the two resources, the evaluation is conducted in two parts. The first part is conducted on C_1 to evaluate the effectiveness of the 5 different object-attribute features. The second part is conducted on C_2 to evaluate the effectiveness of the 3 attribute types.

A standard answer is manually made on 49 terms (they are listed in the Appendix A). All kind-of relations between the terms are extracted as the standard answer. The terms include some very common domain terms such as "计算机" (computer) , "硬件" (hardware), "兼容机" (compatible personal computer) and so on.

1. Evaluation on Different Object-attribute Features

The first set of evaluations, labeled *E1* to *E5* as listed below, is conducted on C_1 over the 5 different objects-attribute features using the simplest attribute type, namely the immediate neighbor type(*INT*):

E1) C_1 *+ SF+INT,*

E2) C_1 *+ WF+INT,*

E3) C_1*+CP+INT,*

E4) C_1*+FDI+INT,*

[4] It is consisted of articles from News in general domains, the size is about 1G byte when stored in Unicode encoding

[5] It is consisted of about 130K terms used in the IT domain

E5) C_1*+FRI+INT,*

where the corpus C_1 is segmented. Pairs of co-occurred terms within 5 segments are extracted. The reason to use *INT* is that it is the simplest attribute type which makes evaluation easier and still has show good insights on the different object-attribute features. So, *E1, E2, E3, E4 and E5* are conducted to compare the performances among different object-attribute features. Since all the five features have different threshold values and types, it is unfair to compare their performance using the same threshold. The performance in each evaluation is listed with the individual best threshold values for a fair comparison.

	Threshold	Precision	Recall	F1-Measure
SF	33	60%	33%	43%
WF	64	3.8%	10%	5.6%
CP	0.03	67%	4.6%	8.5%
FDI	0.060	30%	7.5%	12%
FRI	77	55%	37%	44%

Table 4.2 Performance of Different Object-attribute Features

It can be seen that the feature *FRI* and *SF* performs the best in terms of F-measure. Further analysis shows that frequency is vital to good recall. By observing the information used in the different algorithms, it can be seen that the heavier the frequency information is used in the formula, the better the performance of the algorithm. In fact, the frequency has the heaviest weight in *SF* and *FRI,* heavier weight in *FDI* and *WF*, and the lightest weight in *CP.* Correspondingly, the recalls of *SF* and *FRI* are the highest and the recall of *CP* is the worst. The reason why frequency is vital to recall may be that FCA uses inclusion relation of attribute sets of objects to find the object relation. The higher frequency the attributes is, the more objects the attribute occurs, the easier the inclusion relation is, the higher recall the feature gets.

The feature *CP* performs the best in terms of precision but worst in terms of recall. The worst recall is expected. However the best precision shows that conditional dependency of an attribute to the object is more accurate although its identification may be missed due to low occurrence frequency. The feature *FDI* combines both conditional probability and frequency using multiplication. Yet, its performance improvement in terms of F-measure is minor. The feature *FRI* first gets the log value of mutual information, then multiplies the value with frequency, and finally gets the best F-measure performance. The feature *SF* is the simplest by using frequency information only, while its F-measure is nearly the best. All these above statements show that frequency is the most important feature than others. With the frequency requirements, intuitively, fewer attributes are qualified. Yet, fewer attributes makes it easier to find the kind-of relations.

2. Evaluation of Different Attribute Types

The second set of evaluations, labeled *E6* to *E8*, as listed below, is conducted on the term bank C_2 over the 3 different attribute types using Strong Frequency (*SF*) as its object-attribute feature:

E6) C_2*+SF+INT,*

E7) C_2*+SF+DNT,*

E8) C_2*+SF+DET;*

where the term bank C_2 is also segmented with PoS tags (C_2 is parsed with PoS tags). Pairs of co-occurred terms within 5 segments are extracted. This evaluation uses only *SF* as the object-attribute features because *SF* is the simplest one and it is proven to be one of the best algorithms by the first part of the evaluation. Even though C_2 seems to be a small resource by size, but since it is a domain specific resource containing the term bank in the Chinese IT domain, it contains abundant target object-attribute pairs even though the corpus is small.

Evaluation *E6*, *E7* and *E8* are conducted on the dependency parsed corpus *130K_ITDomain_Termbank* and the performances are show in Figure 4.1.

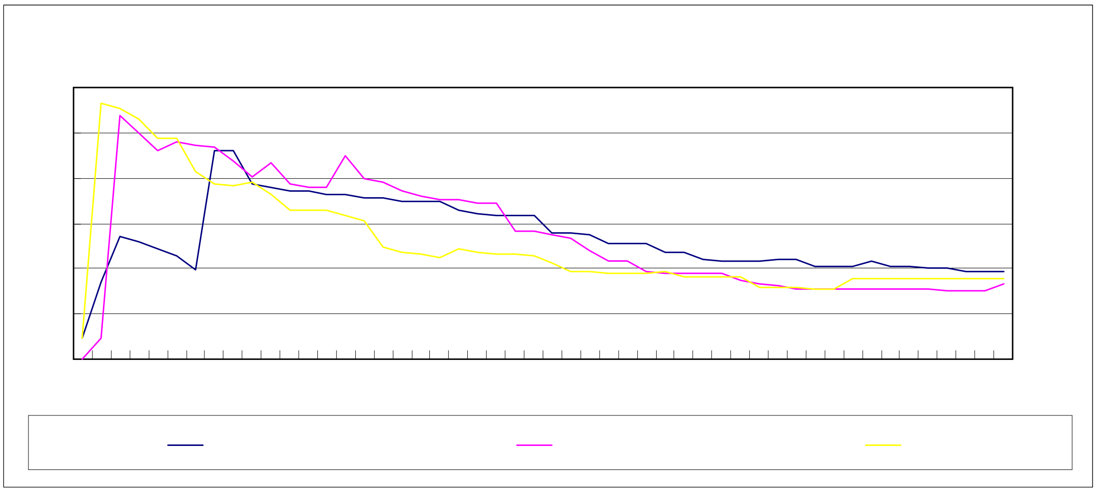


Figure 4.1 F-measure on Three Attribute Types

Figure 4.1 shows that *DET* performs the best using the lowest frequency threshold. *DET* which uses dependency information achieves the best F-Measure at about 80%, *DNT* which uses offset information achieves the second best at about 70%, and *INT* which uses neighboring word only is the worst at about 60%. It shows that more detailed and accurate information gives better result. Even the distance between the attribute and the object can provide more information.

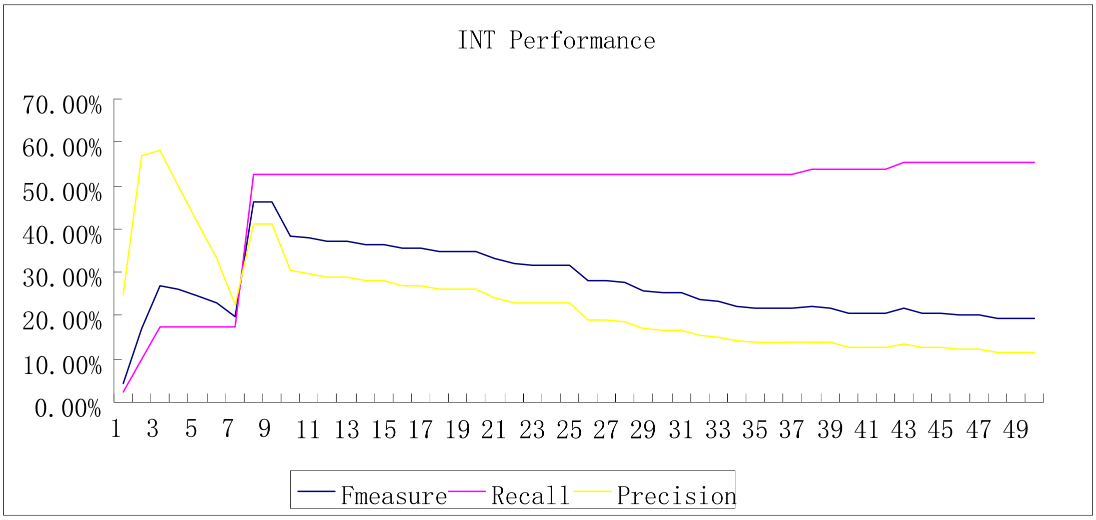


Figure 4.2 *INT* Performance

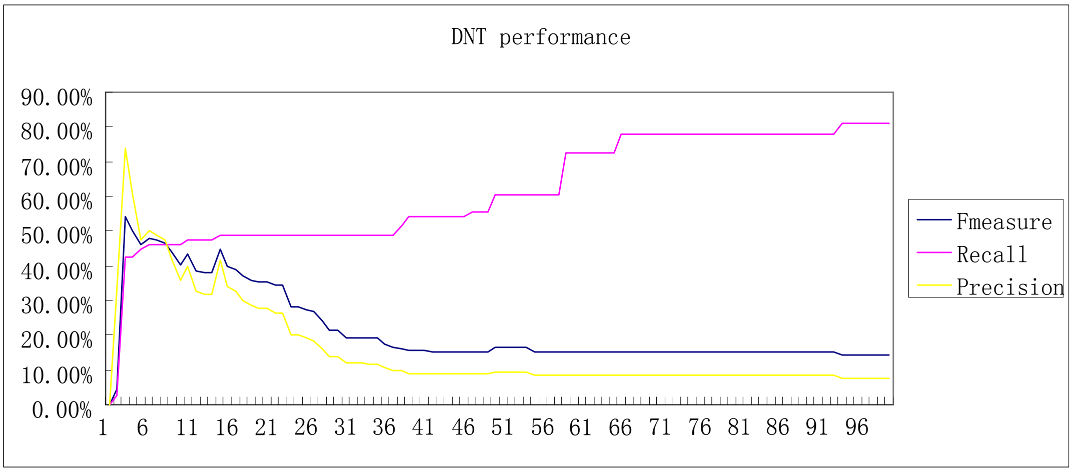


Figure 4.3 *DNT* Performance

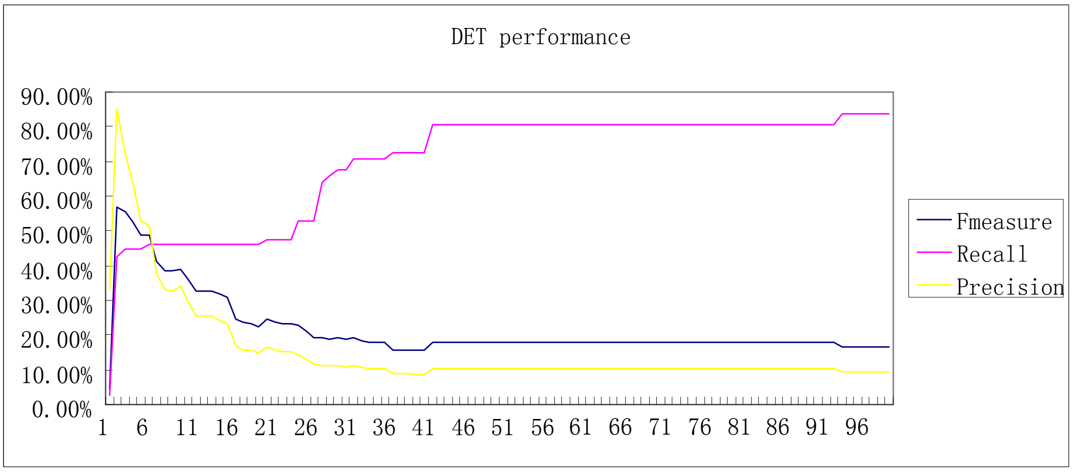


Figure 4.4 *DET* Performance

Figure 4.2, Figure 4.3 and Figure 4.4 show the performance of the three attribute types in terms of precision, recall, and F-measure, respectively. The performance curves for three algorithms are similar in trends. Higher threshold selects less attributes and makes the set of attributes of an object easy to contain all attributes of another object. So the recall rises with the threshold. On the other hand, precision rises at the beginning because the attributes with higher frequency are better. But, the precision goes down after an “optimal” point. This is because less attributes can be used to match. It shows that the number of attributes and the quality of attributes are two factors which conflicts with each other. The solutions include that getting a larger and quality corpus to increase the number and quality of attributes, decompose the attributes to smaller ones and only keep the quality ones.

4.1.3 Conclusion

It can be concluded that FCA can be one effective way for discovering the kind-of relation with careful selected corpus, algorithm, and attributes. To achieve good performance, frequent, stable and accurate attributes should be selected. Among the proposed algorithms, the best algorithm *FRI* has the ability to find frequent and stable attributes. Among the proposed types of pairs, the best type of object-attribute pairs *DET* can provide the most accurate attribute information. However the performance of FCA is not so satisfactory. So combining with other algorithm turns the usage of FCA into reality.

4.2 Observations of Suffix Component Based Kind-Of Relation Extraction

Generally speaking, the larger the term list is, the more multi-world terms there should be in the list. Multi-word terms are normally longer strings which tend to have lower frequencies (By Zipf's law, longer text tends to have lower frequency). It is not hard to observe, that the suffixes of Chinese multiword terms, as the headword of the multi-word term, often is the hypernym of the multi-word term. For example, for the multi-word term "大型**计算机**" (mainframe **computer**), its headword as the suffix "**计算机**" (**computer**) is the hypernym of "大型**计算机**". This gives rise to a method to analyze the suffix components of a multi-word term to find hypernym relations as a kind-of relation. The performance of relation extraction based on suffix component analysis has not been sufficiently studies yet. This work attempts to evaluate the top most frequent term components (which are also a term) on the IT term bank for hypernym relation extraction.

The evaluation is divided into two parts. One is to evaluate the ratio (namely coverage) of terms that have their suffix components from a specified set of the suffix components. Another is to evaluate the ratio (namely precision) of pairs of terms and its suffix components having real hypernymy relation.

4.2.1 The evaluation of coverage of suffix components

The evaluation of coverage of suffix components is done in the following steps:

1. The Chinese and English IT bilingual term bank, *CETBank*, is selected for evaluation. It is an IT domain term bank from the Institute of Computational Linguistics, Peking University which is a bilingual lexicon containing 130K Chinese general terms and IT terms, and their English translations. Here only Chinese terms are used.
2. Segment the terms in *CETBank* with *CETBank* itself using the strategy that each term in the bank is segmented using all other shorter yet as long as possible terms in the bank backwardly.
3. Select segmented terms with only two segments from the segmented CETBank into a list (The original *CETBank* has 139,429 Chinese terms. 56,258 of them are selected because they are composed of two segments).
4. Randomize the order of the terms in the list
5. Generate three subsets from the term list.
    a) 1-5,000 terms, including all the terms from the first position to position 5,000 in the list, denoted by *RANGE_1_5000*.
    b) 5,001-10,000 terms, including all the terms from the position 5,001 to position 10,000 in the list, denoted by *RANGE _5001_10000*.
    c) 1-56,258 terms, including all the terms from the list, denoted by *RANGE_1_56258*.
6. Count the frequencies of suffix components in the term set *RANGE_1_5000* and order the suffix components in descending order to obtain the top 2,000 most frequent suffix components.

7. Use the top 2,000 component list to compute the coverage in the term sets *RANGE_1_5000, RANGE _5001_10000* and *RANGE_1_56258,* respectively against the top *N* components in the list. The result is shown in Figure 4.5.

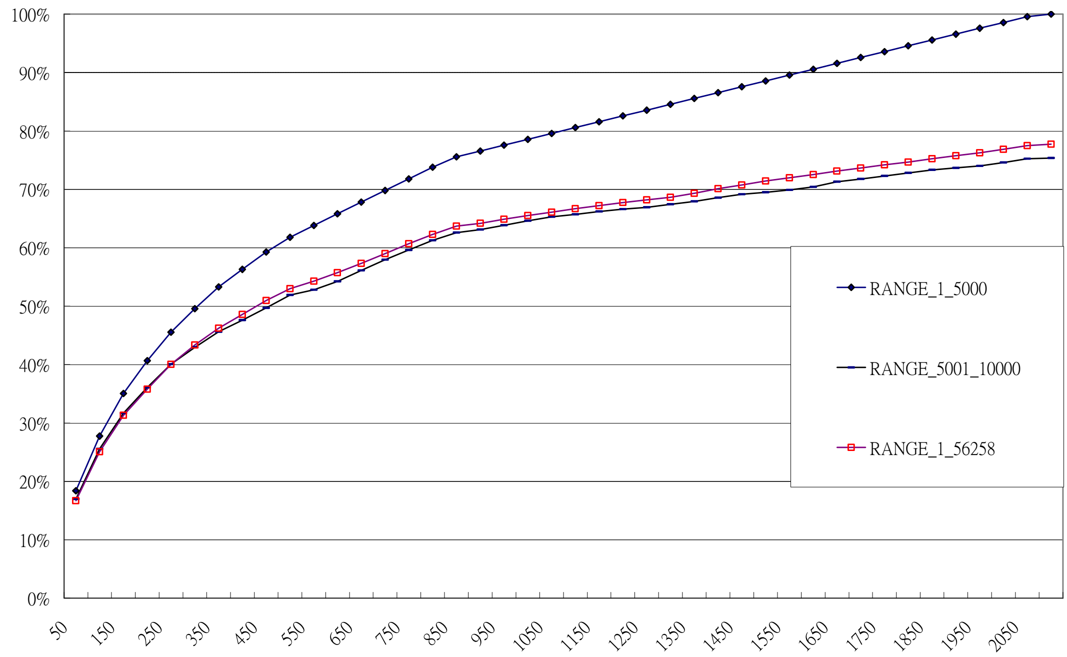


Figure 4.5 Coverage of Suffix Components

Figure 4.5 shows that the ranked suffix components list is very stable and productive so that these curves are close to each other in spite of the different scales of the test sets. This gives hint that a comparative small ranked suffix components list is very useful to term banks of different sizes because of the similar high coverage.

Since the top 1,500 suffixes can already cover more than 60% of bi-word terms, a suffix component based method is worthy of investigation in relation extraction.

4.2.2 The evaluation of precision of suffix components as hypernyms

After randomly sample 40 pairs of terms and their suffix components, about 95% pair has the kind-of relation indeed. This shows that the suffix components can be used very effectively in relation extraction and ontology construction.

The correct cases are numerous. For example, “系统” (system) has hyponyms “操作系统” (operating system) , “能源系统” (energy system) and etc; “计算机” (computer) has hyponyms “大型计算机” (mainframe computer), “微型计算机” (micro-computer), and etc; “程序” (program) has hyponyms “语法分析程序” (syntax analyzer program) , “递归程序” (recursive program) and etc.

The errors in the suffix components assumption are mostly due to the semantic ambiguity of the suffix components itself. For example, the following table shows some examples where “电压表” (voltmeter) is the only exception that the kind-of relation is not existed because the main meaning of 表 (table or meter) in IT domain is “table”, while “电压表” (voltmeter) takes meaning of “meter”.

4.2.3 Conclusion

The above two evaluations show that Chinese suffix components based kind-of relation extraction is simple and reliable with comparatively high coverage and very high precision. These components carry the most productive concepts which should be more important to ontology construction than common terms. In the next section, these terms are indeed be categorized as core terms in its domain.

Chapter 5

Chinese Core Ontology Construction

A core ontology is a mid-level ontology which bridges the gap between an upper ontology and a domain ontology. The concept terms that are used in the core ontology are called core terms which are the most fundamental concepts in a domain [Ji, 2007a]. With the availability of a comprehensive English-Chinese bilingual term bank, a Chinese core term set is first acquired. Automatic Chinese core ontology construction as extension to an upper level ontology can help quickly model domain knowledge. Based on the acquired core terms, graph based core ontology construction algorithm (COCA) is proposed to automatically construct a core ontology as extension to the SUMO upper-level ontology. This algorithm computes the mapping strength from a selected Chinese term to WordNet synset with association to an upper-level SUMO concept in order to link the Chinese term to SUMO in the most appropriate level. The strength is measured using a graph model integrated with several mapping features from multiple information sources. Features include the multiple translations between Chinese core terms and WordNet, the extended string feature and the Part-of-Speech feature.

5.1 Acquisition of Core Terms

Core terms represent the most fundamental concepts in a domain. Generally speaking, the fundamental concepts are frequently used and also the basis to form more complex concepts. Thus, the corresponding core terms must be more commonly used in the domain and are also basic components of longer terms. Consequently, the extraction of core terms should be based on two characteristics of core terms: (1) **highly used and (2) productive to compose longer terms.**

Core terms should have strong ability to form longer terms so that the core ontology can link more domain specific concepts (often represented by longer domain terms) to an upper ontology. An observation made by a study shows that the top 1,500 most productive core terms can serve as suffixes to form more than 50% of the terms in a domain specific term bank (The term bank contains about 130K entries of IT terms)[Chen, 2007b; Chen, 2008b]. And since in most cases a Chinese term acting as the suffix of a longer term is the head words of the longer term (The probability is 93% as reported by [Cui, 2008]), the core ontology constructed using core terms can directly map most of the domain specific concepts to an upper ontology theoretically. The core term extraction algorithm adopted here [Chen, 2009; Chen, 2010] is an improved version of the algorithm proposed by Ji [2007a]. The algorithm sorts the term list in descending order by number of times of the term acting as components of segmented longer terms.

The longer terms are segmented with shorter terms using algorithm “backward maximum dictionary segmentation” which segment a dictionary with the dictionary itself using the strategy that each entry in the dictionary is segmented with all other longest words in the dictionary backwardly. Consequently the entries are definitely segmented into smaller but possible longest components. Here the dictionary is the term bank, and certainly the corpus to be segmented is also the term bank. The frequencies of segmented term components are collected and adopted as their rank value.

Maximum segmentation reduced the intervention of shorter term components to longer term components. For example, “计算机” (computer) embeds “机” (machine). Without maximum segmentation, the frequency of “机” (machine) is much higher than “计算机” (computer) while “computer” is more frequently used term components in IT domain with more distinct meaning. That is why maximum segmentation is used. Appendix B included the list of the top 1500 automatic extracted core terms. From the list, it can be seen that “计算机”(computer) is a core term within top 10. Whereas in Ji’ result [Ji, 2007a], shorter components are preferred. Thus, the shorter Chinese core term “机” takes a much higher rank than the core term “计算机”(computer) , which is much lower in ranking position.

From the definition, core terms should be nouns in a specific domain corresponding to some concepts. However, many morphemes which may serve as components in the domain can be more productive and have stable meaning in composing longer terms. For example, the suffix “式” is only a term component, but it can bring meanings (such as “type”, “style” and other similar meanings) in composing longer terms. This kind of morphemes also is very helpful in constructing domain specific terms for core ontology construction. Thus in the later discussion, core terms are not strictly limited to nouns, they can also include those morphemes with explicit meanings in composing long terms in the domain.

5.2 Design Principles of COCA

The core ontology construction algorithm (COCA) is designed to construct a core ontology for Chinese [Chen, 2007b; Chen, 2008b; Chen, 2009; Chen, 2010]. As there is not much Chinese NLP resources available, COCA is designed to make use of both a comprehensive Chinese-English term bank and also the English WordNet and SUMO where each concept node is mapped to a synset in WordNet already. The main idea of COCA is to map each Chinese core term T_C to the most appropriate synset *SynsetC* in WordNet first. It then makes use of a collection of Chinese core term T_C, the SUMO hierarchy, the SUMO object *Tm* of each synset *SynsetC*, *SynsetC* itself, to build the ontology structure of all the core terms in a domain. As a result, the core ontology is constructed by inheriting the hierarchical structure of SUMO extended by the hypernym structures of WordNet. The main issue, however, is that given a Chinese core term T_C, how to map it to the appropriate synset *S* in the WordNet. That is

$$\arg\max_{S} P(S \mid T_C)$$

To find the appropriate mapping, two levels of ambiguity must be addressed. Firstly, a Chinese core term T_C, as a lexical item, can have multiple translations into English. Secondly, each English lexical item can have multiple senses and thus correspond to different synset in WordNet. Facing these two levels of multiplicity, the main goal of COCA is to find the most appropriate synset in WordNet for a Chinese core term.

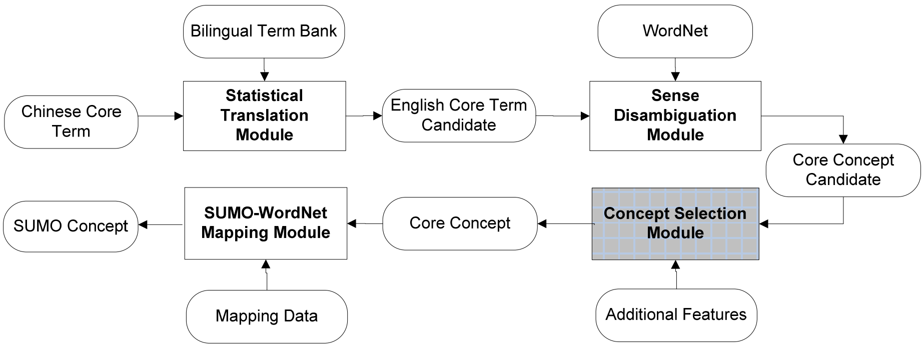


Figure 5.1 the Framework of COCA

Figure 5.1 shows the framework of COCA. It takes a Chinese core term as input and the corresponding core concept and a SUMO concept node as the output. The algorithm first use the bilingual term bank to map Chinese core terms into English core term candidates in the Statistical Translation Module. Second, the English core term candidates are mapped to core concept candidates (namely, synset candidate of Chinese core term) in the Sense Disambiguation Module using WordNet information. Thirdly, a core concept is finally selected among the set of candidates using multiple features (such as hyponym features) in the Concept Selection Module. Finally, using the mapping data in the SUMO-WordNet Mapping Module, the core concept is linked to a SUMO concept as the output.

5.3 Statistical Translation Module

For each Chinese core term T_C, suppose it has a translation T_E in the bilingual term bank. Because a term can have multiple translations, each T_C has a set of translations T_Set_E where $T_E \in T\_Set_E$. The objective of the Statistical Translation Module is to estimate the likelihood of every translation. $P(T_E \mid T_C)$ for all $T\_E \in T\_Set_E$.

For a string s_1 and s_2, if s_1 is a substring of s_2 ($s_1 \subseteq s_2$), Then, s_2 is called the extended string of s_1. For example, the string “software” is the sub-string of the string “public software”. Thus, “public software” is called the extended string of “software”. For a translation pair $<T_C, T_E>$, if there is a another translation pair such that $<T_{C\_e}, T_{E\_e}>$ satisfy the condition that T_{C_e} is the extended string of T_C and T_{E_e} is the extended string of T_E, respectively, the translation pair $<T_{C\_e}, T_{E\_e}>$ is called the extended translation pair of $<T_C, T_E>$. All the T_{E_e} forms an extended translation set for a given T_C, denoted as $ExtT\_Set(T_C)$. In this paper, two heuristics are considered in the probability model: (1) If the total number of extended translation pairs of a translation pair is larger, this translation should be more favorable; (2) If the difference between T_E and T_{E_e} is smaller in term of length, this translation should be more favorable. For example, “网络” has at least two translation “net” and “network”. Since “network” is longer, the weight should be heavier than “net” if they have the same number of extended translation pairs. Then, a weight function $W(T_C, T_E)$ can be expressed as follows:

$$W(T_E \mid T_C) = \sum_{T_{E\_e} \in \text{ExtT\_Set}(T_C)} \frac{len(T_E)}{len(T_{E\_e})} \qquad \text{(E5.1)}$$

where the function *len* returns the length of a string.

The probability of a given Chinese term T_C to be translated into T_E can then be expressed as normalized $W(T_E|T_C)$ given below

$$P(T_E \mid T_C) = \frac{W(T_E \mid T_C)}{\sum_{T_{Ei} \in T\_SetE(T_C)} W(T_{Ei} \mid T_C)} \qquad \text{(E5.2)}$$

5.4 Sense Disambiguation Module

The Sense Disambiguation Module is the second step to map a given T_C to the Synset S through its translation set $T\_SetE(T_C)$. Since a word has probabilities of taking different senses and the sense frequencies of a word are available in WordNet, the mapping probability from a English term T_E to a synset S is given firstly as following,

$$P(S \mid T_E) = \frac{F(T_E, S) + 1}{\sum_{x \in synset(T_E)} (F(T_E, x) + 1)} \qquad \text{(E5.3)}$$

where $F(T_E, S)$ comes from WordNet and gives the frequency of term T_E taking meaning S. For smoothing purpose, "1" is added in formula to each item in the formula E5.3.

Using first-order Markov chain model [Gilks, 1996] with the assumption that T_E is only influenced by T_C and S is only influenced by T_E, the synset path probability $P(S|T_{C,}T_E)$ for a given term T_C to take a particular synset S via an English translation T_E is computed in the following formula,

$$P(S \mid T_{C,} T_E) = P(T_E \mid T_C) * P(S \mid T_E) \qquad \text{(E5.4)}$$

Using the probability function $P(S|T_{C,}T_E)$ is enough to map a core term to a synset meaning. However, the function $P(S|T_E)$ given in formula 5.3 often introduces more errors than function $P(T_E|T_C)$ given in formula 5.2 because the data used in formula 5.4 is taken from WordNet which is a general domain lexicon while the target core concepts are domain specific. To further improve performance, three additional features are introduced in the next section.

5.5 Concept Selection Module

In the Concept Selection Module, COCA takes three more features and methods to further improve performance [Chen, 2007b; Chen, 2008b] including the hypernyms feature, path merging feature, and the POS (part-of-speech) feature. To better improve performance, different combinations of these features are explored. Before coming to these features, a model for integrating independent features is introduced first.

5.5.1 Union Probability of Independent Events

The union probability of independent events comes with the assumption that features are independent events. The formula below is used to compute the union probability of independent events,

$$\underset{x\in E}{U}(p)=\begin{cases}0 & |E|=0\\ p(x)+\underset{y\in E-\{x\}}{U}(p)-p(x)*\underset{y\in E-\{x\}}{U}(p) & |E|>0\end{cases} \qquad \text{(E5.5)}$$

where *E* is an event set, *p(x)* is a probability function which returns the probability of event *x*.

If *E* is an empty set *{}*, then *U(p)=0*; if *E* is set *{x}*, then *U(p)=p(x)*; if E is set *{x, y}*, then *U(p)=p(x)+p(y)-p(x)p(y)*. Obviously function *U* returns the union probability of independent events when |*E*|<=*2*. For all other cases of |*E*| > *2*, since events are independent, given an event *x* in *E*, the event set *Y* = *E* - *{x}* can be considered a single event. In these cases, *y* can be further processed as a new *E*. Then, the final value of *U* can be worked out recursively.

5.5.2 Multiple English Translation Feature

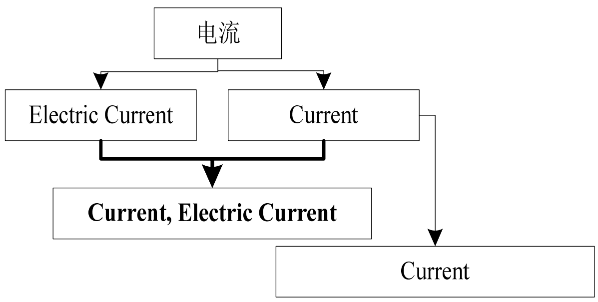


Figure 5.2 Multiple English Translation Feature

Figure 5.2 shows a scenario of mapping of a Chinese term to a synset meaning. The Chinese term "电流"(*current*) has two English translations "*current*" and "*electrical current*". The synset "*current, electrical current*" can be mapped from English term "*Current*" and "*Electrical Current*" respectively.

The mapping probability between a T_C and its synset meaning is denoted by $MP(S|T_C)$ in spite of any English translation adopted. The formula 5.4 of $P(S|T_C, T_E)$ only computes the probability via one English translation T_E even when the synset meaning can be reached via multiple paths from the original Chinese term. To better capture $MP(S|T_C)$, an intuitive method (marked as *MP'*) is to sum up probability of all paths between S and T_C. However, the path with low probability may bring noise. A better method (marked as *MP*) is to select only the path with highest probability via all possible English translation to compute $MP(S|T_C)$. Both the formula are listed as following:

$$MP(S \mid T_C) = \underset{\substack{T_E \in \\ \text{T\_Set}_E(T_C)}}{MAX} P(S \mid T_C, T_E) \quad \text{(E5.6)}$$

$$MP'(S \mid T_C) = \sum_{\substack{T_E \in \\ \text{T\_Set}_E(T_C)}} P(S \mid T_C, T_E) \quad \text{(E5.7)}$$

Obviously, $\sum_S MP(S \mid T_C)$ is less than one whereas $\sum_S MP'(S \mid T_C)$ is equal to one. Since the formula *MP* is only used to rank candidate synset S for a given Chinese term T_C, it not necessary to insist on $\sum_S MP(S \mid T_C) \equiv 1$.

5.5.3 Hyponym Feature

Given a core term, the first kind of additional features comes from all the extended strings which use the core term as headword. Figure 5.3 is an example of a core term "计算机" (computer) which are applied to all the extended string to map to a better concept using the hyponym relations. The Chinese term "计算机" (computer) can be mapped to English term "computer" and "calculator". As the English term computer, when serving as a headword, has more extended terms (extended string) such as "大型计算机" (Mainframe Computer) whose translation is a hyponym of computer. Thus, the translation "computer" suits better than "calculator".

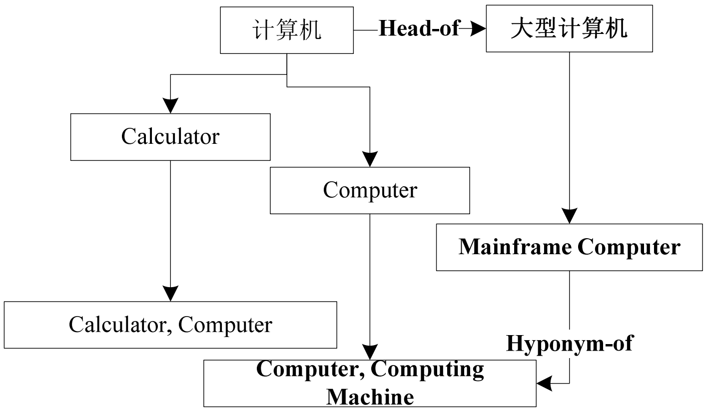


Figure 5.3 Using Hyponym in Domain

According to work in [Cui, 2008], about 93% extended string has their suffix term as the headword. Taking this fact as an assumption, the set of extended terms using T_C as suffix is obtained, using the function *Suffix_Ext*(T_C). A new measure Hyponym Probability, denoted by $HP(S|T_C)$, is proposed to exploit information of hyponyms of T_C and integrate using the function *U*.

$$HP(S \mid T_C) = \underset{t \in \text{Suffix\_Ext}(T_C)}{U} \left(\sum_{h \in hyponym(S)} \frac{len(T_C)}{len(t)} MP(h \mid t) \right) \quad \text{(E5.8)}$$

where function *hyponym(S)* returns all the direct or indirect hyponyms of synset *S*, $len(T_C)/len(t)$ is applied to compute according to the similarity between a core term T_C and its extended term *t*.

5.5.4 Feature of Terms Sharing Suffixes

If term t_1 and t_2 share the same suffix, then t_1 is called the sibling term of t_2 and t_2 is also the sibling term of t_1. Obviously sibling terms are terms sharing suffixes. Chinese sibling terms sharing the same suffix often have their meanings (synset) sharing the same hypernym. For example, “驱动器” (drive, a kind of device) has many sibling terms such as “计算器”(calculator), “传感器”(sensor), and so on. They all share the same hypernym “device” and the same suffix “器”(device) in IT domain. This observation shows that sibling terms can help improve the mapping for a sibling term to take the right meaning (synset).

Feature of terms sharing suffixes [Chen, 2009] [Chen, 2010] is an extension of hyponym feature. Hyponym feature exploits only information of longer terms, while feature of terms sharing suffixes can exploits information of longer terms and shorter terms once they sharing the same suffix.

The problem remained is how to find and use the prominent hypernym while terms can be mapped to different hypernyms. The prominence of hypernyms can be considered in two aspects. Firstly, the more sibling terms a hypernym has, the more important the hypernym is. Secondly, a more general hypernym has less distinguishing power. Thus among choices, the hypernym which is the most specific should be more important. Based on these two hypotheses, a suffix feature is proposed which exploits sibling features to improve performance in the Concept Selection Module.

For each Chinese core term T_C, the processing of this feature are described as following

INPUT:

1) the shared suffix T_H and the terms sharing as suffix
2) the mapping weight from Chinese term to synset candidate with other feature
3) the hierarchical synset tree by hyponym relation simplified from WordNet

OUTPUT:

The adjusted mapping weight from Chinese term to synset candidate

PROCEDURE:

1. Compute the **W**eight of each **S**hared **H**ypernym (SHW for abbreviation) given the longest suffix T_H using the following formula

$$SHW(S_H \mid T_H) = \sum_{T_C \in ext(T_H)} \sum_{\substack{s_i \in synset(T_C) \wedge \\ s_i \in hyponym(S_H))}} \frac{dep(S_H)}{dep(s_i)} MP(s_i \mid T_C) \qquad \text{(E5.9)}$$

where the function $ext(T_H)$ returns all the extended string of T_H, $dep(s)$ returns depth of a synset s to its corresponding root synset (namely, the most abstract synset) in WordNet under hypernym relations, $synset(T_C)$ returns the synsets that can be mapped from a Chinese term T_C.

The above formula utilizes the two hypotheses on the sibling feature. For the first hypothesis, the more sibling terms T_C a hypernym has, the more important the hypernym is reflected by the summation on the number of hypernyms. For the second hypothesis, a more general hypernym S_H has less distinguishing power (less important) because the coefficiency $dep(S_H)/dep(S_i)$ is smaller value as a weight to $COCA(S_i \mid T_C)$.

2. Compute **W**eight **under H**ypernym (abbreviated as *WUH*) for each synset S of a core term T_C with the longest suffix T_H implied. The following formula makes use of the maximum weight of hypernyms of synset S to compute the sibling weight of T_C.

$$WUH(S \mid T_C) = WUH(S \mid T_C, T_H) = \frac{\underset{S_i \in hpr(S)}{MAX}\ SHW(S_i \mid T_H)}{\sum_{S_j \in synset(T_C)} \underset{S_k \in hpr(S_j)}{MAX}\ SHW(S_k \mid T_H)} \quad \text{(E5.10)}$$

where the function *hpr(s)* returns all the hypernyms of a synset *s,* function *MAX* get the maximum value among a set.

The above formula calculates the normalized mapping weight from core term T_C to the synset S with the information of the most important hypernyms. The most important hypernym is the hypernym with maximum *SHW* weight. The normalized value is between 0 and 1 because the numerator is always included by the denominator of the formula.

5.5.5 Part-of-Speech Feature

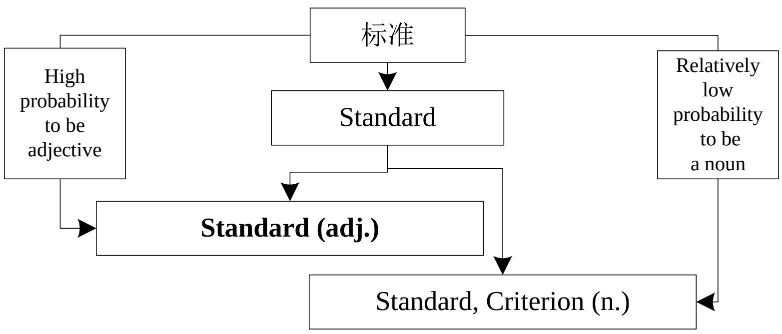


Figure 5.4 Using Part-of-Speech

Figure 5.4 shows a case of a better mapping when POS tag information is used. The term “标准” (standard) can be mapped to either the synset “Standard, Criterion” or

"Standard". From the observation in the extended string of "standard", it can be found that "标准" (standard) has relatively low probability to be a noun while has a high probability to be an adjective. Therefore, the noun mapping should be selected. This shows that POS tagging information is also useful to select the correct meaning (or synset). However, due to the fact that neither the core term list nor its English translation has PoS tagging, a heuristics must be used to estimate the PoS of the Chinese core terms so its estimated PoS can match the synset with the same PoS. The estimation of PoS of *Tc* is done by searching *Tc* in all its extended Chinese terms. A simple heuristics is that it is more likely to be an adjective if it occurs in the beginning as a prefix, a verb if in the middle, and a noun at the end as a suffix. Then, a simple function $freq\_pos(T_C, tag)$ is then developed which can obtain the frequency of T_C taking *tag* as its POS tag by searching through the term bank ***CETBank.*** Then, the probability with POS tagging being considered is denoted as $OP(S \mid Tc)$ which takes the following form:

$$OP(S \mid T_C) = \frac{freq\_pos(T_C, pos(S))}{\sum_{po \in \{noun, verb, adjective\}} freq\_pos(T_C, po)} \qquad \text{(E5.11)}$$

where the function pos(S) returns the POS tag of a synset S.

5.5.6 Feature Integration for Concept Selection

Finally, the probability for the mapping between T_C, and *S*, denoted by $COCA(S|T_C)$, is worked out by integrating all the features as independent events in the formula of union probability. The integration formula is listed below:

$$COCA(S \mid T_C) = \bigcup_{x \in \{MP(S|T_C), HP(S|T_C), OP(S|T_C), WUH(S|T_C)\}} (x) \qquad \text{(E5.12)}$$

5.6 Evaluation

The set of 1,500 ranked core terms extracted by a core term extract algorithm is used [Chen, 2009] [Chen, 2010] in the COCA algorithm [Chen, 2008b]. The evaluation is conducted manually on the list of 400 core terms, which according statistical theory, should give no more than 5% of margin of error [Scheuren, 2004], a sampling size 400 is needed. The samples are evenly selected from the items of the whole 1500 list to complete the evaluation.

In the evaluation, the following sets of resources are used: (1) a Chinese and English bilingual term bank, ***CETBank***, an IT domain term bank from Institute of Computational Linguistics, Peking University which is a bilingual lexicon containing 130K Chinese general terms and IT terms, and their English translations; (2) WordNet 1.6; (3) the mappings between WordNet 1.6 and SUMO nodes [Niles, 2003].

The domain specific core term list, used in this work, referred to as ***ITCTerm***, is the same as that used in [Chen, 2007]. ***ITCTerm*** is extracted from the term bank ***CETBank*** by a simple segmentation algorithm which forces every entry *Ti* in ***CETBank*** be segmented so as to see what are the smaller component words/terms used to form *Ti* [Ji, 2007] [Chen, 2007]. As a core term should not only be domain specific but also have strong ability to form other words, the words that are most frequently used as components of other domain terms in ***CETBank*** are selected as core terms.

The output of ***COCA*** contains pairs of <*Chinese Core Term, Best Synset*> and <*Chinese Core Term, SUMO Concept Mapped from Best Synset*>. The evaluation are conducted on the 400 core terms by checking if their mappings to synset and SUMO concept are the best match among all synset candidates of that Chinese core term.

To evaluate the effect of different features used in the algorithm, comparisons are made with either individual features or combinations of features. The algorithm labelled **B** is the base line algorithm which selects the synset with the highest frequency recorded in WordNet for a core term. Base line algorithm is proposed to see the performance that can be achieved using only general domain information. The algorithm labelled **S** uses sense disambiguation using formula 5.4 only. The labels **1**, **2** and **3** represent the three features defined in Section 5.5 for Multiple English translation, Hyponyms, and Part of Speech, respectively. The feature of Multiple English translation has two measuring methods, the first method *MP* is labelled as **1** and the second method *MP'* is labelled as **1'**. The label **4** stands for the feature of terms sharing suffixes. The evaluation results are listed in a bar chart shown in Figure 5.5

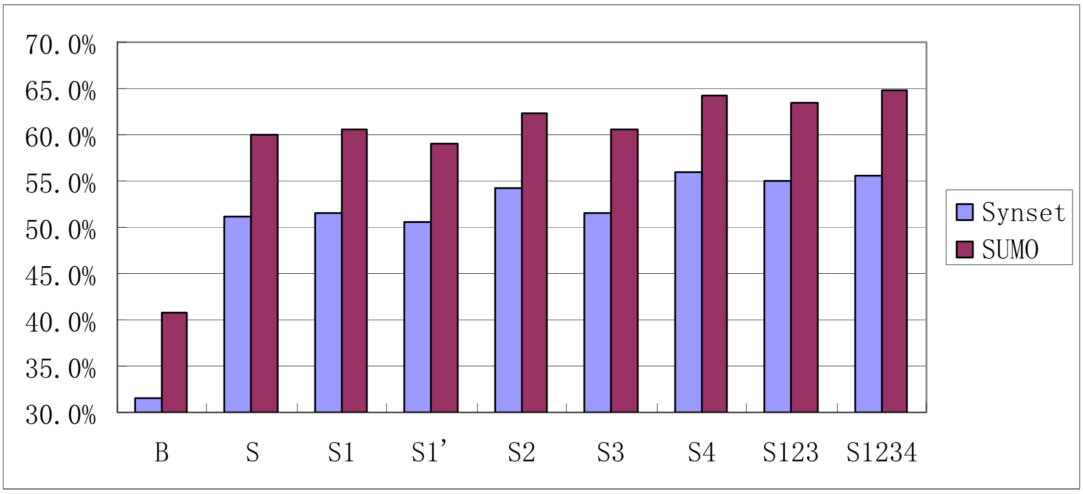


Figure 5.5 Performances in Bar Chart for Different Combination of Features

From the figure above, it can be seen that the combined method **S4** using the feature of terms sharing suffixes achieved the most significant improvement both in terms of synset and SUMO. The improvement by using only the feature of terms sharing suffixes can improve the performance by 78.9% compared to the baseline for synset correctness. Even though the mapping to synset by **S4** gives the best performance, it is still the combination of all features labeled by **S1234** that gives the best performance on SUMO concept selection which is the main objective of the algorithm. By looking at the mapping to SUMO in more details, almost every single feature can improve performance except **S1'** which proves that feature should be carefully used with noise reduced. However the combined effect still shows that the feature of terms sharing suffixes is the most dominant. The evaluation proved the success of the feature of terms sharing suffixes and further prospective improvement on the feature combination.

An error analysis shows that errors are mainly originated from three sources. The first source is the introduction of general lexicon "WordNet". For example, "电阻"（resistance）is always wrongly translated into "resistance, opposition" which means "he action of opposing something that you disapprove or disagree with”. While in IT domain, the correct meaning should be “a material's opposition to the flow of electric current; measured in ohms”. The corresponding SUMO concept is UnitOfMeasure (the concept path in SUMO is “/Entity/Abstract/Quantity/PhysicalQuantity/UnitOfMeasure”). The feature of terms sharing suffixes solved the part of the problem. For example, “驱动程序” (driver) is wrongly translated into “car driver” (司机),but after applying the feature of terms sharing same suffixes the right synset “driver program” (驱动程序) is selected. The second source is that some of the core terms are not existed in WordNet such as multi-access (“多路存取”) and etc. It is found that 4% of the core terms can’t find their corresponding meaning in WordNet. Consequently these core terms can’t be mapped to SUMO concept automatically. The third source is the translation errors caused by lacking of context. The bilingual term bank is only a dictionary within which contextual information is missing. This made the difficulty in deciding which meaning is the best in an exact domain. The following graph gives an example of a fragment of automatically constructed Chinese core ontology in IT domain. The top level concept is from SUMO, and others are inherited core concepts.

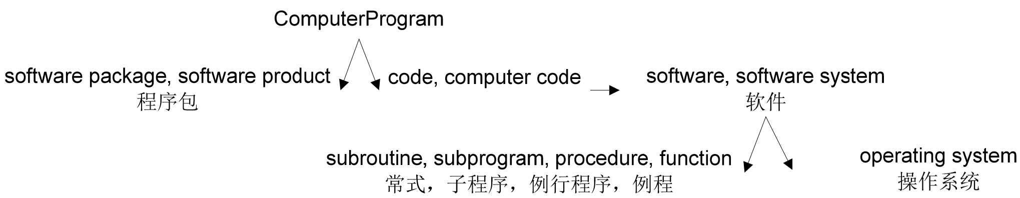


Figure 5.6 A Fragment of Automatically Constructed Chinese Core Ontology

5.7 Conclusion

The algorithm COCA is proved to be effective in automatic Chinese core ontology construction. Among all features of the algorithms, the feature of terms sharing suffixes with the basic statistical module and sense disambiguation module achieved the best performance in nearly all evaluations. In the future, more information sources will be integrated and better combination method will be tackled to achieve better performance.

Chapter 6

Improving Chinese Core Ontology Construction with Bilingual WordNet

The COCA algorithm presented in Chapter 5 identifies the best matched synset using a Chinese to English translation as the intermediary agent. However, the translation in the process brings translation ambiguity problems as already been shown in Chapter 5. Intuitively, the problem no longer exists if a Chinese lexicon ontology is applied. In other words, a bilingual WordNet would eliminate translation ambiguity because no additional ambiguity is introduced due to translation.

This chapter first attempts to investigate the feasibility to use a Chinese-English lexical ontological resource, the so called Sinica BOW, to help building the Chinese IT core ontology [Huang, 2004]. Sinica BOW as a Chinese lexical ontology contains English translation, links to WordNet and thus SUMO, and other properties of these Chinese lexical items. After the analysis of the information in Sinica BOW, this chapter then, try to make use of the useful information in Sinica BOW as an additional resource to improve the the performance of COCA.

6.1 Direct Use of Sinica BOW

The full name of Sina BOW is the Academia Sinica Bilingual Ontological WordNet (abbreviated as S*BOW*) which was developed by the Academia Sinica in Taiwan [Huang, 2004]. *SBOW* is a high quality, large scale, manually created bilingual WordNet (Chinese-English) which has integrated three resources: WordNet, English-Chinese Translation

Equivalents Database (ECTED) and the Suggested Upper Merged Ontology (SUMO). For *SBOW*, the native language is Chinese, the translated language is English.

The scale of *SBOW* is large. After pre-processing, *SBOW* contains about 100K Chinese words, 150K pairs of Chinese word and synset mapping in WordNet, 266K triples of Chinese word, English word and synset mapping. Besides, there are about 122K English words, 174K pairs of English word and synset mapping, and 253K pairs of Chinese word and English word in *SBOW*. Chinese resource with scale containing mapping to WordNet makes a good candidate for our purpose.

6.1.1 Algorithm Design

To evaluate how *SBOW* can be used to help with core ontology construction, two algorithms are proposed here to see how the information in *SBOW* can be used for core ontology construction. The first algorithm is an intuitive method which directly finds the best synset mapping for core terms using *SBOW*. This method can show the performance of constructing a Chinese domain core ontology directly from a general domain ontology. The second algorithm exploits the triplelets <Chinese term, English term, synset node> in *SBOW* to obtain mapping into *SUMO*.

This **d**irect **m**apping **m**ethod (*DMM*) is a simple and intuitive algorithm which directly finds the best synset mapping for core terms in *SBOW*. Although there is no ambiguity from translated terms, there is still ambiguity from the term in Chinese to synset mapping.

DMM select the most frequent synset mapping for each Chinese core term. The frequency of synset node is summed up using the following formula

$$F(S) = \sum_{T_E \in EN(S)} F(T_E, S) \qquad \text{(E13)}$$

where $F(T_E, S)$ comes from WordNet and gives the frequency of English term T_E taking the synset node *S*, *EN(S)* returns all the English words that can take the synset node *S.* Then DMM algorithm can be formulated as follows

$$DMM\ (Tc) = \mathop{\arg\max}_{s \in \{s' | <Tc, s'> \in BO2\}} F(s) \qquad \text{(E}\mathbf{14}\text{)}$$

where *BO2* is a set of 2-tuples <*Chinese term, synset node*> in *SBOW*.

Since the triplelets <*Chinese Term, English Term, Synset Node*> in SBOW contains more information, and are thus more useful than two-tuples <*Chinese Term, Synset Node*>, this algorithm "**m**apping with **a**pproval of **t**erm **b**ank" (*MATB*) identifies the Chinese core term and then find its synset mapping through *SBOW*. In the mean time, it has to make sure that the Chinese term and the English term also exist in a domain specific bilingual term bank. Namely, the translated English term should be the translation of the core term in the term bank and have the candidate synset mapping in WordNet. *MATB* can also be described by the following formula

$$MATB\ (Tc) = \mathop{\arg\max}_{s \in \{s' | \exists Te (<Tc, Te> \in TB \ \wedge \ <Tc, Te, s'> \in BO3)\}} F(s) \qquad \text{(E}\mathbf{15}\text{)}$$

where *BO3* is a set of triplelets <*Chinese term, English term, synset node*> in *SBOW*, *TB* is a set of 2-tuples <*Chinese term, English term*> in the bilingual term bank.

All together, there are three algorithm parameters for the *COCA_BW* algorithm including probability β , minimal length *ML* and the Boolean indicator *BE.*

6.1.2 Preprocessing of SBOW

. The Chinese entries in *SBOW* are manually created as a Chinese lexicon ontology and they tend to be formal and accompanied by many modifiers. For this work, however, many of the modifiers are not used and are thus omitted. Since core term list is an extracted list, a preprocessing is necessary before the matching between core terms and entries in *SBOW*. Each Chinese lexical entry in *SBOW* is preprocessed according to the following rules

a) Since core terms are in simplified Chinese while *SBOW* is in traditional Chinese, the entries in *SBOW* is converted into simplified Chinese using the open source Perl module “Encode::HanConvert”[[6]]. The conversion done by this module not only includes character level conversion, but also word level conversion for some frequently used terms.

b) Delete all the bracketed explanations in the lexical entry

Many words or phrases come with bracketed explanations in styles such as “（domain or other constraint description）words or phrases”. For example, “(化)分子”((Chemistry)Molecule) should be recovered into “分子(Molecule)”.

c) Change “与 … 相关的” and “和…相关的” (… related) into “…”

In Chinese, “something related” is expressed in the ways such as “与 something 相关的”, “和 something 相关的” or directly “something”. Very commonly the simplest form “something” is used so that the core term is “something” directly. While lexicon entries often use the other two longer forms to make the entries more accurate. To better find core term in a lexicon, this conversion becomes a necessity.

d) Delete suffixes “的” and “地” in the end of lexical entries if available

Formally Chinese adjectives and adverbs accompany with “的” (similar to the adjective suffixes “-ful”, “-tive” in English) and “地” (similar to the adverb suffix “–ly” in English) as suffix, respectively. But as a lexical item, it is better to remove the suffix.

6.1.3 Performance Evaluation of DMM and MATB

The term bank and core term list used in the evaluation is the same as those in Chapter 5 so as to make the evaluation to COCA meaningful. The performance of *DMM* and *MATB* are

---

[6] This perl module is developed by Tang Feng, hosted in http://search.cpan.org/~audreyt/Encode-HanConvert-0.35/lib/Encode/HanConvert.pm. The conversion table used in this module comes from various sources, including Lingua::ZH::HanConvert by David Chan, hc by Ricky Yeung & Fung F. Lee, and Doggy Chinese Big5-GB Conversion Master from Doggy Digital Creative Inc. (http://www.miniasp.com/), Rei-Li Chen (rexchen), Unicode consortium's Unicode Character Database (http://www.unicode.org/ucd/), as well as mappings used in Microsoft Word 2000, Far East edition.

conducted using *SBOW* and on the standard answer set and testing set as used in COCA. COCA is also listed for comparison purpose. The results are shown in the following table.

	Precision on Synset Mapping	Precision on SUMO Concept
DMM	32.1%	50.7%
MATB	39.4%	52.3%
COCA	**56.0%**	**64.0%**

Table 6.1 Precision of *DMM*, *MATB* and *COCA*

It can be seen that using SBOW as a single resource, its performance is not as good as the *COCA* algorithm discussed in Chapter 5. The reason is that *SBOW* is a general domain lexicon ontology where domain specific knowledge is not comprehensive enough, while the IT term bank contains more comprehensive domain information. It is not hard to see that *DMM* performs worse than *MATB* because it is not using sufficient information whereas *MATB* the information of bilingual IT term bank is added. This evaluation also indicates that COCA approach of using bilingual term bank with mapping into synset of WordNet and thus SUMO is valuable and performs better. Furthermore, it shows that the use of *SBOW* needs better strategy and a careful analysis. A further analysis of matching correct synset meanings for each Chinese core term in *SBOW* is then conducted to explore the deeper reason behind.

6.1.4 Analysis of Matching Types

There are four types of matching cases in finding a correct synset node for each Chinese core term in *SBOW*. The worst case is that the Chinese core term just cannot be found in *BOW* (named as *NoMatch*, abbreviated as *NM*). The best case is that the Chinese core term can be found in *BOW* and that the set of pairs of the core term and its corresponding synset node in the result is the same to the corresponding set in BOW (named as *TotallyMatched*, abbreviated as *TM*). If a Chinese core term in the result can be found in SBOW, the case "*Partially Matched Without Correct Answer*" (abbreviated as PMNA) refers to that the correct synset node for the core term cannot be found in SBOW, the case "*Partially Matched*

With Correct Answer" (abbreviated as PMA) refers to that the correct synset node for the core term can be found in SBOW but there are still some other candidate synset nodes not in SBOW. For a quick overview on these matching cases, the following table and figure is presented,

Type	Abbreviation	Number
NoMatch	*NM*	92
PartiallyMatchedWithoutCorrectAnswer	*PMNA*	154
PartiallyMatchedWithCorrectAnswer	*PMA*	143
TotallyMatched	*TM*	11

Table 6.1 Four Types of Matching cases between *SBOW* and 400 answers

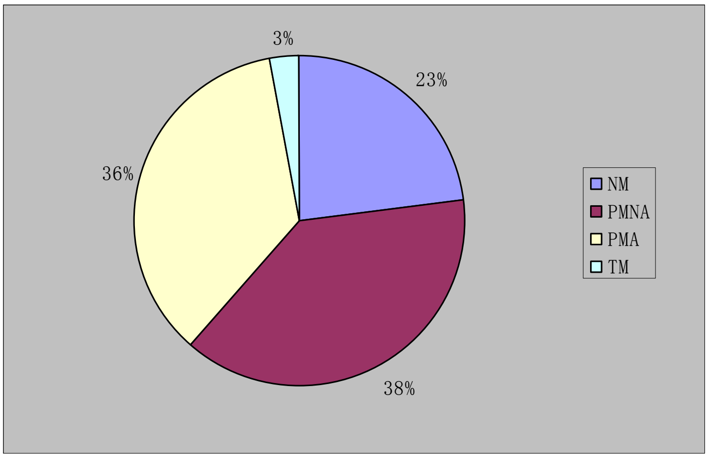


Figure 6.1 Four Types of Matching cases between *SBOW* and 400 answers

From the above table and diagram, it can be seen that 23% terms (92 out of 400) cannot be found in *SBOW*, 61.8% terms (92+154 out of 400) never get their correct synset meaning from *SBOW* which means the coverage of SBOW for the core term set is less than 40%. So any method only based on *SBOW* can never reach the 40% accuracy and is inferior than COCA because COCA has achieved more than 56% in accuracy. The analysis shows that the COCA algorithm with bilingual domain term bank and English WordNet is much more effective in constructing domain specific core ontology than using *SBOW* alone. The above data only shows the deficiency of BOW. More detailed analysis is given below.

In 400 Answers			***In SBOW***		
Chinese	*English*	*Synset ID*	*Chinese*	*English*	*Synset ID*
兼容性	compatibility	03707482n	一致	compatibility	05616177n
			一致	compatibility	03707482n
			调和	compatibility	03707482n
带宽	Bandwidth	09758629n	宽频	bandwidth	09758629n
			带幅	bandwidth	09758629n

Table 6.2 *NoMatch* Cases in Matching between *SBOW* and 400 answers

Table 6.2 shows examples of some "*NoMatch"* cases. Reasons that most the core terms cannot be mapped can be categorized intuitively into two cases.

1. Less coverage by SBOW for domain specific terms.

   E.g.: the core term in the bilingual term bank, "兼容性" (compatibility), is not in *SBOW* so it cannot be found using SBOW only. Yet, COCA is able to find it through its translation "compatibility" via WordNet.

2. The difference in IT terminologies in Mainland, Hong Kong, and Taiwan

   Such as "光盘 – 光碟" (Optical Disc), "软件 – 软体" (software).

Then "*PartiallyMatchedWithoutCorrectAnswer"* cases are presented in Table 6.3 and some intuitive reasons are induced based on it.

In 400 Answers			***In*** SBOW		
Chinese	*English*	*Synset ID*	*Chinese*	*English*	*Synset ID*
集合	Set	**05990251n**	系列, 组, 一	Set	**05990251n**
			集合	Rally	00946962v
				Rally	06189096n
				congregating	00799654n
				assemblage	00798100n
				assemblage	02828573n
				congeries	06225016n
				conglomeration	06225016n

				convoke	00538841v
				assembly	06071059n
				assembly	00798100n
				Come_up	00946962v
				beat_up	00949975v
				Drum_up	00949975v
				Rally	00946962v

Table 6.3 "*Partially Matched Without Correct Answer"* Cases in Matching

From the examples given in Table 6.3, it can be seen that the Chinese core terms and corresponding synset mapping are not linked as a pair in *SBOW* even if both of them are present. For example, the Chinese core term "集合" and synset mapping "Set, 系列, 组, 一" (with synset id 05990251n) are both existed and have the same meaning in *SBOW*, while "集合" is not listed in this synsets. And this kind of deficiency cases is worse than "*NoMatch*" cases because its forms 38% of all the cases, much higher than the 23% for the "*NoMatch*" case.

6.2 Improved COCA Using SBOW

It is obvious from Section 6.1 that *SBOW*, if used as a single resource for core ontology construction, would suffer from both the coverage problem and broken links problem. Thus, is natural to consider using *SBOW* as an additional knowledge source to extend *COCA* knowledge for a better performance.

6.2.1 Design of the Improved COCA

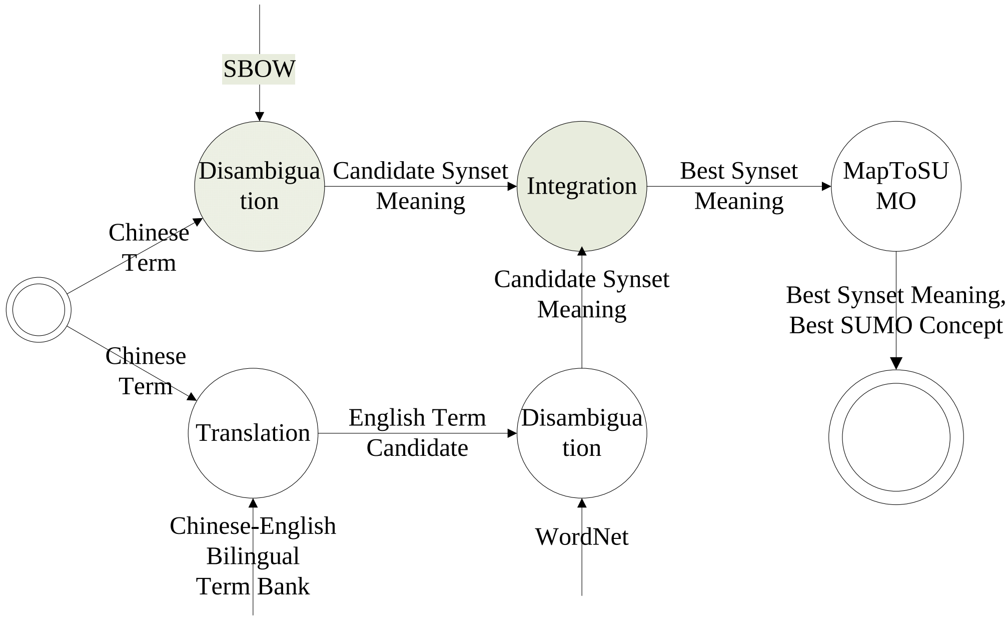


Figure 6.1 Framework o Improving *COCA* with Bilingual WordNet

COCA provides a good framework to integrate multiple evidences as well as evidences from other bilingual resources. Figure 6.1 shows the framework of how COCA can incorporate SBOW. It can be seen that the translation process, the disambiguation process with WordNet, the integration process and the *MapToSUMO* process (mapping to the upper ontology - SUMO) are all in *COCA* already. *SBOW* is used in the feature integration process of COCA. The use of SBOW is applied to COCA in re-weighting of the path strength with the following algorithm:

$$COCA\_BW(S \mid T_C) = \begin{cases} \underset{\{\beta, COCA(S|T_C)\}}{U} & CB(S|T_C) = true \\ COCA(S \mid T_C) & CB(S|T_C) = false \end{cases} \qquad \text{(E16)}$$

where the function U is the union probability of independent events, T_C represents Chinese core term, S represents a synset node, the parameter β is a configurable constant which stands for the probability of taking path in SBOW if the path exists, $CB(S|T_C)$ is the condition to use the data from *SBOW*. The condition is needed because some of the *SBOW* data cannot be used. From the evaluation of the 400 cases, two observations are made: (1) The better English translation of the Chinese term, the higher the performance, and (2) The longer the Chinese term, the higher the performance. Based on these two observations, two constraints for applying *BOW* data are proposed to improve performance. The first constraint is that the length of Chinese term is equal to or longer than a minimal length *ML*. Another optional constraint *BE* is that only the best English translation is adopted for the Chinese term to take the synset mapping which is then written as the two conditions:

$$CB(S \mid T_C) = \begin{cases} len(T_C) >= ML \text{ and } \exists T_E(S \in synset(T_E) \wedge \\ T_E = \arg\max_{T_E} P(T_E \mid T_C)) & BE = true \\ len(T_C) >= ML & BE = false \end{cases} \quad \text{(E17)}$$

where function $len(T_C)$ returns the length of the Chinese Term T_C, *ML* is a configurable algorithm parameter which specifies a minimal length, T_E refers to an English term, function $synset(T_E)$ returns the set of synset mappings of the English term T_E, the configurable algorithm parameter *BE* is whether optional best English translation constraints should be applied or not.

All together, there are three algorithm parameters for the *COCA_BW* algorithm including probability β , minimal length *ML* and the Boolean indicator *BE*.

6.2.2 Performance of the Improved COCA with SBOW

From the previous description of the integration algorithm, three configurable parameters *ML(*minimal length), β and *BE* (best English constraint) are introduced. Several different configurations are evaluated to see how better performance can be achieved. The notations and detailed explanation of the parameters are given in the following table.

Notation	Name	Description
ML	Minimal Length	
β	Possibility	PXX represents a possibility "0.XX". For example, P99 means that the possibility is 0.99
BE	Best English Constraint	That the map between the synset meaning and Chinese term should include and only include the best English translation for the Chinese term

Table 6.4 Notations of Configurable Parameters

Three groups of experiments are conducted to explore different configuration of the three parameters, respectively. Once the best configuration of a parameter is confirmed, it is kept in the next group of experiments. The detailed result is reported in Table 6.5.

ML? β *5*		*ML2* β *?*		*ML2* β *5BE?*	
ML1 β *5*	54.5%	*ML 2* β *3BE*	58.5%	*ML2* β *5*	55.3%
ML 2 β *5*	**55.3%**	*ML 2* β *4BE*	58.3%	*ML2* β *5BE*	**58.5%**
ML 3 β *5*	56.3%	***ML 2*** $\boldsymbol{\beta}$ ***5BE***	**58.5%**		
		ML2 β *6BE*	57.8%		
		ML2 β *7BE*	57.5%		

Table 6.5 Three Groups of Evaluation Result on Three Parameters Respectively

As shown in Table 6.5, the best performance is 58.5% under the configuration of a minimal length 2, β equal to 0.5, and with best English constraint. The performance result shows that the exploitation of two observations made consistent and significant improvements for *SBOW* to use in *COCA*.

To learn more detailed reason behind the performance of adopting longer minimal length, the ratio of matching type against the length is given in Table 6.6.

Length / Type	1	2	3
RatioOfPartialMatchedWithCorrectAnswer	38.7%	54.7%	81.3%

Table 6.6 Ratio of Matching Type against the Length

It is obvious from the table above that the longer the Chinese term, the higher the performance.

6.3 Conclusion

Existing bilingual ontology is a general domain ontology which is easy to import from it to build domain ontology quickly. However from the evaluations, it can be seem that without domain information and a good algorithm, the ontology automatically imported from bilingual ontology is very poor and hard to compare with *COCA* introduced in previous chapter. If the bilingual ontology is used under best translation constraint and with longer term length, it is still a useful complement to *COCA*.

Chapter 7

Conclusions and Future Work

Automatic domain ontology construction is complex and daunting task. Firstly, the conceptualization needs terms to carry concepts, relations to link concepts. Secondly, resources need to build a domain ontology can be extraneous, thus automatic construction methods must be exploited. The construction of domain ontology using a purely bottom up approach is extremely difficult because of data sparseness issue of domain corpus. Good quality and comprehensive domain corpus in Chinese is not available at current time for domain ontology construction. Thus, this makes it even more important to build a core ontology in a domain which can help to map a domain ontology to an upper-level ontology.

This work contributes to domain ontology construction in three main parts: (1) the development of an effective term extraction algorithm for Chinese based on the measures of both linguistic unit and domain specificity; (2) the design and implementation of a relation extraction algorithm for Chinese using the FCA method to provide a framework for domain specific ontology construction using a bottom-up approach, and (3) the construction and mapping of a Chinese core ontology to the upper ontology with limited resources to build foundational mid-level ontology which be help bridge the gap and link domain specific ontology to upper-level ontologies.

In all these three parts, internal information and contextual information are applied to improve performance. In terminology extraction, the internal measure of association between components and external information of neighboring terms are integrated. Kind-of-relations are extracted using contextual associated neighboring attributes and internal components. In Core ontology construction, contextual features such as pair frequency and internal features such as suffix components are integrated under a uniform framework. All these integrated evidences are incorporated in this work and are proved to achieve better performance than what is in the literature. The algorithm *UREval* in terminology extraction exploited four types of information and are integrated together with carefully designed formula to achieve very good performance in terms of precision (nearly 100%) in the evaluation of the top 5,000 candidates in character based bi-gram term unit extraction.

From the engineering point of view, core ontology is a good entry point for human intervention to domain ontology construction because it contains the most commonly used and productive domain concepts with mappings to upper ontology. A good core ontology can greatly reduce labor intensive work and make domain concepts more usable to interoperate with other ontology. The introducing of core ontology with a solution to actually automatic construct a core ontology with a large scale Chinese IT domain are another contribution. Moreover, the design of COCA provides a uniform framework to use and integrate multiple features to identify domain specific terms and map them to upper-level ontology. Evaluations show that the use of suffix as a feature for multi-word terms significantly improves the performance with the best record among all.

Although automatic solutions are developed in three parts, the performances of the algorithms developed in the three parts all have room for improvement. In fact, their performance in current form still cannot fulfill realistic comprehensiveness requirement as complete automated systems. In other words, to build a domain ontology with reasonably good precision and coverage, and alignment to upper level ontology for a good general conceptualization framework, much manual revision and verification efforts are still needed, to make the final ontology useful.

Four main areas of work can be conducted in the future. Firstly, the gap between terminology extraction and relation extraction, relation extraction and core ontology construction should be tackled to attempt to fully automate the process of domain ontology construction. Secondly better terminology extraction algorithms should be studied especially to those terms with very low frequency. Attempts have been made at the completion time of this work to make use of features of term delimiters for terminology extraction which is a good direction to go [Yang, 2008]. Thirdly, the performance of relation extraction should be further improved and more relation types such as hyponym, usage, ownership and etc. will be explored. Finally, more resources, especially internet resources, such as those manually annotated semantic rich resources such as Wikipedia, can be exploited to assist in the construction of core ontology.

Bibliography

Bisson, G., Nedellec, C., & Canamero, L. 2000. Designing clustering methods for ontology building - The Mo'K work-bench. In proceedings of the ECAI Ontology Learning Workshop, pp.13-19, Seatle, USA, Aug. 22, 2000.

Borst, P. 1997. Construction of Engineering Ontologies for Knowledge Sharing and Reuse, In PhD thesis, Universiteit Twente, Enschede, NL.

Bray, T. and Paoli, J. and Sperberg-McQueen, C.M. and Maler, E. and Yergeau, F. 2000. Extensible markup language (XML) 1.0. In W3C Recommendation, vol. 6.

Buitelaar, P. 1998. Corelex: systematic polysemy and underspecification. In PhD thesis, Brandeis University Waltham, MA, USA.

Buitelaar, P., Cimiano, P. and Magnini, B. 2005. Ontology Learning from Text: Methods, Evaluation and Applications. Amersterdam: IOS Press.

Caraballo. S. 1999. Automatic Construction of a hypernym-labeled noun hierarchy from text . In proceeding of 37th Annual Meeting of Association of Computational Linguistics (ACL), pp. 120-126, Maryland, USA, Jun. 20-26, 1999.

Caraballo, S.A. 2001. Automatic Acquisition of a Hypernym-Labeled Noun Hierarchy from Text. In Ph.D. Thesis , Brown University.

Chang J. S., 2005. Domain Specific Word Extraction from Hierarchical Web Documents: A First Step Toward Building Lexicon Trees from Web Corpora. In proceedings of the Fourth SIGHAN Workshop on Chinese Language Learning, pp. 64-71, Sapporo, Japan, Oct. 14-15, 2005.

Chen, S. and Williams, M.A. 2008a. Learning Personalized Ontologies from Text: A Review on an Inherently. In Personalized Information Retrieval and Access: Concepts, Methods and Practices, pp. 1-29. IGI Global.

Chen Y.R. 2005. The Research on Automatic Chinese Term Extraction Integrated with Unithood and Domain Feature. In Master Thesis, Peking University.

Chen, Y.R., Lu, Q., Li, W.J., Sui, Z.F., Ji, L.N. 2006. A Study on Terminology Extraction based on Classified Corpora. In Proceedings of the International Conference on Language Resources and Evaluation (LREC 2006), Genoa, Italy, May 24-26, 2006.

Chen, Y.R., Lu, Q. Li, W.J., Cui, G.Y. 2007a. Discovering Kind-of Relation for Ontology Construction. In Chinese Lexical Semantics Workshop (CLSW2007), Hong Kong, May 21-23, 2007.

Chen, Y.R., Lu, Q., Li, W., Li, W., Ji, L., Cui, G.Y. 2007b. Automatic Construction of a Chinese Core Ontology from an English-Chinese Term Bank. In Proceedings of Workshop OntoLex07 From Text to Knowledge: The Lexicon/Ontology Interface, the 6th International Semantic Web Conference, Busan, South Korea, Nov. 11, 2007.

Chen, Y.R., Lu, Q., Li, W., Cui, G. 2008b. Chinese Core Ontology Construction from a Bilingual Term Bank. In Proceedings of the 6th Language Resources and Evaluation Conference (LREC2008), Marrakech, Morocco, May 28-30, 2008.

Chen, Y.R., Lu, Q., Li, W., Cui, G. 2009. A Novel Method to Improve Chinese Core Ontology Construction with Terms Sharing Suffixes. In the proceedings of the 10th Chinese National Conference on Computational Linguistics (CNCCL 2009), Yantai, China, July 24-26, 2009.

Chen, Y.R., Lu, Q., Li, W., Cui, G. 2010. A Novel Method for Chinese Core Ontology Construction, In Journal of Chinese Information Processing, vol. 24, no. 1, pp. 48-53.

Chien, L.F. 1999. Pat-tree-based adaptive keyphrase extraction for intelligent Chinese information retrieval. In Information Processing and Management vol.35 pp.501-521

Church, K.W., & Gale, W.A. 1991. Concordances for parallel text. In Proceedings of the 7th Annual Conference of the UW Center for ITE New OED & Text Research, pp. 40-62, Oxford.

Church, K. W. and Hanks, P. 1990. Word Association Norms, Mutual Information, and Lexicography. In Computational Linguistics, vol. 6, no. 1, pp. 22-29.

Cimiano, P. & Staab, S. & Tane, J. 2003. Automatic Acquisition of Taxonomies from Text: FCA Meets NLP. In Proceedings of the International Workshop on Adaptive Text Extraction and Mining, Cavtat-Dubrovnik , Croatia, Sep. 22, 2003.

Cimiano P., Steffen Staab. 2004. Learning by Googling. In ACM SIGKDD Explorations Newsletter, vol. 6, no. 2, pp.24-33.

Cimiano P., A. Hotho, Steffen Staab. 2005a. Learning Concept Hierarchies from Text Corpora using Formal Concept Analysis. In Journal of Artificial Intelligence Research vol. 24, pp. 305-309.

Cimiano P., Günter Ladwig, Steffen Staab. 2005b. Gimme' the context: context-driven automatic semantic annotation with C-PANKOW. In Proceedings of the 14th International Conference on World Wide Web, Chiba, Japan, May 10-14, 2005.

Cimiano P. 2006. Ontology Learning and Population from Text – Algorithms, Evaluations and Applications. Springer Science+Business Media.

Cohen, J. D. 1995. Highlights: Language- and Domain-independent Automatic Indexing Terms for Abstracting. In Journal of American Soc. for Information Science, vol. 46, no. 3, pp. 162-174.

Concept. 2007. In Wikipedia, The Free Encyclopedia. Retrieved 21:59, January 15, 2007, from http://en.wikipedia.org/w/index.php?title=Concept&oldid=98215510

Culotta A. and Sorensen J. 2004. Dependency tree kernels for relation extraction. In Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics, Barcelona, Spain, July 21-26, 2004.

Cui, G.Y., Lu, Q., Li, W.J. and Chen Y.R. 2007. Attributes Selection in Ontology Taxonomies Acquisition with FCA. In proceedings of Chinese Lexical Semantics Workshop (CLSW2007), Hong Kong, May 21-23, 2007.

Cui, G.Y., Lu, Q., Li, W.J. and Chen Y.R. 2008a. Corpus Exploitation from Wikipedia. In proceedings of the International Conference on Language Resources and Evaluation (LREC 2008), Marrakech, Morocco, May 28-30, 2008.

Cui, G.Y., Lu, Q. and Li, W.J. 2008b. Preliminary Chinese Term Classification for Ontology Construction. In Proceedings of the 6th Workshop on Asian Language Resources, in the Third International Joint Conference on Natural Language Processing (IJCNLP), Hyderabad, India, Jan. 11-12, 2008 .

Cui, G.Y., Lu, Q. and Li, W.J. 2008c. Attributes Selection in Chinese Ontology Acquisition with FCA. In International Journal of Computer Processing of Oriental Languages (IJCPOL), vol. 21, no. 1, pages 77-95.

Cui, G.Y., Lu, Q., Li, W.J. and Chen Y.R. 2009. Automatic Acquisition of Attributes for Ontology Construction. In the 22nd International Conference on the Computer Processing of Oriental Languages (ICCPOL2009), Hong Kong, Mar. 26-27, 2009.

Daille, B. 1994. Study and Implementation of Combined Techniques for Automatic Extraction of Terminology. In the Balancing Act: Combining Symbolic and Statistical Approaches to Language, New Mexico State University, Las Cruces.

Dice, L.R. 1945. Measures of the amount of ecologic associations between species. In Journal of Ecology, 26, 1945.

Doerr, M., Hunter, J., Lagoze, C. 2003. Towards a Core Ontology for Information Integration. In Journal of Digital Information, vol. 4, no. 1, pp. 169-190.

Dong, Z., Dong, Q. 2006. Hownet And the Computation of Meaning. World Scientific Publishing Co., Inc. River Edge, NJ, USA..

Dunning, T. 1993. Accurate Method for the Statistics of Surprise and Coincidence, In Computational Linguistics, vol. 19, no.1, pp. 61-74.

Frank, E., Paynter, G. W., Witten, I. H., Gutwin, C. and Nevill-Manning, C.G. 1999. Domain-specific keyphrase Extraction, In Proceedings of 16th International Joint Conference on Artificial Intelligence IJCAI-99, pp. 668-673, Stockholm, Sweden. Jul. 31 – Aug. 6, 19999.

Fellbaum, C., NetLibrary, I. 1998. WordNet: an electronic lexical database. MIT Press USA.

Ferreira da Silva, J., Pereira Lopes, G. 1999. A local maxima method and a fair dispersion normalization for extracting multi-word units from corpora. In proceedings of Sixth Meeting on Mathematics of Language, pp. 369-381, Madison, Wisconsin USA, Jul. 24-27, 1998.

Firth, J. (1957). A synopsis of linguistic theory. 1930-1955. Studies in Linguistic Analysis, Philological Society, Oxford

Fontenelle, T., Brüls, W., Thomas, L., Vanallemeersch, T., Jansen, J. 1994. DECIDE, MLAP-Project 93-19, deliverable D-1a: Survey of collocation extraction tools. In Tech. Report, Univ. of Liege, Liege, Belgium.

Ganter, B. & Wille, R. 1999. Formal Concept Analysis – Mathematical Foundations, Springer Verlag

Genesereth, M. R., and Fikes, R. E. 1992. Knowledge Interchange Format, Version 3.0 Reference Manual. In Technical Report Logic-92-1, Computer Science Department, Stanford University.

Giles. J. 2005. Special Report--Internet encyclopaedias go head to head. In Nature, vol. 438, no. 15, pp 900-901.

Gilks, W., Richardson, S., Spiegelhalter, D. 1996. Markov Chain Monte Carlo in Practice. Chapman & Hall/CRC.

Giuliano, V. E. 1964. The interpretation of word associations. In M.E. Stevens et al. (Eds.) Statistical association methods for mechanized documentation, pp. 25-32. National Bureau of Standards Miscellaneous Publication 269, Dec. 15, 1965.

Gruber, T and others. 1993. A translation approach to portable ontology specifications. In Knowledge Acquisition, Academic Press, vol.5.

Gruber, T. 1994. Towards principles for the design of ontologies used for knowledge sharing. In Int. J. of Human and computer studies, vol. 43, pp. 907-928.

Gulli, A. and Signorini, A. 2005. The indexable web is more than 11.5 billion pages. In WWW 2005 Conference Proceedings, ACM, pp. 902–903, Chiba, Japan, May 10-14, 2005.

Haav, H-M., 2003, An Application of Inductive Concept Analysis to Construction of Domain-specific Ontologies, In B. Thalheim, Gunar Fiedler (Eds), Emerging Database Research in East Europe, In Proceedings of the Pre-conference Workshop of VLDB 2003, Computer Science Reports, vol. 14, no.3, pp 63-67, Berlin, Germany, Sep. 9-12, 2003.

Harris. Z. 1968. Mathematical Structures of Language, Wiley.

Hazewinkel, M. 2002. Encyclopaedia of Mathematics. Kluwer Academic Publishers.

Hearst, M.A. 1992. Automatic Acquisition of hyponyms from large text corpora. In the proceedings of the 14th international Conference of Computational Linguistics, pp. 539-545, Newark, Deleware, USA, Jun. 28 – Jul. 2, 1992.

Hearst, M.A. 1998. Automatic Discovery of WordNet Relations. In Christiane Fellbaum, editor, WordNet: An Electronic Lexical Database. MIT Press, Cambridge, MA.

Chapter 1 Hindle, D. 1990. Noun classification from predicate-argument structures, In Proceedings of the Annual Meeting of Association of Computational Linguistics (ACL), pp. 268-275, Pittsburgh, Pennsylvania, USA, Jun. 6-9, 1990.

Hirst, G., 2004. Ontology and the Lexicon. In Handbook on Ontologies, S. Staab and R. Studer: Springer, Karlsruhe, pp. 209-230.

Hisamitsu, T. and Niwa, Y. 2002. A measure of term representativeness based on the number of co-occurring salient words. In Proceedings of the 19th 19th International Conference on Computational Linguistics, Taipei, Taiwan, Aug. 2 - Sep.1, 2002

Huang, C., Chang, R. and Lee, S. 2004. Sinica BOW (Bilingual Ontological Wordnet): Integration of Bilingual WordNet and SUMO. In Proceedings of the 4th International Conference on Language Resources and Evaluation, pp. 26–28, Lisbon, Portugal, May 26-28, 2004.

Ji, L.N., Lu, Q., Li, W.J. and Chen, Y.R. 2007a. Automatic Construction of a Core Lexicon for Specific Domain. In Proceeding of the 6th International Conference on Advanced Language Processing and Web Information Technology (ALPIT 2007), Luoyang, China, August 22-27, 2007.

Ji, L.N., Sum, M.T., Lu, Q., Li, W.J. and Chen, Y.R. 2007b. Chinese Terminology Extraction using Window-Based Contextual Information. In Conference on Intelligent Text Processing and Computational Linguistics (CICLing), Mexico, Feb. 18-24, 2007.

Kageura, K. and Umino, B. 1996. Methods of Automatic Term Recognition: A Review, Terminology, vol. 3, no. 2.

Klyne, G. and Carroll, J.J. and McBride, B. 2004. Resource description framework (RDF): Concepts and abstract syntax. In W3C recommendation, vol. 10.

Li, H. and Huang, C.N. and Gao, J. and Fan, X. 2004. The use of SVM for Chinese new word identification. In the proceedings of the First International Joint Conference on Natural Language Processing, pp. 22—24, Hainan Island, Mar. 22-24, 2004.

Li, S.J., Lu Q. and Li W.J.. 2005. Experiments of Ontology Construction with Formal Concept Analysis. In OntoLex Workshop IJCNLP 2005, pp67-75, Jeju Island, Korea, Oct. 11-13, 2005.

Li, S.J., Lu, Q. and so on. 2006. Interaction between Lexical Base and Ontology with Formal Concept Analysis. In Language Resources and Evaluation Conference (LREC), Genoa, Italy, May 24-26, 2006.

Lu, Q., Chan, S.T., Xu, R.F., Chiu, T.S., Li, B.L. and Yu, S.W. 2003. A Unicode Based Adaptive Segmentor. In 2nd SIGHAN Workshop on Chinese Language Processing, ACL 2003, pp. l64-167, Sapporo, Japan, July 11-12, 2003.

Lv, X.Q., Zhang, L., Huang, Z.D. and Hu, J.F. 2004. A Fast Algorithm of Substring Reduction based on Hash technology (基于散列技术的快速子串归并算法). In Journal of Fudan University, vol 43, no. 5.

Joachims, T. 1997. A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization. In the proceedings of the 14th international conference on machine learning ICML, pp.143-151. Nashville, Tennessee, USA, Jul. 8-12, 1997

Luo, S.F., and Sun, M.S. 2003. Two-Character Chinese Word Extraction Based on Hybrid of Internal and Contextual Measures: In Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, pp. 24-30, Sapporo, Japan, July 11-12, 2003.

Maedche, A. 2001. Ontology learning for the Semantic Web. In IEEE Intelligent systems, vol. 16, no. 2, pp.72-79.

Martin, L.E. 1990. Knowledge Extraction. In Proceedings of the Twelfth Annual Conference of the Cognitive Science Society. Hillsdale, NJ: Lawrence Erlbaum Associates, pp. 252-262.

McGuinness, D.L. and Van Harmelen, F. and others. 2004. Owl web ontology language overview. In W3C Recommendation, vol. 10.

Miller, G., Beckwith, R., Fellbaum, C., Gross, D., Miller, K. 1990. Introduction to WordNet: An On-line Lexical Database*. In International Journal of Lexicography, volume 3, no 4, pages 235–244.

Navigli, R., Velardi, P. 2004. Learning Domain Ontologies from Document Warehouses and Dedicated Web Sites. In Computational Linguistics vol. 30, pp. 151–179

Nakagawa, H., and Mori, T. 2002. A simple but powerful automatic term extraction method. In COMPUTERM-2002 Proceedings of the 2nd International Workshop on Computational Terminology, pp. 29-35, Taipei, Taiwan, August 31, 2002,.

Navigli, R., Velardi, P. 2004 Learning domain ontologies from document warehouses and dedicated websites. In Computational Linguistics, vol. 30, pp. 151--179.

Niles, I., Pease, A. 2001. Towards a Standard Upper Ontology. In Proceedings of the international conference on Formal Ontology in Information Systems, vol. 2001, pp. 2–9, Ogunquit, Maine, October 17-19, 2001.

Niles, I., Pease, A. 2003. Linking Lexicons and Ontologies: Mapping WordNet to the Suggested Upper Merged Ontology. In Proceedings of the IEEE International Conference on Information and Knowledge Engineering, pp. 412–416, Las Vegas, Nevada, USA, Jun. 23 - 26, 2003.

Ontology (computer science). 2007. In Wikipedia, The Free Encyclopedia. Retrieved 00:44, April 12, 2007, from http://en.wikipedia.org/w/index.php?title=Ontology_%28computer_science%29&oldid=120437387.

Pease, A., Niles, I., Li, J. 2002. The Suggested Upper Merged Ontology: A Large Ontology for the Semantic Web and its Applications. In Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web, vol. 28, Edmonton, Alberta, Canada, Jul. 29, 2002.

Pinto, H.S. and Martins, J.P. 2004. Ontologies: How can they be built? In Knowledge and Information Systems. vol. 6, no. 4, pp. 441-464.

Resnik, P. 1996. Selectional constraints: an information-theoretic model and its computational realization. In Cognition. Vol. 61, pp. 127-159.

Rodr´ıguez, H., Climent, S., Vossen, P., Bloksma, L., Peters, W., Alonge, A., Bertagna, F., Roventini, A. 1998. The Top-Down Strategy for Building EuroWord-Net: Vocabulary Coverage, Base Concepts and Top Ontology. In Computers and the Humanities, vol 32, no. 2, pp. 117–152.

Ruiz-Casado, M. and Alfonseca, E. and Castells, P. 2005. Automatic assignment of wikipedia encyclopedic entries to wordnet synsets. In Advances in Web Intelligence, 3528:380-386. Springer.

Scheuren, F. and Association, A.S. 2004. What is a Survey? American Statistical Association.

Semy, Salim K., Mary K. Pulvermacher and Leo J. Obrst. 2004 . Toward the Use of an Upper Ontology for U.S. Government and U.S. Military Domains: An Evaluation, In

MITRE Technical Report 04B0000063, The MITRE Corporation, available online at: http://www.mitre.org/work/tech_papers/tech_papers_04/04_0603/.

Schone, P., Jurafsky D. 2001. Is knowledge-free induction of multiword unit dictionary headwords a solved problem? In Proceedings of the 6th Conference on Empirical Methods in Natural Language Processing (EMNLP 2001), pp. 100-108, Carnegie Mellon University, Pittsburgh, PA USA, Jun. 3 - 4, 2001.

Smadja, F. 1993. Retrieving collocations from text: Xtract. In Computational Linguistics, vol. 19, no.1, pp. 143-177.

Sornlertlamvanich V., Potipiti T., Charoenporn T. 2000. Automatic corpus-based Thai word extraction with the C4.5 learning algorithm. In Proceedings of COLING 2000, Saarbrücken, Germany, Jul. 31 – Aug. 4, 2000.

Sowa, J.F., 2000, Knowledge Representation, Logical, Philosophical, and Computational Foundations, Brooks/Cole Thomson Learning.

Studer, R., Benjamins, V.R. and Fensel, D. 1998. Knowledge engineering: Principles and methods. In Data & Knowledge Engineering, vol. 25, no 1-2, pp. 161-197.

Suchanek, F.M. and Kasneci, G. and Weikum, G. 2007. Yago: a core of semantic knowledge. In Proceedings of the 16th international conference on World Wide Web, pp. 697-706. ACM New York, NY, USA, May 8-12, 2007.

Sui, Z.F., Chen Y.R. and so on. 2002. The Research on the Automatic Term Extraction in the Domain of Information Science and Technology. In Proceedings of the 5th East Asia Forum of the Terminology, Haikou city, Hainan province, China, Dec. 1-8, 2002.

SUMO. 2002. Suggested Upper Merged Ontology, web site, http://ontology.teknowledge.com

Standard Upper Ontology Knowledge Interchange Format, 2008. Retrieved at Jan. 16th, 2009 from http://suo.ieee.org/SUO/KIF/suo-kif.html

Tang, A., Zhen, Z., Fan, J. 2005. Thesaurus-based Approach to Build Domain Ontology. In New Technology of Library and Information Service (in Chinese), pp. 1–5.

Wikipedia:Introduction. 2009. Retrieved in April 16th, 2009 from Web Page http://en.wikipedia.org/w/index.php?title=Wikipedia:Introduction&oldid=284144782.

WordNet 3.0, 2007. Retrieved at Jan 16th, 2007 from http://wordnet.princeton.edu/perl/webwn

Yang, Y., Lu, Q. and Zhao, T. 2008. Chinese term extraction using minimal resources. In Proceedings of the 22nd International Conference on Computational Linguistics, pp 1033-1040, vol. 1, Manchester, United Kingdom, Jun. 16-18, 2008.

Zhang, H.P. and Yu, H.K. and Xiong, D.Y. and Liu, Q. 2003. HHMM-based Chinese Lexical Analyzer ICTCLAS, In Proceedings of the Second SIGHAN Workshop on Chinese Language Processing, pp. 184-187, vol. 17, Sapporo, Japan, Jul. 11-12, 2003.

Zhang, Q.L. and Sui, Z.F. 2006. The calculation of Termhood in Automatic Term Extraction, In the proceedings of the International Conference on Terminology, Standardization and Technology Transfer. Beijing, China, Aug. 25-26, 2006.

Appendix 1

The 49 Chinese Terms used in Evaluation of FCA

ASCII	电子邮件	计算机	模式识别	微型机
CPU	调制解调器	计算机辅助	内存	硬磁盘
编程	读取	计算机化	屏幕	硬盘
操作系统	服务器	计算中心	软硬件	中央处理器
存储器	工作站	监视器	数字计算机	终端
存取	光标	兼容机	图标	终端机
大型机	硅谷	键盘	微处理机	主板
单板机	缓存	解码	微处理器	字节
电脑	回车	空格	微电脑	总线
电子商务	寄存器	联机	微机	

Appendix 2

Mappings of 1500 Chinese IT Core Terms to Synsets and SUMO

The mapping of 1,500 core terms to synset and SUMO is listed as following according to the format which print the serial number and core term first, then print the corresponding SUMO concept in result and corresponding one in answer with a punctuation “/” between them, print a line break, and finally print the corresponding synset (with synset id bracketed, “synset id” is a id for synset used in WordNet 1.6 [Fellbaum, 1998], the last character of synset id is its part of speech) in both result and answer respectively. Each synset will be printed in a exclusive line. If there is no answer for the core term, the answer parts will be blank, as well as the result parts.

1 系统 Device / Device

system,unit(03459836n)

system,unit(03459836n)

2 器 Device /

liquid_crystal_display,LCD(02931570n)

3 控制 agent / agent

control,controlling(00513665n)

control,controlling(00513665n)

4 式 manner / Class

manner,mode,style,way,fashion(03856995n)

type(04497251n)

5 程序 ComputerProgram / ComputerProgram

program,programme,computer_program,computer_programme(04930290n)

program,programme,computer_program,computer_programme(04930290n)

6 数据 Proposition / Proposition

data,information(06250971n)

data,information(06250971n)

7 自动 Device / Device

automatic(00182347a)

automatic(00182347a)

8 设备 Device / Device

device(02560468n)

device(02560468n)

9 多 Collection / Collection

multiple(02112966a)

multiple(02112966a)

10 计算机 Machine / Machine

computer,data_processor,electronic_computer,information_processing_system(02481557n)

computer,data_processor,electronic_computer,information_processing_system(02481557n)

11 电 Radiating / Radiating

electric,electrical(02622421a)

electric,electrical(02622421a)

12 文件 Text / Text

file,data_file(04885060n)

file,data_file(04885060n)

13 线 ShapeAttribute / EngineeringComponent

line(09988142n)

wire,conducting_wire(03625791n)

14 网络 Artifact / Collection

network,net,mesh,meshwork,reticulation(03038207n)

network,web(06239044n)

15 信息 Text / Text

information,info(04977171n)

information,info(04977171n)

16 电路 EngineeringComponent / EngineeringComponent

circuit,electrical_circuit,electric_circuit(02443096n)

circuit,electrical_circuit,electric_circuit(02443096n)

17 时间 TimeMeasure / TimeMeasure

time(00015594n)

time(00015594n)

18 操作 ContentDevelopment / ContentDevelopment

operation(09747131n)

operation(09747131n)

19 表 Text / ContentBearingObject

list,listing(04866394n)

table,tabular_array(06138367n)

20 非 not / not

not(00022819r)

not(00022819r)

21 语言 Language / Language

language,linguistic_communication(04748361n) language,linguistic_communication(04748361n)

22 数字 NonnegativeInteger / Device

digit(09893975n)

digital(00114349a)

23 记录 ContentBearingObject / Text

recording(03220068n)

record(04984758n)

24 信号 Icon / Icon

signal,signaling,sign(05085885n)

signal,signaling,sign(05085885n)

25 电子 Electron / Electron

electron(06678414n)

electron(06678414n)

26 型 Class / Class

type(04497251n)

type(04497251n)

27 技术 FieldOfStudy / FieldOfStudy

technology,engineering(00607693n)

technology,engineering(00607693n)

28 逻辑 Procedure / SubjectiveAssessmentAttribute

logic,logical_system,system_of_logic(04379355n)

logic(04348694n)

29 结构 Artifact / Artifact

structure,construction(03431817n)

structure,construction(03431817n)

30 单 SubjectiveAssessmentAttribute / Artifact

simple(02090222a)

sheet,piece_of_paper,sheet_of_paper(04731186n)

31 管 Artifact / EngineeringComponent

tube,tubing(03547838n)

tube,vacuum_tube,thermionic_vacuum_tube,thermionic_tube,electron_tube,thermionic_valve(03548414n)

32 分析 Investigating / Investigating

analysis(00422134n)

analysis(00422134n)

33 模型 Icon / Proposition

model,simulation(03007566n)

model,theoretical_account,framework(04527384n)

34 输入 Icon / Icon

input_signal,input(05421788n)

input_signal,input(05421788n)

35 过程 IntentionalProcess / IntentionalProcess

procedure,process(00660718n)

procedure,process(00660718n)

36 标准 NormativeAttribute / NormativeAttribute

standard,criterion,measure,touchstone(05418696n)

standard(02185491a)

37 主 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

primary(01781200a)

primary(01781200a)

38 光 RadiatingLight / RadiatingLight

light,visible_light,visible_radiation(07811448n) light,visible_light,visible_radiation(07811448n)

39 处理 Process / Process

processing(09745007n)

processing(09745007n)

40 函数 Function / Function

function,mathematical_function(09926856n)

function,mathematical_function(09926856n)

41 存储器 EngineeringComponent / EngineeringComponent

memory,storage,store,memory_board(02984419n)

memory,storage,store,memory_board(02984419n)

42 带 Group / Artifact

set,circle,band,lot(06117068n)

strip,slip(03430059n)

43 开关 EngineeringComponent / EngineeringComponent

switch(03456067n)

switch(03456067n)

44 双 Collection / Collection

double,dual,duple(02114353s)

double,dual,duple(02114353s)

45 点 exactlyLocated / exactlyLocated

point(06351684n)

point(06351684n)

46 图 Icon / Icon

chart(05252690n)

graph,graphical_record(05253047n)

47 问题 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

problem,job(10340761n)

trouble,problem(04395081n)

48 服务 IntentionalProcess / Position

service(00379388n)

service(00384296n)

49 装置 Device / Device

device(02560468n)

device(02560468n)

50 指令 Procedure / Procedure

direction,instruction(05082722n)

direction,instruction(05082722n)

51 字符 Character / Character

character,grapheme,graphic_symbol(05107649n)

character,grapheme,graphic_symbol(05107649n)

52 位 NonnegativeInteger / part

digit(09893975n)

bit,chip,flake,fleck,scrap(06687090n)

53 显示 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

expose,exhibit,display(01464775v)

expose,exhibit,display(01464775v)

54 方法 Procedure / Procedure

method(04376254n)

method(04376254n)

55 码 Obligation / Procedure

code,codification(05000141n)

code(04790774n)

56 协会 Organization / Organization

association(05999585n)

association(05999585n)

57 模式 manner / Abstract

manner,mode,style,way,fashion(03856995n)

form,shape,pattern(04554317n)

58 软件 ComputerProgram / ComputerProgram

software,software_system(04927772n)

software,software_system(04927772n)

59 终端 Device / Device

terminal(03488343n)

terminal(03488343n)

60 功能 Function / hasPurpose

function,mathematical_function(09926856n)

function,purpose,role,use(04011535n)

61 磁 Meter / Integer

meter,metre,m(09822324n)

thousand,a_thousand,one_thousand,1000,m,k(02102663s)

62 测试 ConstantQuantity / Investigating

trial,test,tryout(04469208n)

test,prove,try,try_out,examine,essay(01726067v)

63 板 Organization / Artifact

board(06171035n)

board(02302683n)

64 脉冲 Radiating / Process

nerve_impulse,impulse(05448312n)

pulsation,pulsing,pulse,impulse(05522821n)

65 形 /

66 存储 Keeping / Keeping

storage(00517908n)

storage(00517908n)

67 数 Number / Number

number(09765658n)

number(09765658n)

68 无 NormativeAttribute / not

nugatory,null,void(02381059s)

none(00023400r)

69 电流 Motion / Radiating

current,stream(05513139n)

current,electric_current(07788451n)

70 通信 Communication / Communication

communication,communicating(04728675n)

communication,communicating(04728675n)

71 设计 Device / PsychologicalProcess

device(02560468n)

design,designing(00594989n)

72 连接 ContentDevelopment / Relation

concatenate(00130632v)

connection,connexion,connectedness(09931077n)

73 用户 experiencer / experiencer

user,enjoyer(07658488n)

user,enjoyer(07658488n)

74 图形 ArtWork / ArtWork

graphics(02767450n)

graphic(02572017a)

75 输出 result / result

end_product,output(02639315n)

output,yield(00587304n)

76 命令 Ordering / Ordering

command,bid,bidding,dictation(05359574n)

command,bid,bidding,dictation(05359574n)

77 选择 SubjectiveAssessmentAttribute / Selecting

option,alternative,choice(04463475n)

choice,selection,pick(00104850n)

78 电压 UnitOfMeasure / UnitOfMeasure

voltage,electromotive_force,emf(07847141n)

potential_difference,potential_drop,voltage(07824877n)

79 键 Device / Device

key(02886812n)

key(02886812n)

80 通用 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

cosmopolitan,ecumenical,oecumenical,general,universal,worldwide(00493390s)

cosmopolitan,ecumenical,oecumenical,general,universal,worldwide(00493390s)

81 地址 uniqueIdentifier / uniqueIdentifier

address,computer_address(04792458n)

address,computer_address(04792458n)

82 计 Device / Device

meter(02990606n)

meter(02990606n)

83 波 Motion / Motion

wave,moving_ridge(05477241n)

wave,undulation(05473259n)

84 状态 SocialRole / Attribute

status,position(10048803n)

state(00016185n)

85 块 Artifact / Collection

block(02299245n)

block(05967612n)

86 热 Heating / Heating

thermal,thermic,caloric(02613791a)

thermal,thermic,caloric(02613791a)

87 值 Quantity / Quantity

value(04508194n)

value(04508194n)

88 符号 Character / Character

symbol(05097156n)

symbol(05097156n)

89 区 GeographicArea / Room

area,country(06268839n)

area(02207321n)

90 管理 Managing / Managing

management,direction,managing(00734610n)

management,direction,managing(00734610n)

91 基本 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

basic(01784247a)

basic(01784247a)

92 分 Separating / Separating

divide,split,split_up,separate,dissever,carve_up(01681144v)

divide,split,split_up,separate,dissever,carve_up(01681144v)

93 子 NonFullyFormed / Human

child,kid,youngster,minor,shaver,nipper,small_fry,tiddler,tike,tyke,fry,nestling(07153837n)

child,kid(07154377n)

94 接 connected / connected

joining,connection,connexion,connecting(00094409n)

joining,connection,connexion,connecting(00094409n)

95 理论 Proposition / Proposition

theory(04592375n)

theory(04592375n)

96 试验 Investigating / Investigating

experiment,experimentation(00418704n)

experiment,experimentation(00418704n)

97 磁盘 Device / Device

magnetic_disk,magnetic_disc,disk,disc(02955026n)

magnetic_disk,magnetic_disc,disk,disc(02955026n)

98 线路 EngineeringComponent / EngineeringComponent

circuit,electrical_circuit,electric_circuit(02443096n)

cable,electrical_cable,line,transmission_line(02364710n)

99 字 Word / Word

word(04750884n)

word(04750884n)

100 方式　manner / manner

manner,mode,style,way,fashion(03856995n)

manner,mode,style,way,fashion(03856995n)

101 动态　SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

dynamic,dynamical(00769090a)

dynamic,dynamical(00769090a)

102 频率　TimeDependentQuantity /

frequency,frequence,oftenness(10975635n)

103 用　BodyPart /

finger(04312497n)

104 转换　Calculating /

conversion(04471523n)

105 体　Organism /

body,organic_structure,physical_structure(04055158n)

106 误差　SubjectiveAssessmentAttribute /

mistake,error,fault(00042411n)

107 组　Group /

group,grouping(00017954n)

108 单元　UnitOfMeasure /

unit_of_measurement,unit(09760609n)

109 电缆　EngineeringComponent /

cable,electrical_cable,line,transmission_line(02364710n)

110 自　modalAttribute / modalAttribute

own(a),ain(01704265s)

own(a),ain(01704265s)

111 控制系统　　Machine /

control_system(02493845n)

112 寄存器　Device /

cash_register,register(02399372n)

113 流　　Motion /

flow,flowing(05512579n)

114 微　　SubjectiveAssessmentAttribute /

micro(01338580s)

115 率　　TimeDependentQuantity /

rate(10980504n)

116 级　　ConstantQuantity /

degree,grade,level(03973686n)

117 应用　　uses /

application,practical_application(00607299n)

118 天线　　Device /

antenna,aerial,transmitting_aerial(02191609n)

119 故障　　NormativeAttribute /

failure(05454021n)

120 空间　　Set / Set

space,topological_space(05993009n)

space,topological_space(05993009n)

121 模拟　　represents /

simulation(00577896n)

122 通道　　instrument /

channel,transmission_channel(04733874n)

123 代码　　ComputerProgram /

code,computer_code(04792090n)

124 格式　ContentDevelopment /

format,formatting,data_format,data_formatting(04979062n)

125 度　LengthMeasure /

dimension(03973414n)

126 顺序　SubjectiveAssessmentAttribute /

consecutive,sequent,sequential,serial,successive(01608832s)

127 线性　Relation /

linear,additive(01360662a)

128 同步　cooccur /

synchronous,synchronal,synchronic(02265894a)

129 元件　subsumesContentInstance / EngineeringComponent

component,constituent,element,factor,ingredient(04515709n)

component,constituent,element(02479780n)

130 量　PhysicalQuantity /

measure,quantity,amount,quantum(00020056n)

131 编码　Encoding /

coding(00405074n)

132 中断　Communication /

interrupt,disrupt,break_up,cut_off(00528650v)

133 循环　ShapeAttribute /

annular,annulate,annulated,circinate,circular,ringed,ring-shaped(01969290s)

134 机器　Device /

machine(02949521n)

135 卡　ContentBearingObject /

card(02387006n)

136 类型　Class /

type(04497251n)

137 接口　EngineeringComponent /

interface,port(02862795n)

138 数据库 ContentBearingObject / ContentBearingObject

database(04979821n)

database(04979821n)

139 错误　SubjectiveAssessmentAttribute /

mistake,error,fault(00042411n)

140 路　/

141 网　Artifact /

network,net,mesh,meshwork,reticulation(03038207n)

142 面　BodyPart /

face,human_face(04340595n)

143 目标　CorpuscularObject /

object,physical_object(00009457n)

144 打印　Publication /

print,publish(01193001v)

145 环境　SubjectiveAssessmentAttribute /

environment(10040804n)

146 传输　Transfer /

transmission,transmittal,transmitting(00077376n)

147 角　StreamWaterArea /

delta(06728279n)

148 直接　SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

straightforward(00853079s)

straightforward(00853079s)

149 计算 Device /

computational(02685764a)

150 行 Collection /

line(06235973n)

151 工程 FieldOfStudy /

engineering,engineering_science,applied_science,technology(04660658n)

152 算法 Procedure /

algorithm,algorithmic_rule,algorithmic_program(04502720n)

153 放大器 Device /

amplifier(02185415n)

154 任务 IntentionalProcess /

undertaking,project,task,labor(00508925n)

155 制 Collection /

system(06239374n)

156 有效 SubjectiveAssessmentAttribute /

effective,effectual,efficacious(00795039a)

157 参数 Quantity / Quantity

parameter,parametric_quantity(04509624n)

parameter,parametric_quantity(04509624n)

158 协议 NormativeAttribute /

protocol,communications_protocol(04998026n)

159 区域 GeographicArea /

area,country(06268839n)

160 效应 IntentionalProcess /

action(00022246n)

161 远程　PositionalAttribute /

distant,remote,removed(00411239s)

162 超　SubjectiveAssessmentAttribute /

ace,A-one,crack,first-rate,super,tiptop,topnotch,tops(p)(02230912s)

163 集　Set /

set(05992373n)

164 条件　manner /

condition,status(10032130n)

165 控制器　EngineeringComponent /

control,controller(02492359n)

166 电阻　UnitOfMeasure / EngineeringComponent

electric_resistance,electrical_resistance,impedance,resistance,resistivity,ohmic_resistance(07832149n)

resistor,resistance(03232583n)

167 压　Touching /

press(00992533v)

168 层　part /

layer(02913016n)

169 窗口　Artifact /

window(03619904n)

170 交换　Substituting /

switch,switching,shift(00128676n)

171 处理器　EngineeringComponent /

central_processing_unit,CPU,C.P.U.,central_processor,processor,mainframe(02413762n)

172 内部 located /

internal(00900967a)

173 存取 ContentDevelopment /

access,memory_access(02159706n)

174 辅助 Position /

aide,auxiliary(07068776n)

175 加 SubjectiveAssessmentAttribute / ConstantQuantity

asset,plus(04015246n)

plus(01749034a)

176 学 /

177 综合 ChemicalSynthesis /

synthesis(09689446n)

178 传送 Device /

tape_drive,tape_transport,transport(03471487n)

179 打印机 Machine /

printer,printing_machine(03173458n)

180 最优 SubjectiveAssessmentAttribute /

optimum,optimal(00226342s)

181 扫描 ContentDevelopment /

scanning(09676192n)

182 平均 Quantity /

average,norm(04609375n)

183 特性 SubjectiveAssessmentAttribute /

characteristic(00346121a)

184 固定 Number /

fixed(00705447s)

185 检验 Reasoning / Investigating

confirmation,verification,check,substantiation(04486906n)

check,checkout,check-out_procedure(00092615n)

186 输入输出 /

187 磁带 Device /

magnetic_tape,mag_tape,tape(02956039n)

188 硬件 Weapon /

hardware(02798996n)

189 总线 TransportationDevice /

bus,autobus,coach,charabanc,double-decker,jitney,motorbus,motorcoach,omnibus(02356526n)

190 标记 Character /

mark(05106558n)

191 扩展 Increasing /

expansion,enlargement(00235235n)

192 反 ReciprocalFn /

inverse,reciprocal(09973100n)

193 检查 Pursuing /

examination,inspection,scrutiny(00419850n)

194 卡片 ContentBearingObject /

card(02387006n)

195 索引 Quantity / Putting

index,index_number,indicant,indicator(04981133n)

index(01684781v)

196 目录 Text /

directory(04872091n)

197 通讯　Communication /

communication,communicating(04728675n)

198 源　origin /

beginning,origin,root,source(06275359n)

199 环　ShapeAttribute /

ring,halo,annulus,anulus,doughnut,anchor_ring(09996637n)

200 图像　Icon /

image,mental_image(04551473n)

201 单位　UnitOfMeasure /

unit_of_measurement,unit(09760609n)

202 位置　located /

location(00014887n)

203 极　Radiating /

polar(00348317s)

204 射　RadiatingLight /

fire(05444579n)

205 分类　Class / Classifying

kind,sort,form,variety(04496504n)

categorization,categorisation,classification(00653558n)

206 零　SubjectiveAssessmentAttribute /

nothing,nil,nix,nada,aught,cipher,cypher,goose_egg,naught,zero,zilch,zip(09893278n)

207 外部　located /

external(00900427a)

208 半　part /

one-half,half(09891093n)

209 并行　SpatialRelation /

parallel(01655462a)

210 域　SubjectiveAssessmentAttribute /

sphere,domain,area,orbit,field,arena(10410497n)

211 系数　RealNumber /

coefficient(09762118n)

212 模块　capability / EngineeringComponent

faculty,mental_faculty,module(04369800n)

module(03008339n)

213 段　component /

section,segment(03296757n)

214 指示器　Device /

indicator(02854245n)

215 气　Gas /

gas(10671811n)

216 中心　Region /

center,centre,middle,heart,eye(06286659n)

217 面向　orientation /

oriented,orientated(01622433a)

218 安全　SubjectiveAssessmentAttribute /

safety(10425881n)

219 功率　SubjectiveAssessmentAttribute /

power,powerfulness,potency(04041746n)

220 变换　Function /

transformation(09927401n)

221 范围　SubjectiveAssessmentAttribute /

scope,range,reach,orbit,compass,ambit(03993027n)

222 站　Building / Building

station(03404271n)

station(03404271n)

223 片　ContentBearingObject /

card(02387006n)

224 总　CurrencyMeasure /

gross(01526297a)

225 分布　RelationExtendedToQuantities /

distribution(04611872n)

226 库　Reasoning /

establish,base,ground,found(00430666v)

227 运算　ContentDevelopment /

operation(09747131n)

228 盘　Device /

magnetic_disk,magnetic_disc,disk,disc(02955026n)

229 基　Reasoning /

establish,base,ground,found(00430666v)

230 包　Collection /

package,bundle,packet,parcel(05997364n)

231 页　Artifact /

page(04731849n)

232 分配　part / Giving

allotment,allocation(09565241n)

allotment,apportionment,apportioning,allocation,parceling,parcelling(00701814n)

233 周期 TimeDuration /

time_period,period,period_of_time,amount_of_time(10843624n)

234 处理机 EngineeringComponent /

central_processing_unit,CPU,C.P.U.,central_processor,processor,mainframe(02413762n)

235 水 Water /

aquifer(06692025n)

236 变量 Quantity /

variable,variable_quantity(04508699n)

237 项 Text /

item,point(04867079n)

238 雷达 Device /

radar,microwave_radar,radio_detection_and_ranging,radiolocation(03201157n)

239 保护 Maintaining /

protection,protecting,guarding(00522858n)

240 速度 SpeedFn /

speed,velocity(10978183n)

241 绝缘 SurfaceChange /

insulation,insularity,insularism,detachment(10343743n)

242 部分 part / part

part,portion,component_part,component(09945970n)

part,portion,component_part,component(09945970n)

243 虚拟 True /

virtual(a),practical(a)(01872957s)

244 空 NormativeAttribute /

nugatory,null,void(02381059s)

245 随机　SubjectiveAssessmentAttribute /

random(01856328a)

246 振荡器 Device /

oscillator(03065015n)

247 延迟　TimeInterval /

delay,hold,time_lag,postponement,wait(10970463n)

248 测量　PhysicalQuantity /

measure,quantity,amount,quantum(00020056n)

249 端　superficialPart / superficialPart

end(06314878n)

end(06314878n)

250 滤波器 Device /

filter(02679962n)

251 类　Class /

class,category,family(05991008n)

252 自适应 Text /

adaptation,version(04830233n)

253 最小　MinuteDuration /

minute,min(10943650n)

254 识别　Classifying /

designation,identification(00098857n)

255 标志　Icon /

flag(02692272n)

256 特征　Attribute /

feature,characteristic(04504455n)

257 场　Region /

field,field_of_force,force_field(07798336n)

258 平衡　equal / SubjectiveAssessmentAttribute

balance(10087814n)

balance,equilibrium,equipoise,counterbalance(10015195n)

259 相　TimeInterval /

phase,stage(10983365n)

260 孔　Device /

aperture(02200211n)

261 连续　forall /

continuous,uninterrupted(00562039a)

262 程序设计　Encoding /

programming,computer_programming(00595769n)

263 力　Radiating /

force(07799815n)

264 线圈　EngineeringComponent /

coil(02468210n)

265 作业　Position /

job,employment,work(00383807n)

266 视频　Icon /

video,picture(04745386n)

267 最大　SubjectiveAssessmentAttribute /

maximal,maximum(01433642a)

268 说明　Plan /

specification,spec(05042568n)

269 键盘　Device / Device

keyboard(02887166n)

keyboard(02887166n)

270 机械　Device /

machinery(02950954n)

271 头　BodyPart /

head,caput(04290247n)

272 专用　SubjectiveAssessmentAttribute /

private(01787580a)

273 矩阵　ContentBearingObject /

matrix(06139195n)

274 干扰　Law /

intervention,interference(04995117n)

275 列　Motion /

column(06234736n)

276 性能　realization / Process

performance,public_presentation(05156358n)

operation,functioning,performance(00378579n)

277 继电器 Communication /

relay(00503786v)

278 压缩　Encoding /

compression(00405341n)

279 卷　VolumeMeasure /

volume(09923970n)

280 工具　agent /

execution,implementation,carrying_out(00729882n)

281 金属　Metal /

metallic_element,metal(10476248n)

282 图象　Icon /

image,mental_image(04551473n)

283 调制　Communication /

modulation(04747347n)

284 响应　causes / causes

response(07769123n)

response(07769123n)

285 曲线　ShapeAttribute /

curve,curved_shape(09990997n)

286 驱动　Transportation /

drive(01317321v)

287 转换器 Device /

converter,convertor(02494885n)

288 灯　Device /

lamp(02901979n)

289 局部　City /

local(01056664a)

290 作用　Process /

natural_process,natural_action,action,activity(09670326n)

291 组合　SelfConnectedObject /

combination(05971411n)

292 请求　Requesting /

request,bespeak,call_for,quest(00510998v)

293 运行　Walking /

run(01314495v)

294 检测　Perception /

detection,sensing(04412425n)

295 机构 Government / Government

agency,government_agency,bureau,office,authority(06182017n)

administration,governance,establishment,brass,organization,organisation(06071657n)

296 形式 NormativeAttribute /

formal(00990821a)

297 快速 SubjectiveAssessmentAttribute /

fast(00925538a)

298 树 FloweringPlant /

tree(09396070n)

299 复 SubjectiveAssessmentAttribute /

complex(02091753a)

300 转 DirectionChange /

turn(01301287v)

301 无线电 Device /

radio,wireless(03202746n)

302 负载 Object /

load,loading,burden(02934345n)

303 语句 Stating / Directing

statement(05040541n)

instruction,command,statement,program_line(04941372n)

304 低 SubjectiveAssessmentAttribute /

low(01161459a)

305 信息系统 Machine /

data_system,information_system(02546349n)

306 电话 Device /

telephone,phone,telephone_set(03478277n)

307 模　　manner /

manner,mode,style,way,fashion(03856995n)

308 路径　　path /

path,route(06348591n)

309 支持　　Maintaining /

support(00788240n)

310 序列　　List /

sequence(06247375n)

311 定义　　Stating /

definition(05054071n)

312 描述　　Stating /

description,verbal_description(05042407n)

313 配置　　InternalChange / InternalChange

configuration,constellation(04424071n)

configuration,constellation(04424071n)

314 人工　　Book /

manual(04840046n)

315 解　　Solution /

solution(10451252n)

316 发射　　Radiating /

emission(09679767n)

317 字节　　Byte /

byte(09796150n)

318 资源　　origin /

beginning,origin,root,source(06275359n)

319 实时　　/

320 串　　Artifact /

string,twine(03428889n)

321 编辑　ContentDevelopment /

editing,redaction(04844438n)

322 群　　Group / Group

group,grouping(00017954n)

group,grouping(00017954n)

323 链　　Collection /

chain,concatenation(06247825n)

324 联机　path /

on-line(01590949a)

325 混合　Food /

mix,premix(05909690n)

326 二进制 Number /

binary(02521946a)

327 电容器 EngineeringComponent /

capacitor,condenser,electrical_condenser(02381151n)

328 复合　CorpuscularObject /

complex,composite(04516911n)

329 电池　EngineeringComponent /

cell,electric_cell(02410439n)

330 反馈　Radiating /

feedback(09743971n)

331 部件　subsumesContentInstance /

component,constituent,element,factor,ingredient(04515709n)

332 声音　RadiatingSound / RadiatingSound

sound(03892643n)

sound(03892643n)

333 参考　Text /

citation,credit,reference,mention,quotation(05068012n)

334 科学　FieldOfStudy /

science,scientific_discipline(04596907n)

335 事件　Process /

event(00017297n)

336 报告　Text /

report,study(05391713n)

337 接收　Getting /

reception,receipt,receiving(00055481n)

338 讯号　Icon /

signal,signaling,sign(05085885n)

339 智能　SubjectiveAssessmentAttribute /

intelligent(01281024a)

340 读　Reading /

read(00423416v)

341 访问　ContentDevelopment /

access,memory_access(02159706n)

342 集成　Organization /

incorporate,incorporated,integrated,merged,unified(02358190s)

343 桥　ShapeAttribute / ShapeAttribute

axis(04600065n)

axis(04600065n)

344 跟踪 Pursuing /

trailing,tracking(00206842n)

345 半导体 EngineeringComponent /

semiconductor_device,semiconductor_unit,semiconductor(03301106n)

346 服务器 Position /

waiter,server(07671545n)

347 相位 TimeInterval /

phase,stage(10983365n)

348 门 Artifact /

gate(02747160n)

349 发送 Transfer /

sending(00077232n)

350 高级 SubjectiveAssessmentAttribute /

advanced(01768437s)

351 载波 Corporation /

carrier,common_carrier(06005839n)

352 分级 Classifying / Classifying

hierarchical,hierarchal,hierarchic(01152594a)

hierarchical,hierarchal,hierarchic(01152594a)

353 阻抗 UnitOfMeasure /

electric_resistance,electrical_resistance,impedance,resistance,resistivity,ohmic_resistance(07832149n)

354 安装 Putting /

installation,installing,installment(00153919n)

355 常数 RealNumber /

constant(09761890n)

356 工业　Corporation /

industry(06011363n)

357 字母　Character / Character

letter,letter_of_the_alphabet,alphabetic_character(05115901n)

letter,letter_of_the_alphabet,alphabetic_character(05115901n)

358 箱　Periodical /

magazine,mag(04947827n)

359 接触　Communication /

contact(00023118n)

360 规则　Procedure /

rule,formula(04502282n)

361 克　Gram /

gram,gramme,gm,g(09879963n)

362 缓冲　CompoundSubstance /

buffer(10603798n)

363 交互式 BiologicalAttribute /

synergistic,interactive(00588063a)

364 电源　origin /

beginning,origin,root,source(06275359n)

365 耦合　EngineeringComponent /

yoke,coupling(03640719n)

366 温度　TemperatureMeasure / TemperatureMeasure

temperature(03914851n)

temperature(03914851n)

367 属性　IntentionalPsychologicalProcess /

impute,ascribe,assign,attribute(00492345v)

368 交叉 Motion /

traverse,track,cover,cross,pass_over,get_over,get_across,cut_through,cut_across(01304824v)

369 统计 Quantity /

statistic(04608966n)

370 文本 Text /

text,textual_matter(04816275n)

371 噪声 RadiatingSound /

noise(05499597n)

372 恢复 SubjectiveAssessmentAttribute /

recovery(05521978n)

373 使用 uses / uses

use,usage,utilization,utilisation,employment,exercise(00605730n)

use,usage,utilization,utilisation,employment,exercise(00605730n)

374 变压器 EngineeringComponent /

transformer(03530611n)

375 槽 Phrase /

slot(04749663n)

376 概念 Proposition /

concept,conception,construct(04493671n)

377 检索 ContentDevelopment /

retrieval(09753207n)

378 呼叫 Communication /

call,phone_call,telephone_call(04742189n)

379 执行 ContentDevelopment /

execution(09750253n)

380 生成　AgeGroup /

coevals,contemporaries,generation(06196326n)

381 能力　capability /

ability(04047716n)

382 静态　SubjectiveAssessmentAttribute /

tranquillity,quiet(10064397n)

383 电容　FunctionQuantity /

capacitance,electrical_capacity,capacity(07782099n)

384 晶体管　EngineeringComponent /

transistor,junction_transistor(03530910n)

385 物理　OrganicObject /

physical(01710523a)

386 轴　ShapeAttribute /

axis(04600065n)

387 核　BodyPart / SubjectiveAssessmentAttribute

kernel,meat(09425258n)

kernel,substance,core,center,essence,gist,heart,inwardness,marrow,meat,nub,pith,sum,nitty-gritty(04546812n)

388 垂直　Vertical /

vertical,perpendicular(01182362a)

389 油　Liquid /

oil(10738422n)

390 拉　Guiding /

pull,draw,force(00993032v)

391 因数　RealNumber /

coefficient(09762118n)

392 长度　LengthMeasure /

length(03995742n)

393 仪器　Device /

instrument(02859502n)

394 相关　RelationExtendedToQuantities /

correlation,correlational_statistics(04616407n)

395 设置　Region /

setting,scene(06316112n)

396 格　Artifact / Artifact

shelf(03314483n)

shelf(03314483n)

397 分布式 FinancialTransaction /

distributed(00746303s)

398 差　equal /

difference(03731027n)

399 可变　SubjectiveAssessmentAttribute /

variable(02384171a)

400 电动机 Machine /

motor(03017435n)

401 帧　Artifact /

framework,frame,framing(02720094n)

402 搜索　Pursuing /

search,searching,hunt,hunting(00604336n)

403 辐射　Radiating /

radiation(07828438n)

404 容量　VolumeMeasure / VolumeMeasure

capacity,content(09924294n)

capacity,content(09924294n)

405 圆　ShapeAttribute /

circle(09995464n)

406 异步　Radiating /

asynchronous(02268481a)

407 语音　Disseminating /

address,speech(05404801n)

408 驱动器　Machine / Device

drive(02605312n)

drive(02605125n)

409 指示　Icon /

indication(05090749n)

410 补偿　Giving /

compensation(09559232n)

411 增益　experiencer / FunctionQuantity

derive,gain(01567251v)

amplification,gain(03984926n)

412 杂音　RadiatingSound /

noise(05499597n)

413 校验　IntentionalPsychologicalProcess /

check,check_up_on,look_into,check_out,suss_out,check_over,go_over,check_into(00445764v)

414 计算机辅助　/

415 台　Building /

station(03404271n)

416 绝对　SubjectiveAssessmentAttribute /

absolute(00005515a)

417 内存　Remembering /

memory(04557437n)

418 规划　Planning /

planning(00741673n)

419 记忆　hasPurpose /

mnemonic,mnemotechnic,mnemotechnical(02676785a)

420 光电　/

421 距离　distance /

distance(03967504n)

422 彩色　ColorAttribute /

color,colour,coloring,colouring(03875475n)

423 时钟　Device /

clock(02452507n)

424 维护　Maintaining /

care,maintenance,upkeep(00171721n)

425 定位　located /

location(00014887n)

426 转移　Transportation /

transportation,transfer,transferral,conveyance(00204553n)

427 导　part / part

layer(02913016n)

layer(02913016n)

428 屏幕　Device /

screen,silver_screen,projection_screen(03287008n)

429 穿孔 Device /

perforation(03107644n)

430 表面 surface /

surface(03447223n)

431 材料 Substance /

material,stuff(10446867n)

432 策略 Plan /

scheme,strategy(04538868n)

433 扩充 SubjectiveAssessmentAttribute /

drawn-out,extended,lengthy,prolonged,protracted(01381736s)

434 引导 Process / Process

boot,reboot,bring_up(00066660v)

boot,reboot,bring_up(00066660v)

435 信道 instrument /

channel,transmission_channel(04733874n)

436 移动 Motion /

apparent_motion,motion,apparent_movement,movement(07775426n)

437 像 /

438 显示器 Device /

display(02580665n)

439 串行 SubjectiveAssessmentAttribute /

consecutive,sequent,sequential,serial,successive(01608832s)

440 可靠性 TraitAttribute /

dependability,dependableness,reliability,reliableness(03680488n)

441 光学 Device /

optical(02650392a)

442 初始 starts /

initial(00959792s)

443 电视 Communication /

television,telecasting,TV,video(04745188n)

444 优化 IntentionalProcess /

optimize(00791428v)

445 适配器 Device /

adapter,adaptor(02165316n)

446 软 ShapeAttribute / ShapeAttribute

soft(01104721a)

soft(01104721a)

447 向量 Quantity /

vector(04513190n)

448 实验室 StationaryArtifact /

lab,laboratory,science_lab(02898427n)

449 业务 IntentionalProcess /

service(00379388n)

450 插 /

451 操作员 hasSkill /

operator,manipulator(07444457n)

452 压力 FunctionQuantity /

pressure,force_per_unit_area(07825463n)

453 速 SpeedFn /

speed,velocity(10978183n)

454 磁性 Radiating /

magnetism,magnetic_attraction,magnetic_force(07815486n)

455 记录器 Device / Device

recorder,recording_equipment,recording_machine(03219834n)

recorder,recording_equipment,recording_machine(03219834n)

456 密度 DensityFn /

density,denseness(03865563n)

457 宏 Directing /

macro(04941766n)

458 速率 TimeDependentQuantity /

rate(10980504n)

459 杆 Artifact /

lever(02918897n)

460 旋转 Motion /

rotation(00219234n)

461 二次 ShapeAttribute /

quadratic(02847760a)

462 负 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

negative(01747801a)

negative(01747801a)

463 操作系统 ComputerProgram /

operating_system,OS(04929621n)

464 字体 Character /

font,fount,typeface,face(05113097n)

465 阀 Device /

valve(03565876n)

466 边 superficialPart /

edge,border(06314081n)

467 启动　Process /

begin,get,start_out,start,set_about,set_out,commence(00239960v)

468 局　StationaryArtifact /

office(03052755n)

469 晶体　ElementalSubstance /

crystal(02529438n)

470 卫星　Artifact /

satellite,artificial_satellite(03275905n)

471 相对　SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

relative(00007066a)

reverse,contrary,opposite(09984290n)

472 计数器　Device /

counter,tabulator(02508575n)

473 多重　Collection /

multiple(02112966a)

474 发生器　Machine /

generator(02752377n)

475 调变　Communication /

modulation(04747347n)

476 芯片　EngineeringComponent /

chip,microchip,micro_chip,silicon_chip(02432404n)

477 独立　SubjectiveAssessmentAttribute /

independent(00691708a)

478 反射　considers /

contemplation,meditation,reflection,reflexion,rumination,musing,thoughtfulness(04459998n)

479 弧　Radiating /

discharge,spark,arc,electric_arc,electric_discharge(07837571n)

480 匹配　SubjectiveAssessmentAttribute /

duplicate,matching,twin(a),twinned(01425569s)

481 视　/

482 对象　CorpuscularObject /

object,physical_object(00009457n)

483 校正　Comparing /

calibration,standardization,standardisation(00645793n)

484 电报　Device /

telegraph,telegraphy(03477807n)

485 编程　Planning /

scheduling,programming(00741877n)

486 层次　Classifying / Classifying

hierarchical,hierarchal,hierarchic(01152594a)

hierarchical,hierarchal,hierarchic(01152594a)

487 菜单　Text /

menu,computer_menu(04873746n)

488 判定　Selecting /

decision,determination,conclusion(00105760n)

489 语法　FieldOfStudy /

grammar(04686855n)

490 损失　ContestAttribute / SubjectiveAssessmentAttribute

loss(00039700n)

loss(09595129n)

491 失真　SubjectiveAssessmentAttribute /

distortion,deformation(05530333n)

492 高速　Motion /

high-speed(00927159s)

493 电晶体　EngineeringComponent /

transistor,junction_transistor(03530910n)

494 数学　FieldOfStudy /

mathematics,math,maths(04597590n)

495 自动化　Process / Process

automation,mechanization,mechanisation,high_technology,high-tech(00064182n)

automation,mechanization,mechanisation,high_technology,high-tech(00064182n)

496 标号　IntentionalProcess /

label(00697348v)

497 诊断　DiagnosticProcess /

diagnostic(02740572a)

498 平面　ShapeAttribute /

plane,sheet(09985988n)

499 道　BodyVessel /

duct,canal,channel(04077287n)

500 放　Process /

put,set,place,pose,position,lay(01026409v)

501 缓冲器　Device /

buffer,fender(02345631n)

502 闸　Artifact /

gate(02747160n)

503 动作　IntentionalProcess /

action(00022246n)

504 寻址　uniqueIdentifier /

address,computer_address(04792458n)

505 指针　Device /

pointer(03151315n)

506 调用　Directing /

call(04939242n)

507 离子　Atom /

ion(06689819n)

508 中间　ListOrderFn /

intermediate(00963005a)

509 基于　/

510 逆　ReciprocalFn /

inverse,reciprocal(09973100n)

511 限　Region /

enclosure(02638575n)

512 回路　EngineeringComponent /

circuit,electrical_circuit,electric_circuit(02443096n)

513 框　Artifact / Artifact

box(02323900n)

box(02323900n)

514 对话　Meeting /

session(05345420n)

515 列表　Text /

list,listing(04866394n)

516 优先　SubjectiveAssessmentAttribute /

precedence,precedency,priority(10051706n)

517 标题　Text /

title,statute_title(04784828n)

518 自由　NormativeAttribute /

free(01007914a)

519 实　True /

real,existent(01864936a)

520 复制　copy /

copy(02498378n)

521 仿真　represents / represents

simulation(00577896n)

simulation(00577896n)

522 按钮　Device /

push_button,push,button(03190702n)

523 因子　DivisionFn /

divisor,factor(09888197n)

524 汇编　Making /

assemble,piece,put_together,set_up(01137704v)

525 盒　Device /

cassette(02400775n)

526 反向　BackFn /

back,backward,backwards,rearward,rearwards(00072011r)

527 幅射　Radiating / Radiating

radiation(07828438n)

radiation(07828438n)

528 反应堆　Device /

nuclear_reactor,reactor(03046989n)

529 缺省　ContestAttribute /

default(00039334n)

530 页面　Artifact /

page(04731849n)

531 态　SocialRole /

status,position(10048803n)

532 设施　StationaryArtifact /

facility,installation(02661119n)

533 读出　PsychologicalAttribute /

sense(04388620n)

534 接地　LandArea /

land,dry_land,earth,ground,solid_ground,terra_firma(06667942n)

535 排序　Class / InternalChange

kind,sort,form,variety(04496504n)

order,ordering(00651563n)

536 重复　Text /

repetition(05311563n)

537 计数　Counting /

count(09766572n)

538 锁　Device /

lock(02936009n)

539 正常　SubjectiveAssessmentAttribute /

normal(01536896a)

540 概率　ProbabilityRelation /

probability,chance(03973089n)

541 室　Artifact / Artifact

cabinet(02363665n)

cabinet(02363665n)

542 架　Artifact /

framework,frame,framing(02720094n)

543 积　Object /

merchandise,wares,product(02987655n)

544 能量　FunctionQuantity /

energy(07794689n)

545 静电　Radiating /

electrostatic(02561172a)

546 部　StationaryArtifact /

office(03052755n)

547 每　/

548 字段　LandArea /

field(06317549n)

549 数值　TimeDuration /

value,time_value,note_value(10860945n)

550 效率　RationalNumber /

efficiency(09954508n)

551 引　/

552 共享　SocialInteraction /

sharing(09973418n)

553 栅　Device /

grate,grating(02768138n)

554 导弹　/

555 按　Touching /

press(00992533v)

556 知识　knows / knows

cognition,knowledge(00013243n)

cognition,knowledge(00013243n)

557 扩散　Combining /

diffusion(09736591n)

558 数据集　UnilateralGiving /

Doctor_of_Science,DS,ScD(05028869n)

559 最佳　SubjectiveAssessmentAttribute /

optimum,optimal(00226342s)

560 结　GeographicArea /

confluence,junction,meeting(06297988n)

561 有限　ConstantQuantity /

finite(00954730a)

562 航空　TransportationDevice /

aeronautical,aeronautic(02475203a)

563 感应　Declaring /

initiation,induction,installation(05541335n)

564 阶　Human /

advocate,proponent,exponent(07065692n)

565 轨道　path /

orbit(06345096n)

566 吸收　Putting /

absorption,soaking_up(09735650n)

567 插入　IntentionalProcess /

insert,infix,enter,introduce(00974668v)

568 入口　Artifact /

entrance,entranceway,entryway,entry,entree(02641840n)

569 屏　Region / Substance

home_plate,home,plate(02825985n)

plate(03140308n)

570 维　LengthMeasure /

dimension(03973414n)

571 阵列　Collection /

array(05955653n)

572 算术　FieldOfStudy /

arithmetic(04598167n)

573 音频　frequency /

audio,audio_frequency(03946947n)

574 分解　FieldOfStudy / Damaging

decomposition,vector_decomposition(04603963n)

decomposition,disintegration(10441246n)

575 附加　Putting /

additional,further(a),more(a)(00048219s)

576 积分　Function /

integral(04605569n)

577 二极体　EngineeringComponent /

diode,semiconductor_diode,junction_rectifier,crystal_rectifier(02574009n)

578 高度 LengthMeasure /

height,tallness(04002199n)

579 调度 Planning /

scheduling,programming(00741877n)

580 仪表 Device /

instrument(02859502n)

581 针 EngineeringComponent /

hand(02789843n)

582 短 SubjectiveAssessmentAttribute /

short(01383972a)

583 监视器 Seeing / Device

monitor,supervise,ride_herd_on(01481005v)

monitor(03010795n)

584 子系统 Collection /

subsystem(06239860n)

585 衰减 Decreasing /

decay,decline(09730726n)

586 双向 PositionalAttribute / PositionalAttribute

bidirectional(00231480a)

bidirectional(00231480a)

587 控制台 EngineeringComponent /

console(02489305n)

588 整流器 Machine /

rectifier(03221674n)

589 节 subsumesContentClass /

section,subdivision(04819171n)

590 数组　Collection /

array(05955653n)

591 阴极　EngineeringComponent /

cathode(02405602n)

592 发电机　Machine /

generator(02751999n)

593 备用　SubjectiveAssessmentAttribute /

standby(03401778n)

594 偏　part /

partial(00491270s)

595 广播　Disseminating /

broadcast(04967331n)

596 队列　Relation /

queue(04874469n)

597 分层　Classifying /

hierarchical,hierarchal,hierarchic(01152594a)

598 口　EngineeringComponent /

interface,port(02862795n)

599 调谐　Comparing / Comparing

tuning(00646064n)

tuning(00646064n)

600 计算器　Machine / Machine

calculator,calculating_machine(02368666n)

calculator,calculating_machine(02368666n)

601 管理系统　/

602 阳极　EngineeringComponent / EngineeringComponent

anode(02191374n)

anode(02191374n)

603 清除　SubjectiveAssessmentAttribute /

clear(00398231a)

604 人类　Hominid /

homo,man,human_being,human(01967203n)

605 传真　Communication /

fax,telefax,facsimile(00681381v)

606 绕组　EngineeringComponent /

coil(02468210n)

607 分离　Detaching /

disconnect(00928504v)

608 异常　SubjectiveAssessmentAttribute /

abnormal(01538655a)

609 媒体　Communication /

media,mass_media(04734700n)

610 储存　Keeping /

storage(00517908n)

611 动力　FieldOfStudy /

dynamics,kinetics(04653368n)

612 进程　IntentionalProcess /

process(01660593v)

613 浮点　 /

614 气体　Gas /

gas(10671811n)

615 一般　forall /

general(01050726a)

616 查询　Investigating /

inquiry,enquiry,research(04467719n)

617 决策　Selecting /

decision,determination,conclusion(00105760n)

618 二极管　EngineeringComponent /

diode,rectifying_tube,rectifying_valve(02574189n)

619 间接　SubjectiveAssessmentAttribute /

indirect(00733920s)

620 代　AgeGroup / TimeDuration

coevals,contemporaries,generation(06196326n)

generation(10955750n)

621 资料　Proposition /

data,information(06250971n)

622 硅　ElementalSubstance /

silicon,Si,atomic_number_14(10501078n)

623 指数　LogFn /

characteristic(05102576n)

624 管理程序　ComputerProgram /

supervisory_program,supervisor,executive_program(04937225n)

625 伪　SubjectiveAssessmentAttribute /

pseudo(01069579s)

626 边缘　superficialPart / superficialPart

edge(02621337n)

edge,border(06314081n)

627 推 Motion /

push,force(01278717v)

628 间隔 TimeInterval /

time_interval,interval(10968320n)

629 分区 Artifact /

partition,divider(03091619n)

630 数据处理 ContentDevelopment /

data_processing(09746244n)

631 限制 SubjectiveAssessmentAttribute /

confining,constraining,constrictive,limiting,restricting(01932327s)

632 调制解调器 Device /

modem(03007781n)

633 冗余 Communication /

redundancy(05305375n)

634 硬 SubjectiveAssessmentAttribute /

difficult,hard(00710342a)

635 电极 EngineeringComponent /

electrode(02630022n)

636 空气 Region / Region

atmosphere(06748803n)

atmosphere(06270068n)

637 模糊 Organism /

fuzzed,fuzzy(00212640s)

638 结束 Process /

end,terminate(00246253v)

639 离散　equal /

discrete,distinct(02036455s)

640 酸　CompoundSubstance /

acid(10463819n)

641 调节　EducationalProcess /

conditioning(04442393n)

642 小型　SubjectiveAssessmentAttribute /

compact(00468456a)

643 临界　TraitAttribute /

critical(00610721a)

644 加工　employs /

working(a),on_the_job(p)(00822819s)

645 查找　Pursuing /

search,searching,hunt,hunting(00604336n)

646 机器人 Device /

automaton,robot,golem(02226816n)

647 颜色　ColorAttribute /

color,colour,coloring,colouring(03875475n)

648 横向　Horizontal /

horizontal(01181690a)

649 栈　Putting / Device

stack(01032948v)

push-down_storage,push-down_store,stack(03191147n)

650 实验　Investigating /

experiment,experimentation(00418704n)

651 更新　Repairing /

update(00116934v)

652 地面 LandArea /

land,dry_land,earth,ground,solid_ground,terra_firma(06667942n)

653 计算机系统 Machine /

computer_system,computing_system,automatic_data_processing_system,ADP_system,ADPS(02483424n)

654 处理系统 /

655 元素 subsumesContentInstance /

component,constituent,element,factor,ingredient(04515709n)

656 运动 Motion /

movement,motion(05449557n)

657 组件 subsumesContentInstance /

component,constituent,element,factor,ingredient(04515709n)

658 玻璃 Substance /

glass(10675296n)

659 屏蔽 Device /

shielding(03317132n)

660 模拟器 Machine /

simulator(03339956n)

661 系 Collection /

system(06239374n)

662 束 Icon /

radio_beam,beam(05086966n)

663 棒 Artifact /

rod(03247107n)

664 序 Collection /

series(06246946n)

665 介质 Substance / instrument

insulator,dielectric,nonconductor(10633743n)

medium(04730097n)

666 通信系统 Device /

communication_system(02477478n)

667 增量 Quantity /

increase,increment(03983965n)

668 放电 Radiating /

discharge,spark,arc,electric_arc,electric_discharge(07837571n)

669 印刷 Writing /

print,printed(02174373s)

670 液 /

671 连接器 connects /

connection,connexion,connector,connecter,connective(02488140n)

672 微处理机 EngineeringComponent /

microprocessor(02995539n)

673 商业 Corporation /

business,concern,business_concern,business_organization(06008236n)

674 编 Motion /

pass,go_through,go_across(01401176v)

675 合成 ChemicalSynthesis /

synthesis(09689446n)

676 振荡 StateChange /

oscillation(09673249n)

677 静 Dead /

dead(00098580a)

678 报文　Proposition / Proposition

message(04729537n)

message,content,subject_matter,substance(04949838n)

679 产生器　Machine /

generator(02752377n)

680 规　Device /

gauge,gage(02748539n)

681 锁定　Keeping /

locking,lockup(00530172n)

682 波道　instrument /

channel,transmission_channel(04733874n)

683 文档　Text /

text_file,document(04886842n)

684 减　Quantity /

decrease,decrement(03985095n)

685 方向　orientation / orientation

direction,way(06389665n)

direction,way(06389665n)

686 原理　Proposition /

principle,rule(04543461n)

687 飞行　Group /

flight(06102009n)

688 装配　Collection /

assembly(02217607n)

689 比例　RationalNumber /

ratio(09953487n)

690 调试 ContentDevelopment /

debug(00136974v)

691 传感器 Device / Device

detector,sensor(02558982n)

detector,sensor(02558982n)

692 极限 SubjectiveAssessmentAttribute /

limit,bounds,boundary(03992325n)

693 起始 starts /

initial(00959792s)

694 应用程序 ComputerProgram /

application,application_program,applications_programme(04931025n)

695 表达式 ShapeAttribute /

expression,look,aspect,facial_expression,face(03687456n)

696 活 inhabits /

dwell,shack,reside,live,inhabit,people,populate(01809405v)

697 电动 Radiating /

electric,electrical(02622421a)

698 标识 Classifying /

designation,identification(00098857n)

699 监视 Seeing /

monitoring(00565252n)

700 工程师 Position /

engineer,applied_scientist,technologist(06945718n)

701 激光 Device /

laser,optical_maser(02907851n)

702 预测　Predicting /

prediction,anticipation,foresight(04453620n)

703 加速　FunctionQuantity /

acceleration(03950360n)

704 电阻器　EngineeringComponent / EngineeringComponent

resistor,resistance(03232583n)

resistor,resistance(03232583n)

705 通路　StationaryArtifact /

road,route(03243979n)

706 代数　FieldOfStudy /

algebra(04603367n)

707 端口　EngineeringComponent /

interface,port(02862795n)

708 交流　Metal /

actinium,Ac,atomic_number_89(10478091n)

709 柱　SubjectiveAssessmentAttribute / ShapeAttribute

pillar(04543711n)

column,tower,pillar(10017100n)

710 界　Region / superficialPart

boundary,bound,bounds(06279059n)

boundary,edge,bound(10018921n)

711 集合　Collection / Collection

set(05990251n)

set(05990251n)

712 电平　ConstantQuantity / ConstantQuantity

degree,grade,level(03973686n)

degree,grade,level(03973686n)

713 节点　Region / connects

node(06269762n)

node(10024693n)

714 振动　Motion / Motion

vibration,quiver,quivering(00221734n)

vibration,quiver,quivering(00221734n)

715 维修　Maintaining / Maintaining

care,maintenance,upkeep(00171721n)

care,maintenance,upkeep(00171721n)

716 电荷　ViolentContest / FunctionQuantity

charge,bear_down(00760988v)

charge,electric_charge(07783247n)

717 链接　IntentionalPsychologicalProcess / IntentionalPsychologicalProcess

associate,tie_in,relate,link,connect(00482838v)

associate,tie_in,relate,link,connect(00482838v)

718 交替　capability / capability

alternation(03723137n)

alternation(03723137n)

719 编译程序　ComputerProgram / ComputerProgram

compiler,compiling_program(04932870n)

compiler,compiling_program(04932870n)

720 绘图　Icon / Making

plot(05252935n)

drawing,draftsmanship,drafting(00598605n)

721 接收器　Device / Device

receiver,receiving_system(03217746n)

receiver,receiving_system(03217746n)

722 进位　Transportation / Transfer

transport,carry(00994853v)

carry(01373583v)

723 电力　/

724 反应　capability / BiologicalAttribute

reactive(01860809a)

reactive(02030070s)

725 释放　BodyMotion / Removing

let_go_of,let_go,release(01010821v)

free,liberate,release,unloose,loose(01648022v)

726 抽样　Selecting / Selecting

sampling(00105247n)

sampling(00105247n)

727 白　White / White

white,achromatic(00367747a)

white,achromatic(00367747a)

728 简单　SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

simple(02090222a)

simple(02090222a)

729 分散　Motion / Guiding

dispersion,scattering(05463669n)

disperse,dissipate,dispel,break_up,scatter(01387651v)

730 全局　SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

global,planetary,world(a),worldwide(01512420s) global,planetary,world(a),worldwide(01512420s)

731 钢 Mixture / Mixture

steel(10618145n)

steel(10618145n)

732 起动 Maneuver / Process

start,starting(00154220n)

initiation,founding,foundation,institution,origination,creation,instauration(00153747n)

733 灵敏度 capability / capability

sensitivity,sensitiveness,sensibility(04372027n)

sensitivity(03920221n)

734 推理 Reasoning / Reasoning

inference,illation(04453321n)

inference,illation(04453321n)

735 瞬时 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

instantaneous,instant(a)(00927884s)

instantaneous,instant(a)(00927884s)

736 左 Left / Left

left(01960054a)

left(01960054a)

737 信息处理 /

738 磁头 BodyPart / EngineeringComponent

head,caput(04290247n)

read/write_head,head(03216499n)

739 拨号 Selecting / Selecting

dial(00536527v)

dial(00536527v)

740 几何　FieldOfStudy / FieldOfStudy

geometry(04598408n)

geometric,geometrical(02571073a)

741 置　Region / Process

setting,scene(06316112n)

put,set,place,pose,position,lay(01026409v)

742 对称　ShapeAttribute / ShapeAttribute

symmetrical,symmetric(02260614a)

symmetrical,symmetric(02260614a)

743 探测　Learning / Learning

determination,finding(00098543n)

determination,finding(00098543n)

744 情报　Text / Proposition

information,info(04977171n)

news,intelligence,tidings,word(04982831n)

745 错　False / False

incorrect,wrong(00596241a)

incorrect,wrong(00596241a)

746 工作站 Machine / Machine

workstation(03632472n)

workstation(03632472n)

747 可视　Seeing / capability

ocular,optic,optical,visual(02656107a)

ocular,visual(02395415s)

748 名称　names / names

name(04778525n)

name(04778525n)

749 差分　equal / SubtractionFn

difference(03731027n)

difference(04513036n)

750 排列　Collection / Collection

array(05955653n)

array(05955653n)

751 坐标　ShapeAttribute / ShapeAttribute

coordinate(04602264n)

coordinate(04602264n)

752 定时　time / time

timing(03939250n)

timing(03939250n)

753 增　Increasing / Increasing

increase(00107371v)

increase(00107371v)

754 子程序　ComputerProgram / ComputerProgram

routine,subroutine,subprogram,procedure,function(04938902n)

routine,subroutine,subprogram,procedure,function(04938902n)

755 制造　Manufacture / Manufacture

industry,manufacture,manufacturing(00592801n)

industry,manufacture,manufacturing(00592801n)

756 集成电路　EngineeringComponent /

integrated_circuit,microcircuit(02861559n)

757 大小　ConstantQuantity / ConstantQuantity

size(03977358n)

size(03977358n)

758 处理程序 Position / Position

coach,manager,handler(07162467n)

coach,manager,handler(07162467n)

759 十进制 Number / Number

decimal,denary(01845387s)

decimal,denary(01845387s)

760 软盘 Device / Device

diskette,floppy,floppy_disk(02579698n)

diskette,floppy,floppy_disk(02579698n)

761 抑制 PoliticalProcess / prevents

suppression,curtailment(00744204n)

suppression,crushing,quelling,stifling(00697514n)

762 浮动 Motion / capability

floating(01503765s)

floating(01460651s)

763 化学 FieldOfStudy / FieldOfStudy

chemistry,chemical_science(04640400n)

chemical,chemic(02543445a)

764 边界 Region / Region

boundary,bound,bounds(06279059n)

boundary,bound,bounds(06279059n)

765 导体 Device / Substance

conductor(02485878n)

conductor(10633272n)

766 照明　FunctionQuantity / Radiating

illuminance,illumination(03919344n)

light,illume,illumine,light_up,illuminate(00200741v)

767 宽度　LengthMeasure / LengthMeasure

width,breadth(04001445n)

width,breadth(04001445n)

768 隔离　PsychologicalAttribute / Putting

isolation(10343201n)

isolation,closing_off(00778860n)

769 停止　IntentionalProcess / IntentionalProcess

stop,halt(01270990v)

stop,halt(01270990v)

770 正文　Text / Text

text,textual_matter(04816275n)

text(04816802n)

771 监控　Seeing / Seeing

monitoring(00565252n)

monitoring(00565252n)

772 可编程　/

773 位址　uniqueIdentifier / uniqueIdentifier

address,computer_address(04792458n)

address,computer_address(04792458n)

774 火　RadiatingLight / RadiatingLight

fire(05444579n)

fire(05444579n)

775 语义　FieldOfStudy / FieldOfStudy

semantic(02635052a)

semantic(02635052a)

776 修改　Process / Process

alteration,modification,adjustment(00131083n)

modify(00116292v)

777 需求　NormativeAttribute / NormativeAttribute

requirement,demand(04528955n)

requirement,demand(04528955n)

778 标识符 ContentBearingObject / ContentBearingObject

ID,I.D.(05153252n)

ID,I.D.(05153252n)

779 饱和　Putting / SaturationAttribute

permeation,pervasion,impregnation,saturation,suffusion(09737023n)

saturated,concentrated(00722061a)

780 激　IntentionalPsychologicalProcess / Increasing

excite(01210207v)

stimulate,excite(00350457v)

781 切　ShapeAttribute / ShapeAttribute

secant(09993712n)

secant(09993712n)

782 谱　Radiating / Radiating

spectrum(07772246n)

spectrum(07772246n)

783 应答　Communication / Communication

answer,reply,respond(00554194v)

answer,reply,respond(00554194v)

784 射线　ShapeAttribute / RadiatingLight

ray(10026344n)

beam,beam_of_light,light_beam,ray,ray_of_light,shaft,shaft_of_light(07777807n)

785 负荷　Object / ConstantQuantity

load,loading,burden(02934345n)

load,loading(09918904n)

786 导航　Driving / Driving

navigation,pilotage,piloting(00521429n)

navigation,pilotage,piloting(00521429n)

787 估计　Certificate / Calculating

appraisal,estimate,estimation(04901909n)

estimate,estimation,approximation,idea(04471938n)

788 放大　Increasing / Increasing

amplification,elaboration(00238624n)

expansion,enlargement(00235235n)

789 权　SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

significance(04026719n)

significance(04026719n)

790 映射　Function / Function

mapping,map,correspondence(09927753n)

mapping,map,correspondence(09927753n)

791 选择器　Device / Device

selector,selector_switch(03299534n)

selector,selector_switch(03299534n)

792 图标　Icon / Icon

icon(05425814n)

icon(05425814n)

793 常式 ComputerProgram / SubjectiveAssessmentAttribute

routine,subroutine,subprogram,procedure,function(04938902n)

routine,modus_operandi(00662121n)

794 电位 capability / capability

potential,potentiality,potency(10389291n)

potential,potentiality,potency(10389291n)

795 翻译 Text / Process

translation,interlingual_rendition,rendering,version(04787112n)

transformation,translation(00255421n)

796 选项 Permission / SubjectiveAssessmentAttribute

option(09526732n)

option,alternative,choice(04463475n)

797 引线 result / Device

leave,result,lead(01798819v)

lead,leading(02913930n)

798 接收机 Device / Device

receiver,receiving_system(03217746n)

receiver,receiving_system(03217746n)

799 微分 equal / Number

differential(02557540a)

differential(02687153a)

800 界面 ComputerProgram / ComputerProgram

interface,user_interface(04934365n)

interface,user_interface(04934365n)

801 文字 Character / Text

letter,letter_of_the_alphabet,alphabetic_character(05115901n)

text,textual_matter(04816275n)

802 系统设计 /

803 栅极 Abstract / EngineeringComponent

grid,reference_grid(02771524n)

grid,control_grid(02771323n)

804 步 TimeInterval / TimeInterval

phase,stage(10983365n)

phase,stage(10983365n)

805 偶 Clothing / Set

doublet(02594317n)

pair,brace(05982514n)

806 分支 Organization / ShapeAttribute

branch,subdivision,arm(06212528n)

furcation,bifurcation,forking(10005393n)

807 词 Word /

word(04750884n)

808 当前 TimePosition /

current(00628521a)

809 删除 Process /

delete,cancel(01066739v)

810 谐振 RadiatingSound / Radiating

resonant(01939453s)

resonance(07814762n)

811 根 MultiplicationFn /

root(09886750n)

812 编译　Text /

compilation,digest(04946463n)

813 波形　ShapeAttribute / ShapeAttribute

wave_form,waveform,wave_shape(05474508n)

wave_form,waveform,wave_shape(05474508n)

814 电机　EngineeringComponent /

dynamo(02619091n)

815 指标　Icon /

beacon,beacon_fire(05421212n)

816 高频　frequency /

high_frequency,HF(03948032n)

817 公式　SymbolicString /

formula,expression(05046501n)

818 绕　Covering /

wrap,wrap_up(00876484v)

819 自然　Attribute /

nature(03716446n)

820 指定　confersNorm /

delegate,designate,depute,assign(01629170v)

821 复位　IntentionalProcess /

reset(00641718v)

822 分段　component /

section,segment(03296757n)

823 强度　ConstantQuantity /

intensity,strength(03978095n)

824 触点　LandArea /

peak,crown,crest,top,tip,summit(06349777n)

825 发动机 Machine /

engine(02639670n)

826 传播 Communication /

propagation,extension(04729401n)

827 加法 Putting /

addition,improver(02165603n)

828 引用 Text / uses

citation,credit,reference,mention,quotation(05068012n)

reference,consultation(00820543n)

829 主机 Machine /

mainframe,mainframe_computer(02959302n)

830 整流 SubjectiveAssessmentAttribute /

correction,correcting,rectification(00166347n)

831 定律 Law /

law,jurisprudence(06243906n)

832 约束 EngineeringComponent /

tie,tie_beam(03502824n)

833 微音器 Device /

microphone,mike(02995215n)

834 纳 /

835 原始 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

crude,primitive,rude(00779258s)

crude,primitive,rude(00779258s)

836 镜 SubjectiveAssessmentAttribute /

scope,range,reach,orbit,compass,ambit(03993027n)

837 外围　PositionalAttribute /

peripheral(02677239a)

838 转储　Removing /

dump(01522508v)

839 返回　Motion /

return,go_back,get_back,come_back(01367732v)

840 检波器 Device /

detector,sensor(02558982n)

841 定向　orientation /

oriented,orientated(01622433a)

842 转接　Substituting /

switch,switching,shift(00128676n)

843 光标　Icon /

cursor,pointer(02536035n)

844 排队　BodyMotion /

line_up,queue_up,queue(01391776v)

845 圈　RadiatingSound / Artifact

ring,peal(01493348v)

hoop,ring(02829005n)

846 询问　Questioning /

question,inquiry,enquiry,query(05376578n)

847 编号　Number /

number,figure(03990394n)

848 铜　Metal /

copper,Cu,atomic_number_29(10484763n)

849 纵向　Vertical /

vertical,perpendicular(01182362a)

850 例程 ComputerProgram /

routine,subroutine,subprogram,procedure,function(04938902n)

851 出错 SubjectiveAssessmentAttribute /

mistake,error,fault(00042411n)

852 链路 EngineeringComponent /

link,data_link(02930140n)

853 程序库 Collection /

library,program_library(05980872n)

854 递归 Proposition /

recursive(00096467s)

855 登记 OrganizationalProcess /

registration,enrollment,enrolment(00027235n)

856 泵 Device /

pump(03186409n)

857 替换 Substituting /

substitution,permutation,transposition,replacement,switch(05536541n)

858 广义 forall / forall

general(01050726a)

general(01050726a)

859 照相 Icon /

photograph,photo,exposure(03113185n)

860 准 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

quasi(a)(01995049s)

quasi(a)(01995049s)

861 隙 Device /

aperture(02200211n)

862 串联　Collection /

series(06246946n)

863 额定　Comparing /

evaluation,valuation,rating(04428566n)

864 正向　/

865 训练　EducationalProcess /

training,preparation,grooming(00574678n)

866 原子　Atom /

atomic(02667673a)

867 直流　Radiating /

direct_current,DC(07790907n)

868 光谱　Radiating /

spectrum(07772246n)

869 整数　Integer /

integer,whole_number(09884552n)

870 球　BodyPart / Artifact

bulb(09422963n)

bulb(02349720n)

871 编辑器 Position /

editor(07234892n)

872 峰值　SubjectiveAssessmentAttribute /

extremum,peak(09817383n)

873 覆盖　EngineeringComponent /

sheathing,overlay,overlayer(03311911n)

874 尺寸　ConstantQuantity /

size(03977358n)

875 因素　BiologicallyActiveSubstance /

agent(06665737n)

876 除　Organization /

division(06095787n)

877 顶　Abstract /

vertex(04514573n)

878 报警　Communication /

warning(05395080n)

879 传递　Transportation /

transportation,transfer,transferral,conveyance(00204553n)

880 基准　NormativeAttribute /

benchmark(05419027n)

881 失效　SubjectiveAssessmentAttribute /

failure(00038702n)

882 等效　equal /

equality,equivalence,par(10049536n)

883 解释　Text /

interpretative,interpretive(01271601s)

884 平行　SpatialRelation /

parallel(01655462a)

885 电抗　Process / Process

reactance(07832476n)

reactance(07832476n)

886 保留　GeographicArea /

reservation,reserve(06332265n)

887 本地　City /

local(01056664a)

888 宽　SubjectiveAssessmentAttribute /

wide,broad(02436517a)

889 改进　SubjectiveAssessmentAttribute /

improvement(00159157n)

890 分析程序　Device /

analyzer,analyser(02187303n)

891 紧急　SubjectiveAssessmentAttribute /

emergency,exigency,pinch(05520260n)

892 缓冲区　Device / Device

buffer,buffer_storage,buffer_store(02345884n)

buffer,buffer_storage,buffer_store(02345884n)

893 数传　Device /

digital(02557778a)

894 控制程序　EngineeringComponent /

control,controller(02492359n)

895 只读存储器　Device /

read-only_memory,ROM,read-only_storage,fixed_storage(03216271n)

896 可用　capability /

serviceable(02048503a)

897 联结　IntentionalPsychologicalProcess /

associate,tie_in,relate,link,connect(00482838v)

898 右　NormativeAttribute /

right(04030305n)

899 构造　Making /

construction,building(00585093n)

900 遥控　Device /

remote_control,remote(03228452n)

901 数据系统　Machine /

data_system,information_system(02546349n)

902 螺旋　Attaching /

screw(00925560v)

903 电磁　Radiating /

electromagnetism(07815796n)

904 阻尼　FinancialTransaction /

amortize,amortise(01606528v)

905 矢量　Quantity /

vector(04513190n)

906 方案　Plan /

scheme,strategy(04538868n)

907 触发器　Process /

interchange,switch,alternate,flip,flip-flop(00081068v)

908 环路　TimeInterval /

cycle,rhythm,round(10981743n)

909 双极　part /

bipolar(02356472a)

910 投影　Predicting /

projection(04453784n)

911 假　SubjectiveAssessmentAttribute /

pseudo(01069579s)

912 阅读　Reading /

read(00423416v)

913 触 Touching / Touching

touch(00820743v)

touch(00820743v)

914 关键字 Word /

key_word(04873147n)

915 丝 Fabric / Fabric

silk(03338120n)

silk(03338120n)

916 外壳 Weapon /

shell(03314895n)

917 会话 Meeting /

session(05345420n)

918 差动 Function /

derived_function,derivative,differential_coefficient,differential,first_derivative(04605085n)

919 发射机 Device /

transmitter(03531817n)

920 氧化 ChemicalProcess / ChemicalProcess

oxidation,oxidization,oxidisation(09684568n)

oxidation,oxidization,oxidisation(09684568n)

921 无线 Communication /

radio,radiocommunication,wireless(04745080n)

922 斜 Process /

skew(00325408v)

923 磁铁 Metal /

magnet(02954440n)

924 焊　Attaching /

bonding,soldering(00096699n)

925 极化　Radiating /

polarization,polarisation(07824381n)

926 环形　ShapeAttribute / ShapeAttribute

annular,annulate,annulated,circinate,circular,ringed,ring-shaped(01969290s)

annular,annulate,annulated,circinate,circular,ringed,ring-shaped(01969290s)

927 储存器　Position /

curator,conservator(07196812n)

928 评估　Comparing /

appraisal,assessment(04426384n)

929 图表　Text /

schedule(04874965n)

930 漏　Motion /

drain,run_out(01417557v)

931 堆　Collection /

pile,heap,mound(05969773n)

932 固态　partiallyFills /

solidity(10369372n)

933 利　SubjectiveAssessmentAttribute /

advantage,vantage(04016440n)

934 溢出　Motion / Motion

overflow,overrun,well_over,run_over,brim_over(01418065v)

overflow,overrun,well_over,run_over,brim_over(01418065v)

935 尾　BodyPart /

tail(01665222n)

936 分割　Artifact /

partition,divider(03091619n)

937 黑　Black /

black,achromatic(00367459a)

938 失败　SubjectiveAssessmentAttribute /

failure(00038702n)

939 分组　Collection /

package,bundle,packet,parcel(05997364n)

940 装入　Putting /

load,charge(01022920v)

941 真空　/

942 位移　Substituting /

supplanting,displacement(00129660n)

943 栏　Group /

group,grouping(00017954n)

944 磁道　path /

path,track,course(06709272n)

945 确认　SubjectiveAssessmentAttribute /

recognition,acknowledgment,acknowledgement(10341471n)

946 等待　located /

wait(01800838v)

947 中子　Neutron /

neutron(06681284n)

948 光盘　ContentBearingObject / ContentBearingObject

compact_disk,compact_disc,CD(02478312n)

compact_disk,compact_disc,CD(02478312n)

949 单向 PositionalAttribute /

unidirectional(00232145a)

950 波导 Device /

waveguide,wave_guide(03600178n)

951 损耗 Destruction /

dissipation(05463938n)

952 调节器 Device /

regulator(03226504n)

953 飞行器 TransportationDevice /

craft(02515336n)

954 合金 Mixture /

alloy(10449557n)

955 格式化 ContentDevelopment /

format,formatting,data_format,data_formatting(04979062n)

956 炉 Device /

furnace(02728880n)

957 立即 Adjacent /

contiguous,immediate(00418121s)

958 规范 Plan /

specification,spec(05042568n)

959 学会 IntentionalProcess /

establish,found,plant,constitute,institute(01131591v)

960 再生 Growth /

regeneration(09697656n)

961 非线性 Relation /

nonlinear(01360891a)

962 传导 Radiating /

conduction,conductivity(07837945n)

963 电枢 Device / Device

armature(02209201n)

armature(02209201n)

964 器件 Device /

device(02560468n)

965 字符串 Text /

string,string_of_words,word_string,linguistic_string(05261517n)

966 谐波 RadiatingSound /

harmonic(04418581n)

967 偏转 SubjectiveAssessmentAttribute /

deflection,warp(10405415n)

968 报表 Text /

inventory,stock_list(04873270n)

969 高压 Object /

high,high_pressure(10414486n)

970 流量 TimeDependentQuantity /

flux(10975486n)

971 膜 Tissue /

membrane,tissue_layer(04214205n)

972 模态 manner /

manner,mode,style,way,fashion(03856995n)

973 特殊 SubjectiveAssessmentAttribute /

particular(a),peculiar(a),special(a)(01054295s)

974 上下文 Text /

context,linguistic_context,context_of_use(04749988n)

975 表示法 ContentBearingObject / represents

notation,notational_system(05098604n)

representation,mental_representation,internal_representation(04550469n)

976 容许 NormativeAttribute / NormativeAttribute

permission(05015182n)

permission(05015182n)

977 描述符 /

978 管理器 Position /

coach,manager,handler(07162467n)

979 散射 Separating /

break_up,disperse,scatter(00228848v)

980 固有 SubjectiveAssessmentAttribute /

built-in,constitutional,inbuilt,inherent,integral(01294559s)

981 空白 not /

blank,clean,white(01035211s)

982 馈 Giving /

feed(00803617v)

983 停机 IntentionalProcess /

stop,halt(01270990v)

984 击 ViolentContest /

beating,whipping(00762001n)

985 秒 ListOrderFn /

second,2nd,2d(02104904s)

986 塔　Building /

tower(03523510n)

987 日　Day /

day,twenty-four_hours,solar_day,mean_solar_day(10876395n)

988 评价　Comparing /

evaluation,valuation,rating(04428566n)

989 塞　Device / Device

water_faucet,water_tap,tap,spigot,hydrant(03595967n)

tap,spigot(03469972n)

990 应急　SubjectiveAssessmentAttribute /

emergency,exigency,pinch(05520260n)

991 神经　Tissue /

nerve,nervus(04244555n)

992 事务　Transaction /

transaction,dealing,dealings(00717352n)

993 输送　Giving /

feed(00803617v)

994 注册　Communication /

log_in,log_on,login(01539216v)

995 间距　SpatialRelation /

spacing,spatial_arrangement(03967157n)

996 模块化　Artifact / Artifact

modular(02185926s)

modular(02185926s)

997 纯　CurrencyMeasure /

net,nett(01525937a)

998 实用 SubjectiveAssessmentAttribute /

practical(01762874a)

999 比率 RationalNumber /

ratio(09953487n)

1000 操作数 Quantity /

operand(04508586n)

1001 矩 FunctionQuantity /

moment(07818744n)

1002 关闭 OrganizationalProcess /

closure,closedown,closing,shutdown(00148822n)

1003 强制 causes /

coerce,pressure,force(01707058v)

1004 薄膜 Tissue /

membrane,tissue_layer(04214205n)

1005 指示符 Quantity /

index,index_number,indicant,indicator(04981133n)

1006 偏差 NormativeAttribute /

deviation,divergence,departure,difference(05485383n)

1007 表格 ContentBearingObject /

table,tabular_array(06138367n)

1008 现场 LandArea /

site,land_site(06371658n)

1009 变换器 Device / Device

converter,convertor(02494885n)

converter,convertor(02494885n)

1010 多媒体 Text /

hypermedia,hypermedia_system,interactive_multimedia,interactive_multimedia_system(04735017n)

1011 文法 FieldOfStudy /

grammar(04686855n)

1012 频带 Group /

set,circle,band,lot(06117068n)

1013 框架 Artifact /

framework,frame,framing(02720094n)

1014 操纵 causes /

manipulation,use(00102995n)

1015 二进 Number /

binary(02521946a)

1016 微型 /

1017 手册 Book / Book

manual(04840046n)

manual(04840046n)

1018 费用 CurrencyMeasure /

cost(09553349n)

1019 间隙 Hole /

opening,gap(06698176n)

1020 保存 Removing /

salvage,salve,relieve,save(01739424v)

1021 传输系统 /

1022 视觉 PsychologicalProcess /

vision(04448837n)

1023 断路器 EngineeringComponent /

circuit_breaker(02443852n)

1024 汇编程序 ComputerProgram /

assembler,assembly_program(04932604n)

1025 位准 ConstantQuantity /

degree,grade,level(03973686n)

1026 插件 EngineeringComponent / EngineeringComponent

circuit_board,circuit_card,board,card(02443613n)

circuit_board,circuit_card,board,card(02443613n)

1027 光栅 part /

raster(03213684n)

1028 备份 SocialRole /

stand-in,substitute,relief,backup,backup_man,fill-in(07606870n)

1029 互补 SubjectiveAssessmentAttribute /

complemental,complementary,completing(00858284s)

1030 口令 LinguisticExpression /

password,watchword,word,parole,countersign(05004524n)

1031 刻度 UnitOfMeasure / UnitOfMeasure

scale,scale_of_measurement,graduated_table,ordered_series(09978973n)

scale,scale_of_measurement,graduated_table,ordered_series(09978973n)

1032 正交 Calculating /

quadrature(00561436n)

1033 等级 Classifying /

rate,rank,range,order,grade,place(00443221v)

1034 大气 Region /

atmosphere(06748803n)

1035 商 RationalNumber /

quotient(09957148n)

1036 稳定性 SubjectiveAssessmentAttribute /

stability,stableness(03750498n)

1037 关联 Organization /

association(05999585n)

1038 核对 Reasoning /

confirmation,verification,check,substantiation(04486906n)

1039 电离 ChemicalProcess /

ionization(09677314n)

1040 对策 Maneuver /

game(00289585n)

1041 临时 SubjectiveAssessmentAttribute /

impermanent,temporary(01691661a)

1042 过滤器 Device /

filter(02679962n)

1043 准则 NormativeAttribute /

standard,criterion,measure,touchstone(05418696n)

1044 电感 FunctionQuantity /

inductance,induction(03913314n)

1045 补 Relation /

complementation(09938389n)

1046 例行程序 ComputerProgram / SubjectiveAssessmentAttribute

routine,subroutine,subprogram,procedure,function(04938902n)

routine,modus_operandi(00662121n)

1047 面板 EngineeringComponent /

control_panel,instrument_panel,control_board,board,panel(02493245n)

1048 波长 LengthMeasure /

wavelength(03969231n)

1049 加载 Object /

load,loading,burden(02934345n)

1050 抽 Icon /

drawing(05255398n)

1051 系统工程 /

1052 核心 BodyPart /

kernel,meat(09425258n)

1053 回波 /

1054 带宽 FunctionQuantity / FunctionQuantity

bandwidth(09758629n)

bandwidth(09758629n)

1055 多变量 /

1056 插座 EngineeringComponent /

socket(03364873n)

1057 定时器 Device /

timer(03507166n)

1058 日期 Day /

date,day_of_the_month(10880047n)

1059 宽带 capability /

broadband,wideband(02526809a)

1060 助 SubjectiveAssessmentAttribute /

help,assist,aid(01737017v)

1061 中继 Communication /

relay(00503786v)

1062 磁场 Region /

magnetic_field,magnetic_flux,flux(07814030n)

1063 观察 Measuring /

observation(00647699n)

1064 射频 frequency /

radio_frequency(03947051n)

1065 虚 SubjectiveAssessmentAttribute /

dummy(01517414s)

1066 移位 Motion /

transfer,shift(01373274v)

1067 流程图 Text /

flow_chart,flow_diagram(04938633n)

1068 扬声器 Device /

loudspeaker,speaker,speaker_unit,loudspeaker_system,speaker_system(02943222n)

1069 区间 Set /

interval(04606284n)

1070 鼠标 Rodent /

mouse(01832174n)

1071 深度 LengthMeasure /

depth(04000219n)

1072 胶 Attaching / Substance

glue,paste(00909290v)

glue,gum,mucilage(10539758n)

1073 汉字 Character /

character,grapheme,graphic_symbol(05107649n)

1074 探测器 Device / Device

finder,viewfinder,view_finder(02680856n)

finder,viewfinder,view_finder(02680856n)

1075 程序设计语言 ComputerLanguage /

programming_language,programing_language(05162548n)

1076 仿真器 /

1077 分析器 Device /

analyzer,analyser(02187303n)

1078 客户 SocialRole / SocialRole

client(07197725n)

customer,client(07197309n)

1079 轨 Permission /

bail,bail_bond,bond(09612224n)

1080 流水线 Text /

grapevine,pipeline,word_of_mouth(05394636n)

1081 字典 Book / Book

dictionary,lexicon(04837766n)

dictionary,lexicon(04837766n)

1082 关 PathologicProcess /

premenstrual_syndrome,PMS(10272175n)

1083 胶片 Device /

film,photographic_film(02679248n)

1084 电传 Device /

teletypewriter,teleprinter,teletype_machine,telex,telex_machine(03481215n)

1085 设 Declaring /

establish,set_up,found,launch(01652417v)

1086 盖 Clothing /

jacket(02870166n)

1087 规约 NormativeAttribute /

protocol,communications_protocol(04998026n)

1088 冲击 ViolentContest /

attack,onslaught,onset,onrush(00625655n)

1089 阅读器 Human /

reader(07520729n)

1090 密码 Procedure /

cipher,cypher,cryptograph,secret_code(04791545n)

1091 重叠 Covering /

overlay,cover(00912832v)

1092 冷却 Cooling /

cooling,chilling,temperature_reduction(09741080n)

1093 时序 SubjectiveAssessmentAttribute /

consecutive,sequent,sequential,serial,successive(01608832s)

1094 采样 Selecting / Selecting

sampling(00105247n)

sampling(00105247n)

1095 无源 SubjectiveAssessmentAttribute /

passive,inactive(00039964a)

1096 压电 Radiating /

piezoelectric(02745036a)

1097 芯 /

1098 算子 /

1099 合并 Combining /

unify,unite,merge(00257011v)

1100 自动机 Human /

automaton,zombi,zombie(07092577n)

1101 编辑程序 ComputerProgram /

editor_program,editor(04933998n)

1102 共用 Group /

community(06104982n)

1103 发光 RadiatingLight /

luminescence(07813148n)

1104 正规 SubjectiveAssessmentAttribute /

normal(01536896a)

1105 语义学 FieldOfStudy /

semantics(04689282n)

1106 赋值 OrganizationalProcess /

assignment,duty_assignment(00470305n)

1107 绳 Artifact /

rope(03252883n)

1108 注入 Putting /

injection(00208372n)

1109 工艺 FieldOfStudy /

technology,engineering(00607693n)

1110 验证 Reasoning /

confirmation,verification,check,substantiation(04486906n)

1111 简化 Decreasing /

decrease,diminution,reduction,step-down(00225760n)

1112 伏 UnitOfMeasure /

volt,V(09808122n)

1113 数据流 /

1114 短路 Device /

crowbar,wrecking_bar,pry,pry_bar(02525606n)

1115 键控 Process / part

attuned,keyed,tuned(00340546s)

keyed(01318088a)

1116 闪烁 RadiatingLight /

twinkle,scintillation,sparkling(05516369n)

1117 亮度 Seeing /

brightness(03873073n)

1118 允许 Process / Process

enable(00357077v)

enable(00357077v)

1119 窗 Artifact /

window(03619904n)

1120 系统分析 FieldOfStudy /

systems_analysis(04457913n)

1121 对数 LogFn /

mantissa,fixed-point_part(05102381n)

1122 支援 SubjectiveAssessmentAttribute /

support,back_up(01742587v)

1123 观测 Measuring / Measuring

observation(00647699n)

observation(00647699n)

1124 一次 SubjectiveAssessmentAttribute /

primary(01781200a)

1125 确定 Stating /

definition(05054071n)

1126 焊接 Attaching /

welding(00097334n)

1127 计算机化 Putting /

computerize(01596258v)

1128 在线 path /

on-line(01590949a)

1129 界限 Region /

boundary,bound,bounds(06279059n)

1130 普通 forall /

general(01050726a)

1131 相联 Comparing /

associative(02753035a)

1132 接点 Communication /

contact(00023118n)

1133 从属 SocialRole /

slave(07583414n)

1134 声频 RadiatingSound /

audio,sound(04745575n)

1135 字符集 Character /

character_set(04870984n)

1136 共振 Radiating /

resonance(07814762n)

1137 伺服 EngineeringComponent /

servo,servomechanism,servosystem(03304491n)

1138 偏移 SubtractionFn /

deviation(04609725n)

1139 偏压 IntentionalRelation /

bias,prejudice(04701316n)

1140 投 Impelling /

throw(01036035v)

1141 透镜 Device /

lens,lens_system(02917478n)

1142 稳态 SubjectiveAssessmentAttribute /

stable(02180646a)

1143 档案 EducationalOrganization / EducationalOrganization

archive,archives(02206789n)

archive,archives(02206789n)

1144 永久 SubjectiveAssessmentAttribute /

permanent,lasting(01690654a)

1145 计算机控制 /

1146 程序包 ComputerProgram /

software_package,software_product(03367250n)

1147 均匀 Clothing /

uniform(03560722n)

1148 保险 Obligation /

insurance,coverage(09608739n)

1149 微波 Radiating /

microwave(07818302n)

1150 变址 UnitOfMeasure /

index(09979235n)

1151 版本 Stating /

version(05362628n)

1152 调频 Communication /

frequency_modulation,FM(04747736n)

1153 安全性 SubjectiveAssessmentAttribute /

security(10426280n)

1154 穿孔机 Impacting /

punch,plug(00969752v)

1155 可见 capability /

visible,seeable(02394546a)

1156 叠 Covering /

overlay,cover(00912832v)

1157 单片 SubjectiveAssessmentAttribute /

massive,monolithic,monumental(01334723s)

1158 相互 TraitAttribute /

brooding,broody,contemplative,meditative,musing,pensive,pondering,reflective,ruminative(02301888s)

1159 操作符 Function /

operator(09927920n)

1160 档 instrument /

measure,step(00113217n)

1161 文卷 RegulatoryProcess /

file,register(00677641v)

1162 硬盘 Device /

hard_disc,hard_disk,fixed_disk(02798214n)

1163 可靠 SubjectiveAssessmentAttribute /

reliable,dependable(00688247a)

1164 幅度 ConstantQuantity /

magnitude(03971772n)

1165 电桥 /

1166 微程序 /

1167 编址 /

1168 定制 Making /

custom-make,customize,tailor1-make(01116253v)

1169 寿命 SubjectiveAssessmentAttribute /

life(10060845n)

1170 逻辑电路 /

1171 描述语言 /

1172 软件包 Collection /

package,bundle,packet,parcel(05997364n)

1173 图解 Writing /

graphic,graphical,in_writing(p)(02173797s)

1174 成批 Collection / IntentionalProcess

batch(06212088n)

batch(00950379v)

1175 分页 RadiatingSound /

paging(05333587n)

1176 无效 DiseaseOrSyndrome /

invalid,shut-in(07344066n)

1177 多项式 AdditionFn /

polynomial(04511577n)

1178 中继线 BodyPart /

trunk,tree_trunk,bole(09450405n)

1179 法则 Procedure /

rule,formula(04502282n)

1180 清单 Text /

list,listing(04866394n)

1181 铅 Metal /

lead,Pb,atomic_number_82(10490965n)

1182 电台 Building /

station(03404271n)

1183 无用信息 SelfConnectedObject / SelfConnectedObject

garbage,refuse,food_waste,scraps(10656657n)

garbage,refuse,food_waste,scraps(10656657n)

1184 湿 Damp /

wet(02425348a)

1185 修正 SubjectiveAssessmentAttribute /

correction,correcting,rectification(00166347n)

1186 接头 Cooperation /

joint(02037068a)

1187 断开 Detaching /

unplug,disconnect(00974541v)

1188 运转 Walking / SubjectiveAssessmentAttribute

run(01314495v)

run,go,pass,lead,extend(01832891v)

1189 精度 SubjectiveAssessmentAttribute /

preciseness,precision(03768956n)

1190 实用程序 ComputerProgram /

utility_program,utility,service_program(04938103n)

1191 漂移 Motion /

float,drift,be_adrift,blow(01298374v)

1192 径 Device /

aperture(02200211n)

1193 控制论 FieldOfStudy /

cybernetics(04669412n)

1194 壳 Weapon /

shell(03314895n)

1195 冲突 Impacting /

collision,hit(05444467n)

1196 插头 Device /

plug,male_plug(03146673n)

1197 耦合器 connects /

coupling,coupler(02510482n)

1198 容错 /

1199 滤波 /

1200 有效性 SubjectiveAssessmentAttribute /

effectiveness,effectivity,effectualness,effectuality(04047189n)

1201 文件系统 Procedure /

file_system,filing_system(04425644n)

1202 一致 SubjectiveAssessmentAttribute /

consistent(00543564a)

1203 近似 Calculating /

estimate,estimation,approximation,idea(04471938n)

1204 检测器 Device /

detector,sensor(02558982n)

1205 终止 TimeInterval /

termination,expiration,expiry(10967564n)

1206 池 Hole /

pool(03156540n)

1207 会计 Stating / Procedure

accounting(05051021n)

accounting(04378101n)

1208 价 CurrencyMeasure /

price,terms,damage(09576753n)

1209 微处理器 EngineeringComponent /

microprocessor(02995539n)

1210 清 Killing /

kill(00903723v)

1211 累积 /

1212 取样 Selecting /

sampling(00105247n)

1213 音调 RadiatingSound / RadiatingSound

tone(03897210n)

tone,tone_of_voice(05299832n)

1214 馈送 Giving /

feed(00803617v)

1215 定址 uniqueIdentifier /

address,computer_address(04792458n)

1216 液体 Liquid /

liquid(02152669a)

1217 定点 UnitOfMeasure /

set_point(09780858n)

1218 色彩 ColorAttribute /

color,colour,coloring,colouring(03875475n)

1219 运输 TransportationDevice /

transportation_system,transportation,transit(03532413n)

1220 成组 Organization /

gang,pack,ring,mob(06119534n)

1221 分时 uses /

time_sharing(09973537n)

1222 收敛 cooccur /

convergence(05518515n)

1223 方位 AngleMeasure /

azimuth,AZ(10011300n)

1224 盒式 Device /

cartridge(02394559n)

1225 触发 /

1226 博弈 Maneuver /

game(00289585n)

1227 按键 Device /

key(02886812n)

1228 梯度 QuantityChange / QuantityChange

gradient(09985018n)

gradient(09985018n)

1229 打字机 Device /

typewriter(03557088n)

1230 暗 RadiatingLight /

dark(00268709a)

1231 标度 UnitOfMeasure /

scale,scale_of_measurement,graduated_table,ordered_series(09978973n)

1232 制动 Confining /

collar,nail,apprehend,arrest,pick_up,nab,cop(00827485v)

1233 柔性 ShapeAttribute /

flexibility,flexibleness(03922244n)

1234 号码 Number /

number,figure(03990394n)

1235 音量 VolumeMeasure /

volume(09923970n)

1236 生成程序 Machine /

generator(02752280n)

1237 凸 ShapeAttribute /

convex,bulging(00502070a)

1238 轨迹 Region /

venue,locale,locus(06387836n)

1239 超级 /

1240 次序 List /

ordering,order(06246680n)

1241 射束 Device /

pencil(03103254n)

1242 游戏 Maneuver /

game(00289585n)

1243 导线 Device / EngineeringComponent

conductor(02485878n)

wire,conducting_wire(03625791n)

1244 臂 BodyPart /

arm(04310435n)

1245 拂掠 Motion /

brush,sweep(01277218v)

1246 医学 FieldOfStudy /

medicine(04623732n)

1247 过渡 Motion /

passage,transition(00131763n)

1248 变元 increasesLikelihood /

argument,statement(04985978n)

1249 激励 EmotionalState /

excitation,excitement(00818927n)

1250 遥测 Measuring /

telemetry(00723849n)

1251 校 /

1252 粗 SubjectiveAssessmentAttribute /

fat(00934421a)

1253 滚动 RadiatingSound / Transportation

resonant,resonating,resounding,reverberating,reverberative,rolling(01939548s)

wheeling,rolling(00077891n)

1254 试探 Procedure /

heuristic(02641525a)

1255 眼 Organ /

eye,oculus,optic,peeper(04122028n)

1256 交错 Attaching /

intertwine,twine,entwine,enlace,interlace,lace(01042567v)

1257 后台 PastFn /

background(03851300n)

1258 背景 Region /

background,desktop,screen_background(02233096n)

1259 散列 Cooking /

hash(00857210v)

1260 摄影 Device /

camera,photographic_camera(02371168n)

1261 装载 Putting /

load,lade,laden,load_up(01022755v)

1262 驱动程序 ComputerProgram / ComputerProgram

driver,device_driver(04933667n)

driver,device_driver(04933667n)

1263 喇叭 Device /

loudspeaker,speaker,speaker_unit,loudspeaker_system,speaker_system(02943222n)

1264 光纤 CorpuscularObject /

fiber,fibre(10663683n)

1265 口 Hole /

orifice,opening(04076503n)

1266 密封 capability /

hermetic(01342468s)

1267 追踪 Pursuing /

trailing,tracking(00206842n)

1268 探 BodyPart /

acerate_leaf,needle(09443827n)

1269 图案 Abstract /

form,shape,pattern(04554317n)

1270 纹 Fabric /

band,banding,stripe(02245671n)

1271 兼容 SubjectiveAssessmentAttribute /

compatible(00473660a)

1272 温度计 Device /

thermometer(03493877n)

1273 实体 Physical / Physical

entity,something(00001740n)

entity,something(00001740n)

1274 细胞 Cell /

cell(00003095n)

1275 满 SubjectiveAssessmentAttribute /

full(01030147a)

1276 附件 PsychologicalAttribute /

attachment,fond_regard(05608793n)

1277 编码器 /

1278 旁 Region /

side(06370070n)

1279 本质 needs / SubjectiveAssessmentAttribute

essential,indispensable(01524177s)

essential(00856182a)

1280 理想 SubjectiveAssessmentAttribute /

ideal(01687994s)

1281 取消 Declaring /

cancel,call_off(01688387v)

1282 抽象 Reasoning /

abstract(00468095v)

1283 间歇 SubjectiveAssessmentAttribute /

intermittence,intermittency(03745706n)

1284 布线 path /

path,route(06348591n)

1285 制表 Text /

check,chit,tab(04892596n)

1286 屏极 Artifact /

sheath(03311748n)

1287 步骤 instrument /

measure,step(00113217n)

1288 图书馆 Collection /

library(05981110n)

1289 体系结构 StationaryArtifact /

architecture(02206447n)

1290 校准 Comparing /

calibration,standardization,standardisation(00645793n)

1291 文献 Text /

bibliographic,bibliographical(02520236a)

1292 接合器 GeographicArea / connects

confluence,junction,meeting(06297988n)

junction,conjunction(02882380n)

1293 禁止 IntentionalProcess /

suppress,stamp_down,inhibit,subdue,conquer,curb(00322829v)

1294 噪音 RadiatingSound /

noise(05499597n)

1295 升 Liter /

liter,litre,l,cubic_decimeter,cubic_decimetre(09794447n)

1296 前向 /

1297 延时 TimeInterval /

delay,hold,time_lag,postponement,wait(10970463n)

1298 封锁 Keeping /

locking,lockup(00530172n)

1299 教学 EducationalProcess /

teaching,instruction,pedagogy(00570118n)

1300 量化 Separating /

quantization,quantisation(00249932n)

1301 电解 BiologicalProcess /

electrolysis(09733570n)

1302 均衡 SubjectiveAssessmentAttribute /

equalization,equilisation,leveling(00121287n)

1303 罩 Region /

enclosure(02638575n)

1304 剩余 Removing / Removing

residual,residuary(02629457a)

residual,residuary(02629457a)

1305 磁心 Group /

core,nucleus,core_group(06118015n)

1306 三维 ShapeAttribute /

three-dimensional,third-dimensional,three-d(00623140s)

1307 树脂 BodySubstance /

colophony(10687220n)

1308 日志 BodyPart /

log(10722705n)

1309 测定 Device /

analyzer,analyser(02187303n)

1310 制造商 agent /

maker,shaper(07388935n)

1311 交换机 Device /

switchboard,patchboard,plugboard(03456767n)

1312 步进 inhibits /

intervene,step_in,interfere,interpose(01730921v)

1313 笔 Device /

pen(03102377n)

1314 刀 Device /

knife(02893138n)

1315 划 Motion /

dart,dash,scoot,scud,flash,shoot(01410345v)

1316 分子 Molecule / Molecule

molecular(02680831a)

molecular(02680831a)

1317 加密 Encoding /

encoding,encryption(00405203n)

1318 前置 /

1319 平方 ShapeAttribute /

square,foursquare(09998745n)

1320 品 Article /

article(04739220n)

1321 调整器 Device /

regulator(03226504n)

1322 轻 RadiatingLight /

light,visible_light,visible_radiation(07811448n)

1323 电流计 Device /

galvanometer(02735663n)

1324 转接器 Device /

adapter,adaptor(02165316n)

1325 多工 Communication /

multiplex(04744970n)

1326 程序员 hasSkill /

programmer,computer_programmer,software_engineer(07504924n)

1327 电键 hasSkill /

operator,manipulator(07444457n)

1328 十六进制 Number /

hexadecimal,hex(02574606a)

1329 计时 TimeDuration /

clocking(10860098n)

1330 载体 Position /

carrier,bearer,toter(07141701n)

1331 低压 Device /

LP,L-P(02945458n)

1332 弹道 Motion /

ballistic(02516828a)

1333 翻译程序 ComputerProgram /

translator,translating_program(04937943n)

1334 微电脑 Machine / Machine

personal_computer,PC,microcomputer(03109520n)

personal_computer,PC,microcomputer(03109520n)

1335 闭合 finishes /

closing(00958006a)

1336 频谱 Radiating /

spectrum(07772246n)

1337 静止 Motion / Motion

dormancy,quiescence,quiescency(10093691n)

dormancy,quiescence,quiescency(10093691n)

1338 无限 Quantity /

infinite(00955446a)

1339 尺 NormativeAttribute /

rule,regulation(04501800n)

1340 宏指令 Directing /

macro(04941766n)

1341 例 instance /

example,illustration,instance,representative(04483764n)

1342 精确 SubjectiveAssessmentAttribute /

preciseness,precision(03768956n)

1343 巴 Room /

barroom,bar,saloon,ginmill,taproom(02255304n)

1344 损 SubjectiveAssessmentAttribute /

damage,harm,impairment(05522654n)

1345 许可 NormativeAttribute /

permission(05015182n)

1346 确定性 PsychologicalAttribute /

certainty(04401608n)

1347 程式 Relation /

modality,mode(09940425n)

1348 常驻 SocialRole /

resident,occupant,occupier(07530344n)

1349 库存 CurrencyMeasure /

stock(09599721n)

1350 绞 Process /

put,set,place,pose,position,lay(01026409v)

1351 集极 Human /

collector(07165763n)

1352 灯丝 CorpuscularObject /

fibril,filament,strand(10664410n)

1353 问题求解 FieldOfStudy /

problem_solving(04667394n)

1354 扫描器 Device /

scanner(03279906n)

1355 人机 /

1356 网格 Device /

power_system,power_grid,grid(03168544n)

1357 写入 Writing /

write,compose,pen,indite(01164896v)

1358 析 Investigating /

analysis(00422134n)

1359 部份 subsumesContentInstance /

component,constituent,element,factor,ingredient(04515709n)

1360 套管 Device /

cannula(02377323n)

1361 封装 Putting /

packaging(00715158n)

1362 固体 ShapeAttribute /

solid(09985749n)

1363 反应器 agent /

respondent,responder,answerer(07531167n)

1364 迭代 SubjectiveAssessmentAttribute /

iterative,reiterative,repetitious,repetitive(02387175s)

1365 完整 SubjectiveAssessmentAttribute /

holistic(01620845a)

1366 保密 TraitAttribute /

privacy,privateness,seclusion(03647446n)

1367 电脑 Machine / Machine

computer,data_processor,electronic_computer,information_processing_system(02481557n)

computer,data_processor,electronic_computer,information_processing_system(02481557n)

1368 立体 Device / Icon

stereo,stereo_system,stereophonic_system(03411597n)

stereo,stereoscopic_picture,stereoscopic_photograph(03411818n)

1369 截止 Quantity /

cutoff(09908421n)

1370 声能 /

1371 电子学 FieldOfStudy /

electronics(04644705n)

1372 链式 Collection /

chain,concatenation(06247825n)

1373 编译器 ComputerProgram /

compiler,compiling_program(04932870n)

1374 游离 ChemicalProcess /

ionize(00182024v)

1375 传动 Transfer /

transmission,transmittal,transmitting(00077376n)

1376 二级 SubjectiveAssessmentAttribute /

secondary(01782330a)

1377 扭 PositionalAttribute /

skew,skewed(01184506s)

1378 脱机 path /

off-line(01591099a)

1379 陶瓷 Substance /

ceramic(02773360a)

1380 红外线 Radiating /

infrared(02396714s)

1381 电子束 /

1382 嵌入 Putting /

insertion,introduction,intromission(00207301n)

1383 播放 Game /

play(00727813v)

1384 畸变 SubjectiveAssessmentAttribute /

distortion,deformation(05530333n)

1385 多路复用 Communication /

multiplex(04744970n)

1386 未知 knows /

unknown(01322204a)

1387 并联 SpatialRelation /

parallel(01655462a)

1388 布局 Abstract /

layout(04495038n)

1389 红外 Radiating /

infrared(02396714s)

1390 管理员 Position /

administrator,decision_maker(07063507n)

1391 变数 Quantity / Quantity

variable,variable_quantity(04508699n)

variable,variable_quantity(04508699n)

1392 汞 Metal /

mercury,quicksilver,Hg,atomic_number_80(10492641n)

1393 波段 Group /

set,circle,band,lot(06117068n)

1394 登录 Communication /

log_in,log_on,login(01539216v)

1395 调制器 /

1396 聚焦 Perception /

focus,focusing,focussing,centering(04407657n)

1397 馈电 Eating /

eating,feeding(00537892n)

1398 多点 /

1399 存取方法 /

1400 程序控制 /

1401 互换 StationaryArtifact /

interchange(02862021n)

1402 矿 /

1403 供给 Putting /

supply,provide,render,furnish(01590833v)

1404 填充 SubjectiveAssessmentAttribute / IntentionalProcess

filled(01032310s)

fill,fill_up,make_full(00315864v)

1405 开环 /

1406 初始化 ContentDevelopment /

low-level_formatting,initialization(04979478n)

1407 桌面 PositionalAttribute /

desktop(06380677n)

1408 数位 NonnegativeInteger /

digit(09893975n)

1409 映象 Function /

mapping,map,correspondence(09927753n)

1410 随机存取 /

1411 联接 SexualReproduction / IntentionalPsychologicalProcess

copulate,mate,pair,couple(00979316v)

associate,tie_in,relate,link,connect(00482838v)

1412 纸带 Device / Device

tape(03470437n)

magnetic_tape,mag_tape,tape(02956039n)

1413 读取 Reading / Process

read(00423416v)

read,scan(00425290v)

1414 帐 SocialInteraction / SocialInteraction

account,business_relationship(10036540n)

account,business_relationship(10036540n)

1415 拷贝 copy / equivalentContentClass

copy(02498378n)

transcript,copy(04882005n)

1416 示波器 Device / Device

oscilloscope,scope,cathode-ray_oscilloscope,CRO(03065395n)

oscilloscope,scope,cathode-ray_oscilloscope,CRO(03065395n)

1417 多层 /

1418 气象 Process / Process

weather,weather_condition,atmospheric_condition(07847974n)

weather,weather_condition,atmospheric_condition(07847974n)

1419 兼容性 EmotionalState / SubjectiveAssessmentAttribute

compatibility,rapport(05616177n)

compatibility(03707482n)

1420 网络系统 /

1421 延迟线 EngineeringComponent /

delay_line(02553217n)

1422 航天 Region / Region

aerospace(06271859n)

aerospace(06271859n)

1423 交互作用 Process / Process

interaction(00022878n)

interaction(00022878n)

1424 叁数 Quantity / causes

parameter,parametric_quantity(04509624n)

parameter(05461016n)

1425 电器 /

1426 建模 ArtWork / represents

mold,mould,molding,moulding,modeling,clay_sculpture(03009171n)

model,modelling,modeling(00577745n)

1427 供应 /

1428 透明 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

crystalline,crystal_clear,limpid,lucid,pellucid,transparent(00401511s)

crystalline,crystal_clear,limpid,lucid,pellucid,transparent(00401511s)

1429 体积 Book / VolumeMeasure

book,volume(02313137n)

volume(09923970n)

1430 数据结构 Abstract /

data_structure(04424844n)

1431 标签 IntentionalProcess / ContentBearingObject

label(00697348v)

label,gummed_label,sticker(05427590n)

1432 记号 ContentBearingObject / ContentBearingObject

notation,notational_system(05098604n)

notation,notational_system(05098604n)

1433 断路 EngineeringComponent / ListOrderFn

open_circuit(03057400n)

opening(00957036a)

1434 容器 Device / Device

container(02490563n)

container(02490563n)

1435 阴影 RadiatingLight / RadiatingLight

shadow(10074475n)

shadow(10074475n)

1436 功 IntentionalProcess / IntentionalProcess

work(00377835n)

work,piece_of_work(03629197n)

1437 摩擦 Contest / Touching

clash,friction(10071758n)

friction,rubbing(07800619n)

1438 电线 Artifact / Device

cord(02499699n)

cord,electric_cord(02500398n)

1439 动力学 FieldOfStudy / FieldOfStudy

dynamics,kinetics(04653368n)

dynamics,kinetics(04653368n)

1440 分配器 Position / Position

distributor(07219043n)

yardmaster,trainmaster,dispatcher(07691101n)

1441 乘法 SexualReproduction / MultiplicationFn

generation,multiplication,propagation(00546392n)

multiplication,times(00560144n)

1442 磁鼓 MusicalInstrument / Device

drum,membranophone,tympan(02610182n)

brake_drum,drum(02329671n)

1443 恒定 forall / forall

changeless,constant,invariant,steady,unvarying(02385304s)

changeless,constant,invariant,steady,unvarying(02385304s)

1444 碰撞 Impacting / Impacting

collision,hit(05444467n)

collision,hit(05444467n)

1445 理解 PsychologicalProcess / PsychologicalProcess

understanding,apprehension,discernment,savvy(04473075n) understanding,apprehension,discernment,savvy(04473075n)

1446 闭环 /

1447 时分 /

1448 多路存取 mother / mother

ma,mama,mamma,mom,momma,mommy,mammy,mum,mummy,mater(07385387n)

ma,mama,mamma,mom,momma,mommy,mammy,mum,mummy,mater(07385387n)

1449 累加器 Position / Device

collector,gatherer,accumulator(07165543n)

adding_machine,totalizer,totaliser(02165463n)

1450 无用 SocialInteraction / Putting

ignore,disregard,snub,cut(00717937v)

cast-off(a),discarded,junked,scrap(a),waste(02378027s)

1451 量子 PhysicalQuantity / PhysicalQuantity

measure,quantity,amount,quantum(00020056n)

measure,quantity,amount,quantum(00020056n)

1452 多目标 /

1453 分裂 RadiatingSound / Separating

rending,ripping,splitting(00296015s)

divide,split,split_up,separate,dissever,carve_up(01681144v)

1454 平台 Artifact / Clothing

platform(03142093n)

chopine,chopines,platform,platforms(02436712n)

1455 审查 Investigating / Investigating

audited_account,audit(09660129n)

inspection,review(00564770n)

1456 漆 Covering / BodySubstance

lacquer(01155013v)

lacquer(10712043n)

1457 控制设备 /

1458 雷 RadiatingSound / RadiatingSound

boom,roar,roaring,thunder(05493195n)

thunder(05506014n)

1459 对齐 Stating / SubjectiveAssessmentAttribute

justify,warrant(00607583v)

alignment(03962879n)

1460 说明书 Book / Book

manual(04840046n)

manual(04840046n)

1461 精密 SubjectiveAssessmentAttribute / True

preciseness,precision(03768956n)

accurate,exact,precise(00595724s)

1462 磁化 Radiating / Radiating

magnetization,magnetisation,magnetic_induction(09677690n)

magnetization,magnetisation,magnetic_induction(09677690n)

1463 读数 PsychologicalProcess / True

reading(04475899n)

reading,meter_reading(04480932n)

1464 刷新 IntentionalPsychologicalProcess / Process

review,brush_up,refresh(00471139v)

refresh,freshen(00113149v)

1465 转矩 Process / Process

torsion,torque(07843790n)

torsion,torque(07843790n)

1466 展开 Increasing / Stating

expansion,enlargement(00235235n)

expansion,enlargement(05362177n)

1467 译码 Encoding / Decoding

encode,code,encipher,cipher,cypher,encrypt,inscribe,write_in_code(00672235v)

decode(00672478v)

1468 极性 equal / SignumFn

mutual_opposition,polarity(09982188n)

polarity,sign(09982634n)

1469 切换 Substituting / Process

switch,switching,shift(00128676n)

switch_over,switch,exchange(00094296v)

1470 申请 Requesting / Text

request,bespeak,call_for,quest(00510998v)

application(04888151n)

1471 波动 Motion / Motion

fluctuation(05473610n)

wave,undulation(05473259n)

1472 自组织 Managing / Managing

self-organization,self-organisation(00737725n)

self-organization,self-organisation(00737725n)

1473 双工 part / Device

duplex(02115038s)

duplex(00231794s)

1474 感觉 Perception / Perception

sensation,sense_experience,sense_impression,sense_datum(04413760n)

perception(04410319n)

1475 振铃 RadiatingSound / RadiatingSound

ring,ringing,tintinnabulation(05502523n)

ring,ringing,tintinnabulation(05502523n)

1476 位元 SubjectiveAssessmentAttribute / Bit

spot,bit(09909906n)

bit(09795462n)

1477 通量 TimeDependentQuantity / TimeDependentQuantity

flux(10975486n)

flux(10975486n)

1478 粒子 SubatomicParticle / SubatomicParticle

particle(06674828n)

particle(06674828n)

1479 毫 /

1480 干涉 Law / inhibits

intervention,interference(04995117n)

hindrance,interference,interfering(00694910n)

1481 柱面 ShapeAttribute / ShapeAttribute

cylinder(10029497n)

cylinder(10016554n)

1482 衰减器 Artifact / Artifact

pad,pad_of_paper,tablet(10743763n)

pad,pad_of_paper,tablet(10743763n)

1483 激发 EmotionalState / EmotionalState

excitation,excitement(00818927n)

excitation,excitement(00818927n)

1484 战术 FieldOfStudy / FieldOfStudy

tactics(04728084n)

tactics(04728084n)

1485 补偿器 Device / Device

equalizer,equaliser(02644356n)

equalizer,equaliser(02644356n)

1486 位图 Icon / Icon

bitmap,electronic_image(02293498n)

bitmap,electronic_image(02293498n)

1487 基数 PositiveInteger / PositiveInteger

base,radix(09770305n)

base,radix(09770305n)

1488 绝缘体 Substance / Substance

insulator,dielectric,nonconductor(10633743n)

insulator,dielectric,nonconductor(10633743n)

1489 制图 ArtWork / ArtWork

drawing(02598242n)

drawing(02598242n)

1490 开关系统 /

1491 空腔 Hole / Hole

pit,cavity(06729656n)

cavity,enclosed_space(10025075n)

1492 打孔 Impacting / Cutting

punch,plug(00969752v)

punch,perforate(00989536v)

1493 默认 ContestAttribute / Selecting

default(00039334n)

default_option,default(04464325n)

1494 计时器 Device / Device

timer(03507166n)

timer(03507166n)

1495 全球 SubjectiveAssessmentAttribute / SubjectiveAssessmentAttribute

global,planetary,world(a),worldwide(01512420s)

global,planetary,world(a),worldwide(01512420s)

1496 分隔 Separating / Separating

separation(10342852n)

separate(02035198a)

1497 媒介 Communication / instrument

media,mass_media(04734700n)

medium(04730097n)

1498 能源 origin / FunctionQuantity

beginning,origin,root,source(06275359n)

energy(07794689n)

1499 形状 ShapeAttribute / ShapeAttribute

shape,form,configuration,contour(03952527n)

shape,form,configuration,contour(03952527n)

1500 叉 Device / Separating

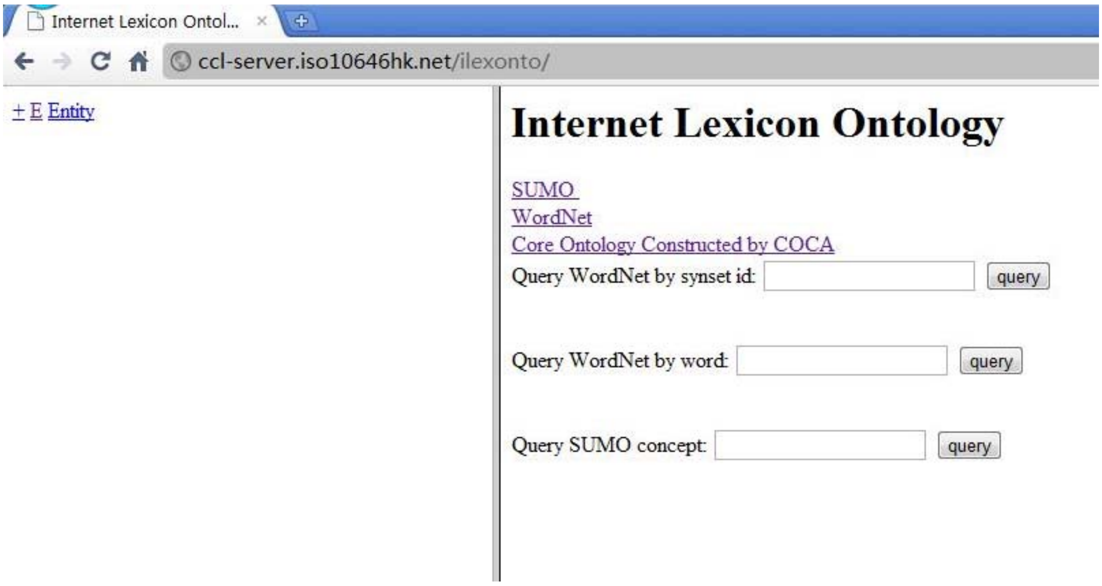
fork(02714767n)

branch,ramify,fork,separate(00228033v)

Appendix 3

Internet Lexicon Ontology with IT Core Ontology by COCA

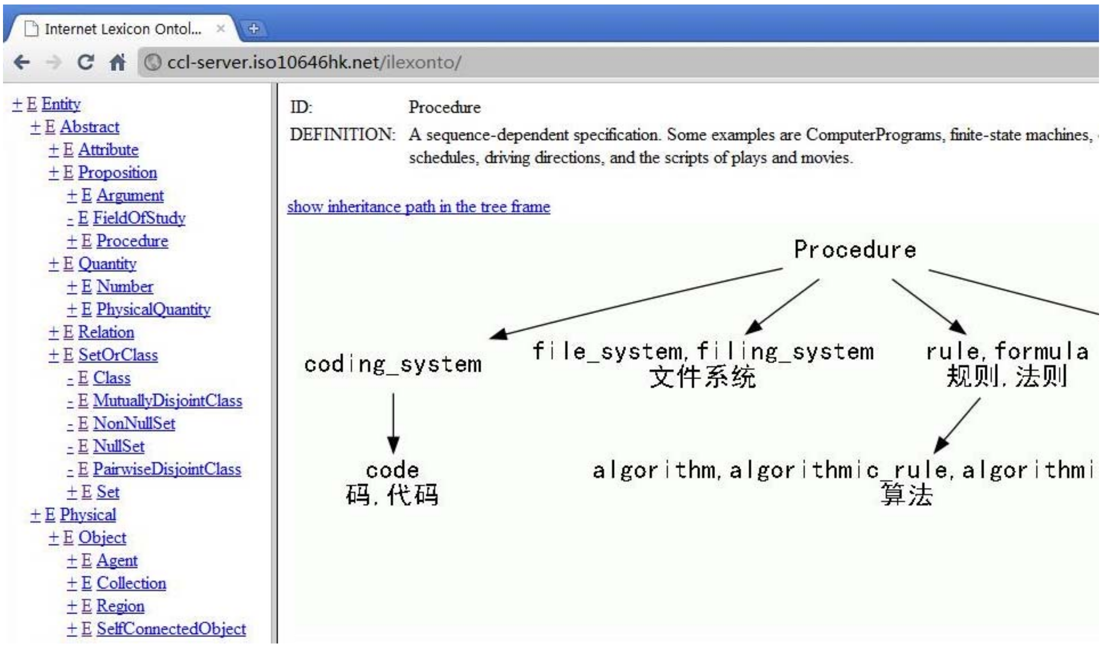
The website "Internet Lexicon Ontology" is a demonstration platform of lexicon ontology and our automatically constructed core ontology in Chinese Computational Linguistics lab of the Hong Kong Polytechnic University. The URL is "http://ccl-server.iso10646hk.net/ilexonto/". The home page of the website is given below.



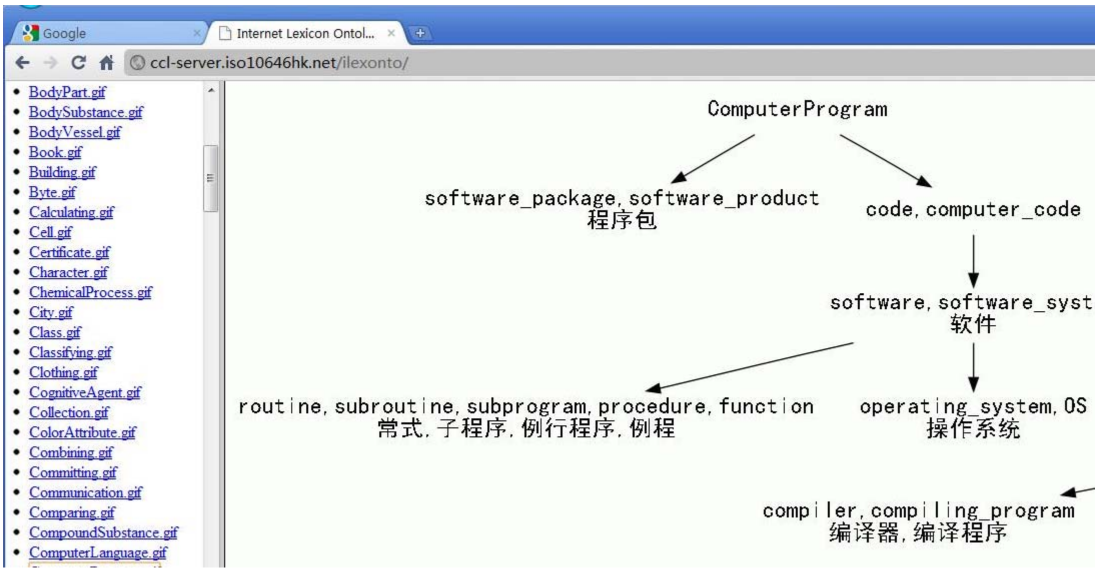
The website mainly demonstrates three lexicon ontology services: SUMO, WordNet and core ontology automatically constructed by COCA.

After clicking the link "SUMO" in the home page, the website shows the whole SUMO hirarchies in left. The link "+" near a concept will expand tis child concepts when it is collapsed. The meaning and corresponding automatically constructed core ontology rooted from the SUMO concept will be shown in the right side by clicking the link "E" besides a

SUMO concept tree node. The example for SUMO concept "Procedure" is demostracted in the following figure.

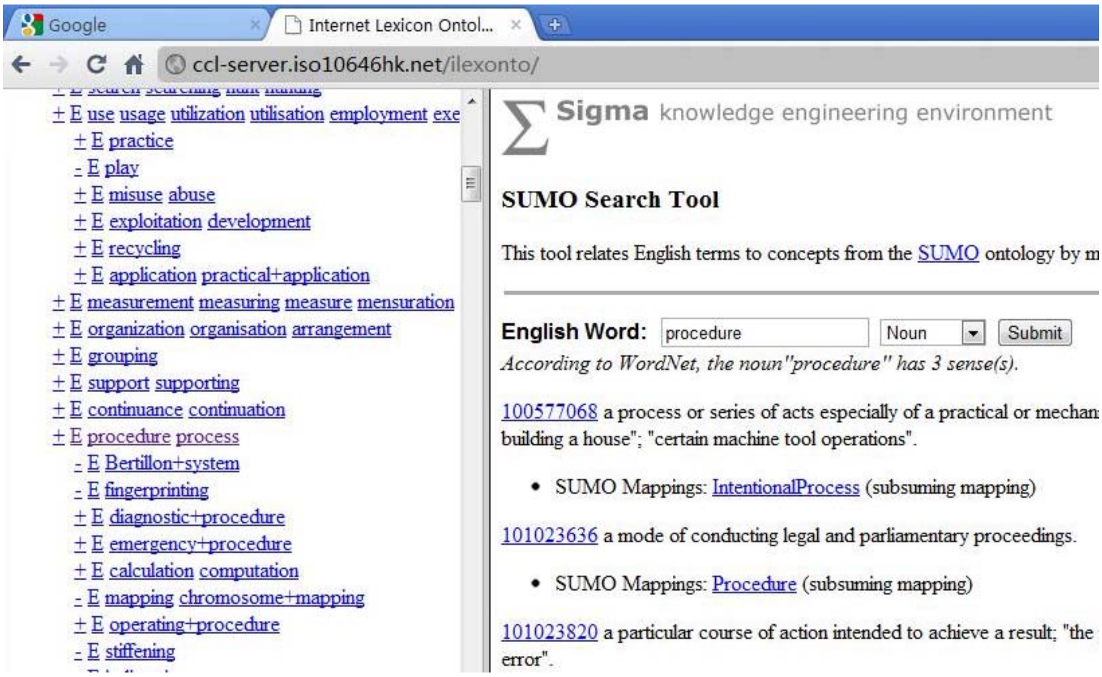


After clicking the link "Core Ontology Constructed by COCA" in the home page, the whole results of core ontology grouped by SUMO concepts will be listed in the left as shown below.



By clicking "WordNet" in the home page, the whole WordNet hirarchies will be shown in the left. After clcking the link "E", the meaning of synset will be shown in the right as below.

Clicking the link of the concept itself in the left will navigate the user to Sigma Knowledge Engineering Environment[7] as shown in the following figure.



In the end, it should be clarified that the Chinene core ontology is automatically constructed without any manual correction and contains some errors to show the practical usability of COCA.

[7] The Sigma knowledge engineering environment is an open source web site for design formal ontologies. (http://sigma.ontologyportal.org:4010/sigma/Browse.jsp?kb=SUMO&lang=EnglishLanguage)