TEKNISKA HÖGSKOLAN

HÖGSKOLAN I JÖNKÖPING

One General Approach For Analysing Compositional Structure Of Terms In Biomedical Field

Yang Chao

Peng Zhang

MASTER THESIS 2012

INFORMATICS

TEKNISKA HÖGSKOLAN
HÖGSKOLAN I JÖNKÖPING

One General Approach For Analysing Compositional Structure Of Terms In Biomedical Field

Yang Chao
Peng Zhang

Detta examensarbete är utfört vid Tekniska Högskolan i Jönköping inom ämnesområdet informatik. Arbetet är ett led i masterutbildningen med inriktning informationsteknik och management. Författarna svarar själva för framförda åsikter, slutsatser och resultat.

Handledare: He Tan
Examinator: Vladimir Trasov
Omfattning: 30 hp (D-nivå)

Datum: 24 March 2013

Arkiveringsnummer:

Abstract

The root is the primary lexical unit of Ontological terms, which carries the most significant aspects of semantic content and cannot be reduced into small constituents. It is the key of ontological term structure. After the identification of root, we can easily get the meaning of terms. According to the meaning, it's helpful to identify the other parts of terms, such as the relation, definition and so on. We have generated a general classification model to identify the roots of terms in this master thesis. There are four features defined in our classification model: the Token, the POS, the Length and the Position. Implementation is followed using Java and algorithm is followed using Naïve Bayes. We implemented and evaluated the classification model using Gene Ontology (GO). The evaluation results showed that our framework and model were effective.

Acknowledgements

We would like to thank our supervisor, Dr. He Tan who supported perfectly during this master thesis work with her wise advices and guidance and also our examiner, Dr. Vladimir Tarasov for his useful advices and discussions on our thesis work.

Yang Chao
Peng Zhang

Key words

Key words: Gene ontology, Machine learning, Classification, Bayes Algorithm, Probability

Contents

List of Figures

List of Abbreviations

ML: Machine Learning

TM: Text Mining

SRL: Semantic Role Labeling

POS: Part Of Speech

SGD: Saccharomyces Genome Database

MGI: Mouse Genome Informatics

SML: Supervised Machine Learning

NN: Noun

JJ: Adjective

NNS: Plural

VB: Verb

RB: Adverb

IN: Preposition

1 Introduction

Ontology links concept labels to their interpretations, specifications of their meanings including concepts definitions and relations to other concepts. Apart from relations such as is-a, generally presented in almost any domain, ontologies also model domain-specific relations, e.g. has-location specific for the biomedical domain. Therefore, ontologies reflect the structure of the domain and constrain the potential interpretations of terms [1].

A term is defined as a textual realization of a specialized concept. Text definitions precisely state the exact meaning of a term. Not all terms have text definitions. Text definitions are interpreted by users, not computers. It has been noted that many term names are compositional, and indicate implicit relationships to terms. The existence of composite terms leads to redundancy in both text definitions and relationships [2]. The redundancy means that the compositional nature of many terms leads to an increase in the number of relationships and consequent increase in complexity of the ontology. Effective term parsing will reduce the redundancy in biomedical ontologies [3].

The term parsing not only can reduce the redundancy, but also can used for text mining applications such as Semantic Role Labeling (SRL) system.

The sentence-level semantic analysis of text is concerned with the characterization of events, such as determining “who” did “what” to “whom”, “where”, “when” and “how”. It plays a key role in TM applications such as Information Extraction, Question Answering and Document Summarization [4]. The predicate of a clause expresses “what” took place, and other sentence constituents express the participants in the event. SRL is a process that, for each predicate in a sentence, indicates what semantic relations hold among the predicate and its associated sentence constituents [5]. However the development of SRL systems for the biomedical domain is frustrated by the lack of large domain specific corpora that are labeled with semantic roles.

Researchers found that ontological concepts typically are associated with textual realization which is to give precise meaning of concepts within context of a particular ontology in many cases [6]. Intuitively, ontological concepts, relations, rules and their associated textual definitions can be used as the frame-semantic descriptions imposed on a corpus. Our supervisor has proposed one method for building corpus which is labeled by semantic role labeling for the biomedical domain. The method makes use of domain knowledge provided by ontology [7]. By this method, a corpus has been built which is related to transport events. In order to extend the corpus and get the ontological concepts, relations, rules and their associated textual defintions not just based on the transport events, we intend to develop a general method that supports parsing and visualizing lexical properties of ontological terms.

As an important step towards fulfilling the objective, formulating correct research questions has a vital role. We have mentioned the research question as followed in next section.

1.1 Purpose/Objectives

In this master thesis, we will do the first step for analyzing the compositional structure by identifying the roots of terms, because the root is the key of the term structure. After the identification of root, we can easily get the meaning of terms. According to the meaning of terms, it's helpful to identify other parts such as relations, rules and so on. The purpose of this thesis work is formulated in two questions:

1. What is the method for identify the roots of terms?

2. What general theory framework is needed regarding support of the method classification we chosen?

1.2 Background

We proposed a classification model for identify the roots of terms and defined four features: Token, Part of speech (POS), Length, Position. Token means different word in ontology terms. POS means linguistics category of words (such as noun, prep, adjective, etc.). Length means numbers of tokens in one term. Position means the position of tokens in each term. Based on this model, we used Gene Ontology (GO) data as implemented and evaluated.

1.3 Limitations

Our method has no limitation. It is a general classification model we developed. It is not only applied for biomedical field, but also for other filed ontologies such as the business filed and the industry field.

The scope of the project is that implemented in the GO.

1.4 Thesis outline

This document is structure as six chapters:

- Chapter 1 which has covered the outline of thesis is basically introduction part that presents ontology, text, term parsing, semantic role labeling, background and objective work.
- Chapter 2 gives the definitions for main concepts used in the implementation.
- Chapter 3 describes the research method followed for reaching the thesis goals.
- Chapter 4 deals with introducing framework as well as explaining the method used for implementation.
- Chapter 5 presents the results achieved during the thesis work.
- Chapter 6 analyzes the result in different ways.

2 Theoretical Background

This chapter will include the basic knowledge regarding the definition of ontology, especially GO and what is the supervised machine learning. It also includes the definition of classification and Naïve Bayes Algorithm, how they work as well as the basic knowledge of POS. The reader of this research could understand basic knowledge which is related to object of our thesis.

2.1 Ontology

In computer science, ontology formally represents knowledge as a set of concepts with a domain and the relationship between pairs of concepts. It can be used to model a domain and support reasoning about entities [8].

In the implementation, ontology is a detailed description of the conceptualization. The core role is to define the specialized vocabulary of a particular area or field, as well as the relationship between them. The basic concept of this series as the cornerstone of a building project provides a unified understanding for the exchanged parties. In support of these series concepts, knowledge of the search, the efficiency of the accumulation and sharing will be greatly improved, and in the true sense of the knowledge reuse and sharing become possible [9].

Ontology can be divided into four types: Domain ontology, General ontology, Application ontology and Representation ontology. Domain ontology contains the knowledge of the specific type of the field (such as electronics, machinery, medicine and teaching); General ontology is covered in a number of domain, often referred to as the core of the ontology; Application ontology contains all the knowledge required for a specific domain modeling; Representation ontology is not only confined to a specific area, but also provide the entity used to describe things, such as the frame ontology, which defines the framework, the concept of slot [10]. In this master thesis, we have used GO as the implemented data. It is a successful application in biomedical field.

2.1.1 Gene Ontology

In this master thesis, we have chosen GO as the data set in the evaluation because the GO is one of the most biggest and successful applications in biomedical field. It covers different kinds of events in biology field. The events are carefully named in ontologies.

GO is made by GO consortium. GO consortium's intention is to create shared biological information resources, which can be able to allow the industry to take advantage of the public vocabulary and semantic description of gene products.

GO consortium was built in the early 1998. GO project is work alliances in the area of information integration. Two sides of work are 1) providing gene production’s consistency description in different databases, 2) classification and characteristics of standardization sequence [11].

The Gene Ontology project is a major bioinformatics initiative with the aim of standardizing the representation of gene and gene product attributes across species and databases. The project is provides a controlled vocabulary of terms for describing gene product characteristics and gene product annotation data from GO Consortium members, as well as tools to access and process this data.

The GO is a collection of three ontologies, which partitioned into orthogonal domain, contains molecular function, biological process and cellular component. The biological process covers different kinds of events. It operates or sets molecular events with a defined beginning and end, pertinent to the functioning of integrated living units: cells, tissues, organs and organisms. A biological process is series of events accomplished by one or more ordered assemblies of molecular functions. A biological process is not equivalent to a pathway; at present, GO does not try to represent the dynamics or dependencies that would be required to fully describe a pathway.

GO consists of terms that are defined as a textual of a specialized concept. Terms may also have one or more synonyms. Terms are interconnected via typed binary relationships, for example, ‘interleukin is a cytokine’ or ‘small ribosomal subunit part of mitochondrial ribosome’ [13]. The existence of composite terms leads to redundancy in both text definitions and relationships in GO [14]. For example, the term ‘cytokine’ has redundant definitions embedded in the text definitions for ‘cytokine metabolism ’and‘cytokine biosynthesis. The redundancy in relationships manifests itself as ‘cytokine metabolism’ and ‘cytokine biosynthesis’ being related via “is-a” to both ‘protein metabolism’ and ‘protein biosynthesis’. Thus, the effective method for terms parsing is necessary.

For SRL systems in the biomedical domain, it lacks of large domain specific corpora that are labeled with semantic roles. Many biomedical text mining systems have mainly used ontologies as terminologies to recognize biomedical terms, by mapping terms occurring in text to concepts in ontologies, or used ontologies to guide and constrain analysis results, by populating ontologies. Our supervisor believes that ontological, as a structured and semantic representation of domain-specific knowledge, can instruct and ease all the tasks in corpus construction.

GO, one of the biggest ontologies in biomedical field, many concepts in GO that comprehensively describe a certain domain of interest in biomedical field. GO biological process ontology, containing 20,368 concepts, provides the structured knowledge of biological processes that are recognized series of events or molecular functions. The ontological terms can be seen as phrase that exhibit underlying compositional structures [41]. For example, for ‘protein transport’ in GO, if we have known the compositional structures of some direct subclasses describing various types of protein transport, we will easily to know what are the possible predicates evoking the protein transport events. Then we can use the classes and relations to define the semantic frame ‘Protein Transport’, decided the participants involved in the event, and listed the domain-specific words evoking the frame [41]. As such, domain knowledge provided by ontologies, such as GO biological process ontology and molecular function ontology will instruct us in building large frame corpora for the domain. An important step towards building large corpora is analyzing the compositional structure of terms. The GO covers different kinds of events related to Gene products. Hence, we used GO as our implemented data.

2.2 Machine Learning

Machine learning is one of the core research areas of artificial intelligence. The initial motivation is to let a computer system with a person's ability to learn in order to achieve artificial intelligence. It is well known that if the system does not have the ability to learn the system can hardly be considered to have intelligent. It is widely used in the definition of machine learning to "use the experience to improve the performance of the computer system itself" [15, 16].

Machine learning has different algorithm types. They are supervised learning, unsupervised learning, semi-supervised learning reinforcement learning, learning to learn. Supervise learning is the machine learning task of inferring a function from labeled training data. The training data consist of a set of training examples. In this master thesis, we want to generate a general method for analyzing the compositional structure of ontological terms. The task is to identify the root of terms. Our problem is a supervised learning problem. It is to classify each token to “Root” or “NoRoot”.

2.2.1 Classification

In machine learning and statistics, classification is the problem that identifies a new observation belongs to a set of predefined categories. It depends on the basis of a training set of data containing observations whose category membership is known [17]. The individual observations are analyzed into a set of quantifiable properties, known as various explanatory variables and features. An algorithm that implements classification, especially in a concrete implementation, is known as a classifier. The term "classifier" sometimes also refers to the mathematical function, implemented by a classification algorithm that maps input data to a category.

Terminology across fields is quite varied. In statistics, where classification is often done with logistic regression or a similar procedure, the properties of observations are termed explanatory variables and the categories to be predicted are known as outcomes, which are considered to be possible values of the dependent variable. In machine learning, the observations are often known as instances, the explanatory variables are termed features and the possible categories to be predicted are classes [17].

In this master thesis, classification means we may know for certain that there are so many classes, and the aim is to establish a rule whereby we can classify a new observation into one of the existing classes. It is considered as an instance of supervised learning. According to our classification model, when the new tokens arrive, we classify them into different two categories: one category is tokens are marked into root, the other is tokens are not marked into root.

2.2.2 Naïve Bayes

In this master thesis we implemented Naïve Bayes algorithms for classification.

A naive Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong independence assumptions. In probability theory, to say that two events are independent means that the occurrence of one does not affect the probability of the other. Similarly, two random variables are independent if the observed value of one does not affect the probability distribution of the other. An advantage of the naive Bayes classifier is that it only requires a small amount of training data to estimate the parameters necessary for classification [18].

Depending on the precise nature of the probability model, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for Naive Bayes model uses the method of maximum likelihood; in other words, one can work with the naive Bayes model without believing in Bayesian probability or using any Bayesian methods [19].

The Naïve Bayes algorithm is a classification algorithm based on Bayes rule, that assumes the attributes $X_1 \ldots X_n$ are all conditionally independent of one another, given Y. The value of this assumption is that it dramatically simplifies the representation of P (X|Y), and the problem of estimating it from the training data. P (X|Y) means the possibility that event X occurs with the condition B.

Consider, for example, the case where $X=X_1$ or $X=X_2$. In this case, $P(X|Y)=P(X_1|X_2,Y)P(X_2|Y)=P(X_1|Y)P(X_2|Y)$. It follows above definition of conditional independence. More generally, when X contains n attributes which are conditionally independent of one another given Y, we have

$$P(X_1 ... X_n | Y) = \prod_{i=1}^{n} P(X_i \mid Y)$$

Equation 2-1: The equation for conditional assumption [19]

Now we derive the Naïve Bayes algorithm, assuming in general that Y is any discrete-valued variable, and the attributes $X_1 \ldots X_n$ are any discrete or real-valued attributes. Our goal is to train a classifier that will output the probability distribution over possible values of Y, for each new instance X that we ask it to classify. The expression for the probability that Y will take on its possible value, according to Bayes rule, is

$$P(\mathrm{Y}=\mathrm{y_k}|\mathrm{X_1 ... X_n}) = \frac{P(\mathrm{Y}=\mathrm{y_k})\prod_{\mathrm{i=1}}^{\mathrm{n}} P(\mathrm{X_i} \mid \mathrm{Y}=\mathrm{y_k})}{\sum_{\mathrm{j}} P(\mathrm{Y}=\mathrm{y_j})\prod_{\mathrm{i=1}}^{\mathrm{n}} P(\mathrm{X_i} \mid \mathrm{Y}=\mathrm{y_j})}$$

Equation 2-2: The fundamental equation for the Naïve Bayes classifier [19]

As the equation 2-2 shown, given a new instance $X_{new}=(X_1…X_n)$, this equation shows how to calculate the probability that Y will take on any given value, given the observed attribute values of X_{new} and given the distributions $P(Y)$ and $P(X_i|Y)$ estimated from the training data. If we are interested only in the most probable value of Y, then we have the Naive Bayes classification rule:

$$\mathrm{Y} \leftarrow \arg\max P(\mathrm{Y}=\mathrm{y_k})\prod_{\mathrm{i=1}}^{\mathrm{n}} P(\mathrm{X_i} \mid \mathrm{Y}=\mathrm{y_k})$$

Equation 2-3: The formula for target value Y [19]

The general classifier function for the probability model is defined as follows:

$$\mathrm{Classify}(\mathrm{f_1......f_n}) = \arg\max P(\mathrm{C}=\mathrm{c})\prod_{\mathrm{i=1}}^{\mathrm{n}} P(\mathrm{F_i}=\mathrm{f_i}|\mathrm{C}=\mathrm{c})\, P(\mathrm{F_i}=\mathrm{f_i}|\mathrm{C}=\mathrm{c})$$

Equation 2-4: The formula of Naïve Bayes probability model [18]

As the equation 2-4 shown, the dependent class variable C means small numbers of classes, F means several feature variables. The naive Bayes classifier combines this model with a decision rule. One common rule is to pick the hypothesis that is most probable; this is known as the maximum a posteriori decision rule. According to this equation, we should calculate the different probabilities according to the feature variables, and then the max probability is most probable. Finally according to these probabilities, we classify into different categories.

The $P(F_i=f_i|C=c)$ means the proportion when the feature $F=f_i$ is in the classes C is equal as c.

There is one normal problem in Naïve Bayes Classification. If a given class and feature value never occurs together in the training data, then the frequency-based probability estimate will be zero. This is problematic because it will wipe out all information in the other probabilities when they are multiplied. Therefore, it is often desirable to incorporate a small-sample correction, called pseudocount, in all probability estimates such that no probability is ever set to be exactly zero.

There are some methods for solving this problem, i.e. Bayesian estimation, Laplace smoothing. We used the Laplace smoothing in this master thesis. Laplace smoothing is a technique used to smooth categorical data. For example, given an observation x = (x_1… x_d) from a multinomial distribution with N trials and parameter vector θ = (θ_1… θ_d), a "smoothed" version of the data gives the estimator:

$$\hat{\theta}_i = \frac{x_i + \alpha}{N + \alpha d} \qquad (i = 1, \ldots, d)$$

Equation 2-5: The laplace smoothing

As the equation 2-5 shown, where $\alpha > 0$ is the smoothing parameter ($\alpha = 0$ corresponds to no smoothing). Additive smoothing is a type of shrinkage estimator, as the resulting estimate will be between the empirical estimate x_i/n, and the uniform probability $1/d$. Using Laplace's rule of succession, some authors have argued that α should be 1, though in practice a smaller value is typically chosen.

2.2.3 Features of the Classification Model

In machine learning and pattern recognition, a feature is an individual measurable heuristic property of a phenomenon being observed. Choosing discriminating and independent feature is the key to any pattern recognition algorithm being successful in classification [42].

There are four features in our classification model. These fearures are: the Token, the POS, the Length and the Position. The token means a word or other atomic parse element. The POS means the part of speech. The Length means different lengths of terms. The position means the token’s position in each term.

2.2.3.1 Part-Of-Speech

Part of speech (POS) is one of the features we choose in our implementation.

In grammar, a part of speech (also a word class, a lexical class, or a lexical category) is a linguistic category of words (or more precisely lexical items), which is generally defined by the syntactic or morphological behavior of the lexical item in question. Common linguistic categories include noun and verb, among others. There are open word classes, which constantly acquire new members, and closed word classes, which acquire new members infrequently if at all.

In corpus linguistics, part-of-speech tagging (POS tagging or POST), also called grammatical tagging or word-category disambiguation, is the process of marking up a word in a text as corresponding to a particular part of speech, based on both its definition, as well as its context—i.e. relationship with adjacent and related words in a phrase, sentence, or paragraph. A simplified form of this is commonly taught to school-age children, in the identification of words as nouns, verbs, adjectives, adverbs, etc.

POS tagging is to give each word in the sentence to the correct lexical token. It is one of the natural language processing and based on research topics, it is also the basis of other intelligent information processing technology; it has been widely used in machine translation, text recognition, speech recognition, and information retrieval [20]. POS tagging is a very useful pretreatment process for subsequent natural language processing, the degree of accuracy will directly affect the effect of the subsequent series of analytical processing task.

So far, POS tagging task has used a variety of technical methods, including rule-based and statistical-based method.

The rules method can accurately describe the phenomenon between POS. The Brill tagger is a rule-based method for doing POS. It was described by Eric Brill in his 1995 PHD thesis. It can be summarized as an "error-driven transformation-based tagger". It is error-driven in the sense that it recourses to supervised learning. It is transformation-based in the sense that a tag is assigned to each word and changed using a set of predefined rules. Applying over and over these rules, changing the incorrect tags, a quite high accuracy is achieved [21].

In this master thesis, we developed an improved POS rule-based method based on Brill tagger. There are two mainly steps for our POS method:

- First step: we used Oxford dictionary that contains about 90000 English words. In this dictionary, each word has different potential tagging lists. Then we assigned a potential POS tagging to each token according to dictionary.

- Second step: we used a number of hand-written rules to adjust the POS for each word based on the result of first step. Then we changed the potential tagging into the final tagging according to the rules. These rules as function format have implemented in our program. Each rule is independent.

The table 2-6 shows the POS tagging table we used.

POS	The definition	Example
NN	Noun	Protein
JJ	Adjective	Yellow
NNS	Plural	Llamas
VB	Verb	Eat
IN	Preposition	In
RB	Adverb	Never

Table 2-6: The POS tagging

3 Research Methods

The goal of the research process is to produce new knowledge or deepen understanding of a topic or issue. Hence, in order to create a new knowledge, or make the knowledge about a subject deeper, choosing the appropriate research method is necessary [37]. There are two main ways for research design: qualitative research and quantitative research [22]. One of these research design types are chosen by researches depending on the research problem which is going to be observed or the research question which is going to be answered [23]. Different steps of conducting this master thesis as a research are explained in this part.

In this master thesis we used design science methodology to present a design science research. Besides, we got much benefit of literature search for theoretical contribution in this subject and then the implementation method is used Java language in structured programming [34, 35].

Design science is a methodology in scientific researches which is mainly used in researches in the field of information systems (IS). This kind of researches focuses on development and performance of artifacts with a clear intention of improving the performance of artifacts [36].

Design process is consisting of some different phases which are defined by various DSR frameworks. These divisions of phases are done by specifying a set of milestones during the design process. These design science research method (DSR) frameworks usually launch repetitive approach including several phases of the design process. The example of this type can be referring to [24, 25].

These steps of DSR are illustrated as following [26]:

1. Awareness of the problem

2. Suggestion

3. Development

4. Evaluation

5. Conclusion

We have followed these steps for using the methodology in our master thesis. Below we have described how we have translated the five steps in the process of our work. Selection 3.1 to 3.5 explains how the structure has been fitted to the steps of design science methodology.

3.1 Awareness of the Problem: Literature Search

It is the first step in starting a research. By reading different literatures related to the subject we have chosen, the awareness of the problem was obtained for our research work. We started the research question for this matter. There is a research question which has been introduced in the beginning of a research and has been answered during that research [28]. Result of the literature review can formulate the problem and become a motivation to the research work. Using a relevant theory is helpful for applying some parts of the theory into the proposed theory. This needs reviewing the past literature.

Regarding our research questions and also implementing the proposed method, we reviewed relevant books and articles which were carefully selected by recently cited sources and authors.

Taking into consideration the role of literature review which is to develop theoretical framework and also conceptual models, the act of combining relevant elements from earlier studies is helpful [38]. Regarding this matter, we have motivated our work by paying attention to the researches done in this field.

In order to getting benefits of relevant researches, various literature which fit into our research area have been studied to get theoretical background knowledge such as GO, classification, Naïve Bayes, machine learning, as well as information regarding similar experiments. This information has been sought both from books on the topic and scientific reports and journals. Books have been found by searching the local and national library catalogues Higgins and Libras.

Finally by reviewing different of systems in the biomedical field, we developed the framework of our system which is illustrated in Figure 4-1 and later described in chapter 4.

3.2 Suggestion

Next step in providing a research is "suggestion" phase. This phase is necessarily used after recognizing the problem of research's field and can be applicable after making a proposal as a output of the problem recognition [29]. Suggestions are the approaches including methods and methodologies which help the proposal to solve the mentioned problem.

By having suggestion's definition in mind, the task of choosing an appropriate method reached us to the suggested following steps:

1) Developed the classification model

   We proposed a classification model that identifies the roots of terms. There are two categories in our classification model: one category is tokens are marked into root, the other is tokens are not marked into root.

2) Chose the appropriate algorithm

   We chose the Naïve Bayes algorithm and it is described in chapter 2.2.2.

3) Selected the features

   With the help of our supervisor, we defined the four features as described in chapter 2.2.3. We have supported one new method for extracting the feature POS.

4) Selected the implementation and evaluation data

   We chose the GO as our implementation and evaluation data

After these suggestions, we decided to use Java as our programming language and My eclipse 9.0 as our programming environment. In the suggestion phase we have described about development of system and the methodology we chose in next sections which are mainly mentioned in 3.3 and 4.2. According to this step, we recognized the need of requirements within system's developing. Developing the suggestion's step in the next phase which we have explained below [28, 39].

3.3 Development

This phase focuses on the development and implementation of the tentative design which was described previously in suggestion phase. Creative efforts are needed while moving from tentative design into complete design requires. Developing and implementing approaches are different due to the differences in making artifacts, sometimes an algorithm is needed in order to build the development technique [30, 31].

In this master thesis, we developed a program for identifying the roots of terms according to our classification model. Our development environment was My eclipse 9.0 and programming language was Java.

In this master thesis, the main implementation is to implement the classification model for identifies the roots of terms. We need not develop a complete system like big software engineering project and only develop some specific programming for each parts implemented. With the help of our supervisor, we developed our implementation part according to the surprised machine learning process. This will be described in chapter 4 later.

3.4 Evaluation: Cross-validation

Evaluation is considered as an activity in software engineering to determine the quality of the proposed software. After developing the proposal, the output should be evaluated. The attention of evaluation phase is judging results according to performance and measurement of algorithm [40].

By searching various of papers about the evaluation method, we decided to use the cross-validation method. Cross-validation is a technique for assessing how the results of a statistical analysis will generalize an independent data set. It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice [32].

Cross-validation is widely used for evaluating training dataset and test dataset in machine learning. All the examples in the dataset contain training dataset and test dataset are eventually for both training and testing. The evaluation result has large persuasive.

Section 4.2.4 will explain more about the cross-validation used in our experiment.

3.5 Conclusion Phase

The conclusion phase is the last step in creating a design science research. The results are focused to the classification model for identifying the roots of terms. The main involvement of the conclusion is to achieve results, which are defined clearly in the purpose or objective of the proposal. We conclude after the evaluation phase from the domain experts and knowledge mentors, that the results are authentic and that they are truly mapped according to the purpose of this thesis.

The analysis of results, taken from evaluation leads us to have an overall understanding of our system. It can be conclude that how accurate is the system according to analyze the results.

4 Framework and practical methods

We presented the general framework that contains the whole process of supervised machine learning. Within this framework, we developed our system and implemented in GO.

4.1 Framework of supervised machine learning method

The overview of the supervised ML is illustrated in figure 4-1.

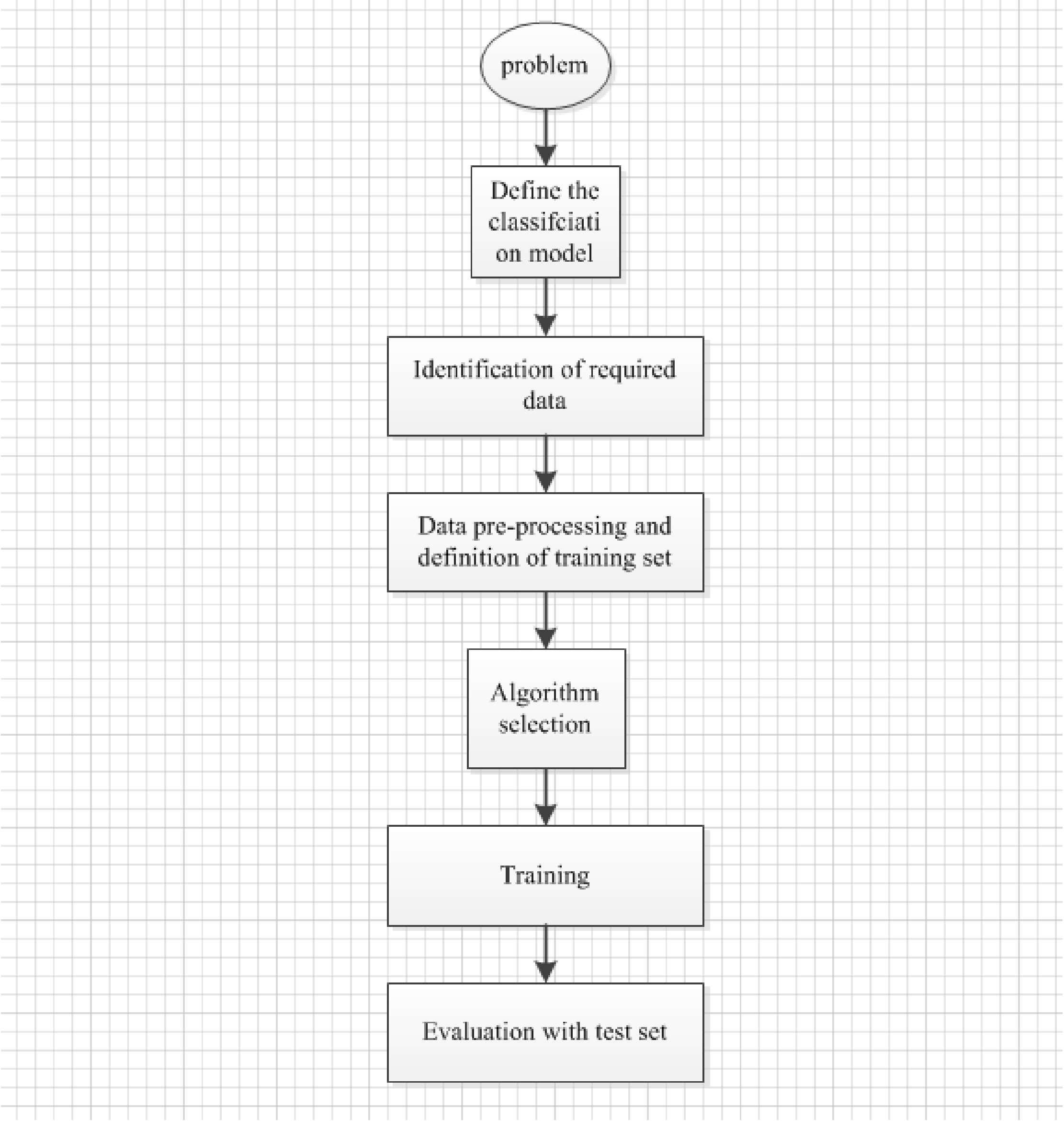

problem
Define the classifciati on model
Identification of required data
Data pre-processing and definition of training set
Algorithm selection
Training
Evaluation with test set

Figure 4-1: The process of supervised ML

Regarding the literature, we found the framework of supervised ML method [3]. As the figure 4-1 shows, there are mainly four steps in the process of supervised ML in our thesis project. The first step is collecting the dataset, the second step includes the data preparation and data pre-processing, the third step includes algorithm selection and training, the last step is evaluation with test set.

- **Problem**

  We first defined our problem that it is to identify the roots of terms by using the classification model.

- **Define the classification model**

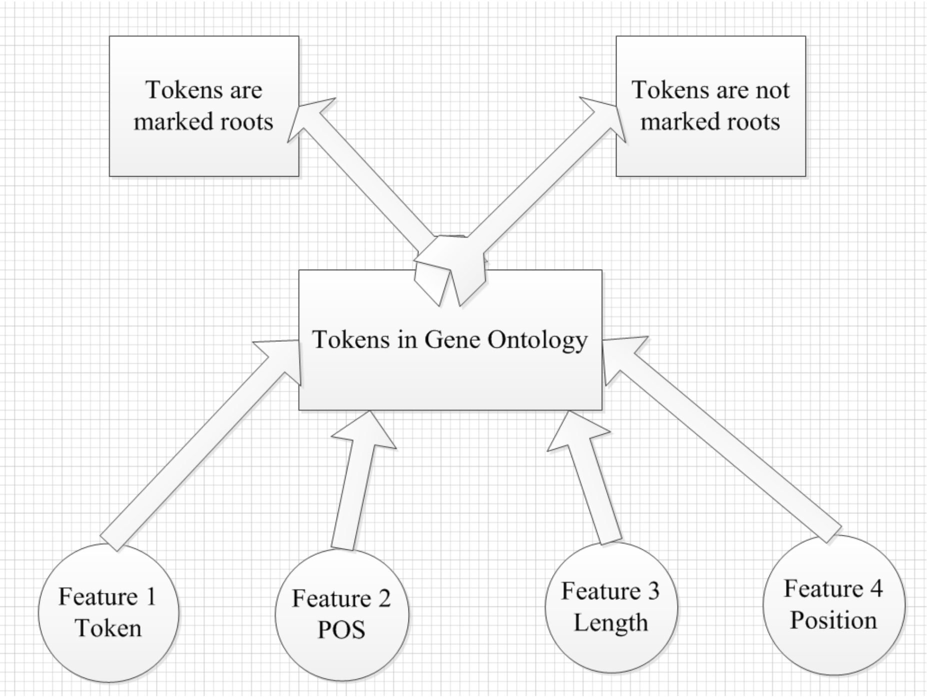


Figure 4-2: The classification model

Regarding the literature and the help of our supervisor, we proposed the classification model as the figure 4-2 shown [43]. In figure 4-2, in our classification model, we defined four features: token, POS, length, position. The token means a word or other pares in lexical analysis. For example, for the term 'transport protein', the tokens are 'transport' and 'protein'. The POS means POS of each token in its term. The Length means the length of a term. It is the number of tokens in a term. The position means a token's position in its term. We defined two categories in our classification model. One category is that a token is recognized as root; the other category is that a token is recognized as not root. According to these four features, we used the Naïve Bayes algorithm to calculate the probability for each token as roots of terms. Compare these probabilities of tokens as roots and the biggest one token will be the root of term.

- **Identification of required data**

  We chose the GO as our required data, because the GO was the biggest applications in the biomedical field. It covers different kinds of events in biomedical field.

  Data pre-processing and definition of training set

  After we decided to choose the GO as our required data, we did some pre-processing. Data pre-processing is an important step in the data mining process. The phrase "garbage in, garbage out" is particular applicable to data mining and machine learning projects. Data-gathering methods are often loosely controlled, resulting in out-of-range values, impossible data combinations, missing values, etc. Analysing data that has not been carefully screened for such problems can produce misleading results. Thus, the representation and quality of data is foremost before running an analysis.

  According to the classification model, we defined the training dataset.

- **Algorithm selection**

  Regarding reading various of literatures, we chose the Naïve Bayes Algorithm for this thesis project.

- **Training and evaluation test set**

After defined the training data set and selected the algorithm we would train our test dataset and do the evaluation with test data set. The training process is a classifier process.

In the selections that follow, we will describe step “identification of required data” to step “training and evaluation test set” in detail.

4.2 The development of system

4.2.1 Identification of required data and collecting the dataset

The first step is collecting the dataset. There are two normal cases for collecting the dataset. If an expert is available, then he or she could suggest which field is the most important. If not, then the simplest method is that of “brute-force”, which means measuring everything available in the hope that the informative and relevant features can be isolated [33].

In this master thesis, we combined these two cases for collecting dataset. With the help of our supervisor, she suggested the GO was the most informative ontology in biomedical field. In our literature search, we also found that the GO is the biggest application in text processing in biomedical field. Then we used the “brute-force” method defined the whole GO terms as our original dataset. Finally, we collected all 32464 terms in the GO. The GO version we used is GO-1.2. In this step, our input is GO-1.2 version txt file and output is all the terms of GO.

In this process, we did not consider the synonyms in GO. We only consider main terms.

4.2.2 Data preparation and data processing

4.2.2.1 Select and extract the dataset

We carefully chose the dataset consider the length of terms.

Length	Numbers of terms	Length	Numbers of terms
1	512	16	110
2	4662	17	33
3	11673	18	25
4	6526	19	10

5	5008	20	13
6	3469	21	3
7	2044	22	7
8	1390	23	1
9	966	24	2
10	636	25	4
11	464	26	2
12	328	27	1
13	234	28	1
14	196	29	1
15	133		

Table 4-3: The number of different lengths of GO terms

As the table 4-3 shown, there are many different lengths of GO terms. The length 3 terms has the most and there are few terms more than length 20. According to the table 4-3 shown, in order to cover the different lengths of terms in suitable dataset, we extracted the terms as array list order in the following proportions: the proportion of the terms length=1 and the numbers of the terms<=100 is 60% and the others are 10%. Because as the table 4-3 shown, the length=1 terms and the numbers of terms <=100 is short compared to other lengths terms. Our dataset should cover different lengths of terms. Finally we got the 4050 terms as our data set. In this step, the input is all the GO terms and the output is dataset with 4050 terms.

In dataset we chose, it covers different lengths of terms. The tokens of these terms have different POS, the same tokens have different positions in each term. It covers all the features we selected.

4.2.2.2 Extract the training set

For this step, our task is to get the training set. In machine learning theory, the training dataset should present representative features of dataset. According to the table 4-2, we extracted the training data set from the dataset we defined in 4.2.2.1 in the following proportion: the proportion is 100% that the length of terms is bigger than 18, the proportion is 10% that the length of terms is smaller than 18 . Thus our training set will cover all the different lengths of terms. Then we do the artificial classification for training set.

4.2.2.3 Extract the features of the dataset

For tokenization, we used the space symbol method for dividing the terms and did the further processing under handling some special symbols, i.e. “（”, “）” and “,” We did not handle the other special symbols.

For POS, we proposed an improved method based on the Brill Tagger. Brill Tagger is one POS method has high accuracy [21]. Brill rules are the general form: tag 1→tag 2 IF Condition, where the ‘*condition’* tests the preceding and following word tokens, or their tags. For example, in Brill’s notation:

IN NN WDPREVTAG DT while

It would change the tag of word from IN to NN, if the preceding word’s tag is DT (determiner) and the word itself is “while”. Based on the existing rules in Brill Tagger, we developed some new rules based on the English grammars and some new rules based on the specification biomedical data. We also done some improvement for unknown-word algorithm based on Brill Tagger. We added more variables in program and used for while to improve the effecnicy of existing unknown-alforihm in Brill Tagger.

There are some new rules we added:

1. NN VB PREVTAG TO: it means when the TO is in pre-position of NN, then the NN should be changed to VB.

2. NN s fhassuf 1 NNS x means the unknown word was tagged NN, but the word end with “s”, it should change to NNS.

3. VB the fgoodright NN x means the unknown word was tagged VB, if this word is in the right of “The”, it should change to NN.

The tokens’ format in dictionary we used is: Token tag 1, tag 2, tag 3... tag n

Tag 1, tag 2…, tag n mean different tagging for each token possible. The tag 1 means the most possible tagging for this token. When the new token arrives, the program will look for the dictionary first. If the token is existed in the dictionary, this token will be marked as tag 1. If the token is not in the dictionary, then will use the unknown-word algorithm based on the rules we defined.

In unknown-word algorithm, first we used the information about word spelling for initialization. For example, the words ending in the letters “s” is most likely a plural noun and the words starting with an uppercase letter word is most likely noun and so on. After initialization and then do the POS tagging according to the rules we defined.

For example, for term “Activation Of MARKKK During Sporulation Sensu Saccharomyces”, we got a protein POS tagging for each token according to dictionary, the protein of POS tagging is:

Activation: NN Of: IN MAPKKK: NN During: IN Sporulation: NN Sensu: NN Saccharomyces: NN.

Then we adjusted these results using the rules. For token saccharomyces, it is marked NN, but according to the rules NN s fhassuf 1 NNS x, it should be changed to NNS. Finally the term of POS tagging is:

Activation: NN Of: IN MAPKKK: NN During: IN Sporulation: NN Sensu: NN

Saccharomyces: NNS.

We tested our improved POS method for data set and got 89.1% average accuracy. The accuracy means the proportion of the words were marked POS rightly in the whole words were marked POS.

For length, we used the “I” variable to control and count the numbers of tokens. The length is the numbers of tokens for each term.

For position, we used the array to present the different positions of each token in terms. We got all the features of the dataset. It shown table 4-4 in below:

Term	Token	POS	Length	Position	Root
Reprodcation	Reproduction	NN	1	1	t

Thioredoxin	Thioredoxin	NN	1	1	t
…	…	…	…	…	…
Mitochondrion Inhertance	Mitochondrion	NN	2	1	f
Mitochondrion Inhertance	Inhertance	NN	2	2	t
…	…	…	…	…	…

Table 4-4: The features of the training set

For example, the term “Reproduction”, the POS NN means reproduction is noun, the length ‘1’ means the term has only one token and the position ‘1’ means the token ‘Reproducation’ is in the first position. In this step, our input is the dataset and the output is the dataset with features.

4.2.3 Algorithm selection and training

The choice of which specific learning algorithm we should use is a critical step. Once preliminary testing is judged to be satisfactory, the classifier (mapping from unlabeled instances to classes) is available for routine use. The classifier’s evaluation is most often based on prediction accuracy. In this master thesis, we chose the Naïve Bayes algorithm. Based on the literature, we found some advantages of Naïve Bayes algorithm. It has the faster speed for learning with respect to numbers of attributes and the numbers of instances, the faster speed of classification, tolerance to missing values are better [18].

Based on the classification model, our target value is given by the following formula in the equation 4-5 based on the equation 2-4 in chapter 2:

$$T_{root} = \underset{token_j \in \{terms.tokens\}}{\arg\max} P(root = "t")P(features_1 = token_{\mathrm{j}} \mid root = "t")\prod_{\mathrm{i}=2}^{4} P(features_{\mathrm{i}} \mid root = "t")$$

Equation 4-5: The target formula is based on the Naïve Bayes algorithm [19]

There is a description of the formula shown in equation 4-5:

(1) $P(root="t")$, this is the proportion that the tokens features are “*t*” in the whole training set token.

If there are ten terms that the each length is 1,2,3,4,5, then $P(root = "t") = \frac{50}{1*10+2*10+3*10+4*10+5*10} = \frac{50}{150}$

(2) *P*(*token*=A| *root*="*t*"), this is the proportion that the tokens features are "*t*" and token feature=A in the whole tokens are treated as roots in the training set

(3) *P*(*POS*=B| *root*="*t*"), this is the proportion that the tokens features are "*t*" and POS feature=B in the whole tokens are treated as roots in the training set.

(4) *P*(*length*=C| *root*="*t*"), this is the proportion that the tokens features are "*t*" and length feature=C in the whole tokens are treated as roots in the training set.

(5) *P*(*position*=D| *root*="*t*"), this is the proportion that the tokens features are "*t*" and position feature=D in the whole tokens are treated as roots in the training set.

For example, we want to identify the root in the term "endopolyphosphatase activity", according to our formula:

For endopolyphosphatase, the probability of root is:

$$P(root = "t")P(token = "endopolyphosphatase \mid root = "t")P(POS = "NN" \mid root = "t")$$
$$P(length = 2 \mid root = "t")P(positon = 1 \mid root = "t")$$

It is based on the equation 4-5.

For activity, the probability of root is:

$$P(root = "t")P(token = "activity" \mid root = "t")P(POS = "NN" \mid root = "t")$$
$$P(length = 2 \mid root = "t")P(positon = 2 \mid root = "t")$$

There is an example how to calculate the probability for each part. For example, *P*(*token*="*activity*" | *root*="*t*"), it is the proportion that the numbers of activity token are treated as roots in the training data set. For example, the training data we used contains 453 terms. There are 453 roots in the training data set and in case 70 of roots is "activity", then we get

$$P(token = activity \mid root = t) = \frac{70}{453}$$

Finally, we got the two probabilities for token ‘activity’ and ‘endopolyhosphatase’, compare these two probabilities, the token activity’s probability is bigger than endopolyphosphatase’s probability and the root will be ‘activity’. There is a special case for P(token="activity" | root="t"), if the token activity is never occurs in the training dataset, the probability will be zero. Then we used the Laplace smoothing for correction this case. If a given class and feature value never occurs together in the training data, we will all one smoothing parameter ‘a’. The ‘a’ is equals one. That means each given class and feature at least occurs one times in the training data. It will be avoid the case of probabiluty zero. In our classification model, there are 453 training terms. There are 453 roots in the training dataset and in case 0 of roots is ‘transport’, we will assume the ‘transport’ occurs one times, and then calculate the probabilities.

4.2.4 Evaluation with test set

In this step, our task is to evaluate the test data set according to cross-validation method. Cross-validation is a technique for assessing how the results of a statistical analysis will generalize an independent data set. It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice [32].

In our experiment, we used the improve hold-out method. We divide our data set into two parts. One part training data set, the other one is test dataset. Then divide the test dataset into 9 parts and test each part. Then get the 9 set results and calculate the average accuracy. The test dataset contains some synonyms.

In this step, the input is the 9 test set files, the output is 9 test set files with possibility and 9 test files without probability.

4.2.5 Implementation

For this paper, we used the Myeclipse as our development tool and implemented in Java program. There are 4 mainly parts in our implementation. First part is for selecting dataset, second part is for selecting training dataset, third part is for selecting dataset with features, fourth part is for testing data set.

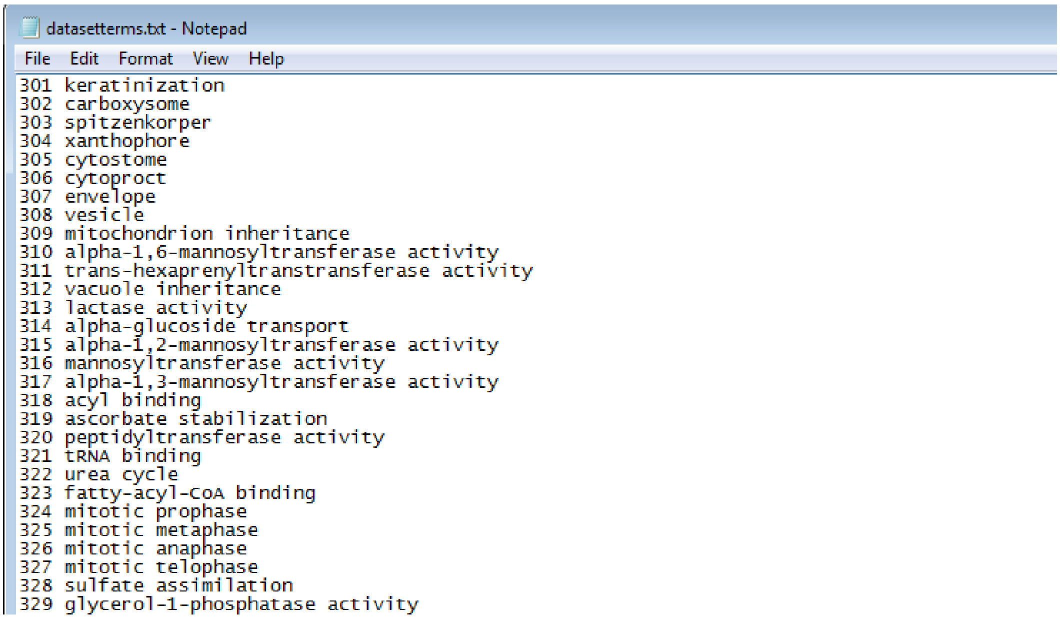
5 Results

This section presents the practical results of our work which we have done during this master thesis. It also states how the objective of this thesis work is reached according to the development process. Once the objective is achieved, the result is ready. The results show how to identify the roots of terms step by step [34].

5.1 Practical Results

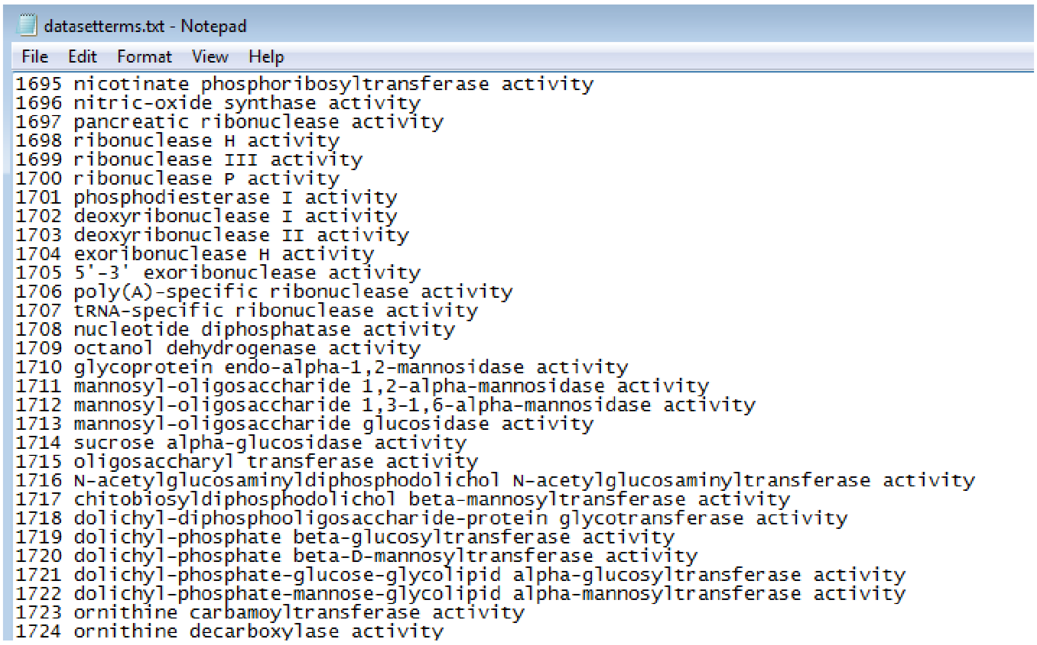
In this selection, we explain the results according to the development of process. Below you can review the results:

- Figure 5-1 and Figure 5-2 showed the dataset collected
- Figure 5-3 and Figure 5-4 showed the dataset with figures
- Figure 5-5 showed the training dataset with classified
- Figure 5-6 showed test dataset 1 with probability
- Figure 5-7 showed test dataset 1 without probability



```
301 keratinization
302 carboxysome
303 spitzenkorper
304 xanthophore
305 cytostome
306 cytoproct
307 envelope
308 vesicle
309 mitochondrion inheritance
310 alpha-1,6-mannosyltransferase activity
311 trans-hexaprenyltranstransferase activity
312 vacuole inheritance
313 lactase activity
314 alpha-glucoside transport
315 alpha-1,2-mannosyltransferase activity
316 mannosyltransferase activity
317 alpha-1,3-mannosyltransferase activity
318 acyl binding
319 ascorbate stabilization
320 peptidyltransferase activity
321 tRNA binding
322 urea cycle
323 fatty-acyl-CoA binding
324 mitotic prophase
325 mitotic metaphase
326 mitotic anaphase
327 mitotic telophase
328 sulfate assimilation
329 glycerol-1-phosphatase activity
```

Figure 5-1: Extract the dataset 1

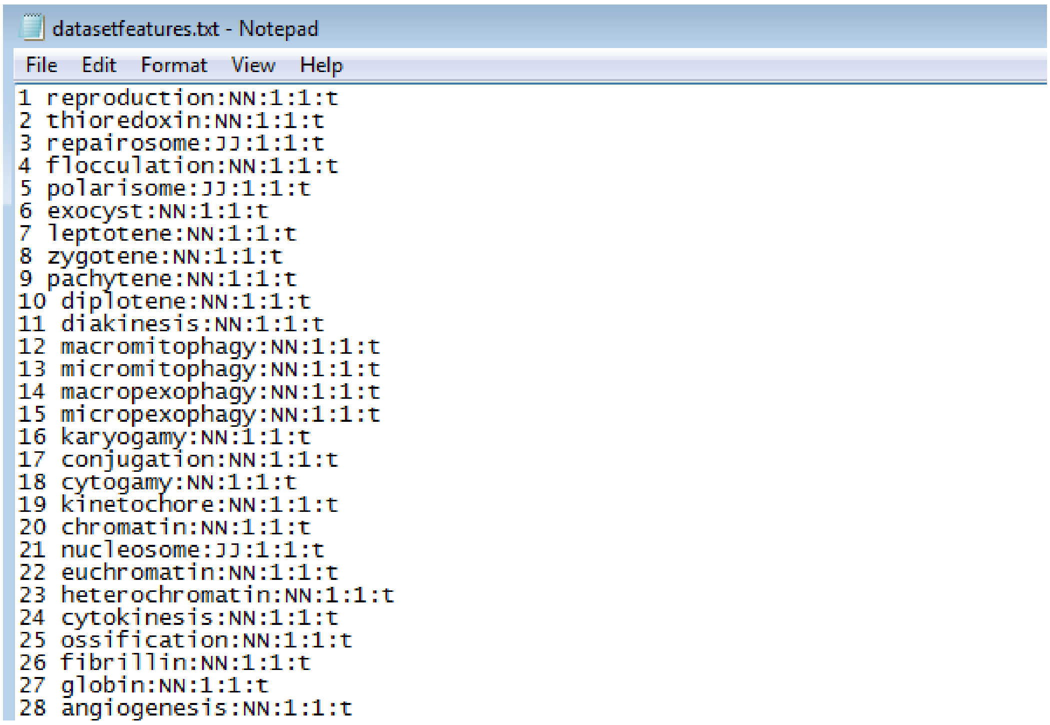


```
1695 nicotinate phosphoribosyltransferase activity
1696 nitric-oxide synthase activity
1697 pancreatic ribonuclease activity
1698 ribonuclease H activity
1699 ribonuclease III activity
1700 ribonuclease P activity
1701 phosphodiesterase I activity
1702 deoxyribonuclease I activity
1703 deoxyribonuclease II activity
1704 exoribonuclease H activity
1705 5'-3' exoribonuclease activity
1706 poly(A)-specific ribonuclease activity
1707 tRNA-specific ribonuclease activity
1708 nucleotide diphosphatase activity
1709 octanol dehydrogenase activity
1710 glycoprotein endo-alpha-1,2-mannosidase activity
1711 mannosyl-oligosaccharide 1,2-alpha-mannosidase activity
1712 mannosyl-oligosaccharide 1,3-1,6-alpha-mannosidase activity
1713 mannosyl-oligosaccharide glucosidase activity
1714 sucrose alpha-glucosidase activity
1715 oligosaccharyl transferase activity
1716 N-acetylglucosaminyldiphosphodolichol N-acetylglucosaminyltransferase activity
1717 chitobiosyldiphosphodolichol beta-mannosyltransferase activity
1718 dolichyl-diphosphooligosaccharide-protein glycotransferase activity
1719 dolichyl-phosphate beta-glucosyltransferase activity
1720 dolichyl-phosphate beta-D-mannosyltransferase activity
1721 dolichyl-phosphate-glucose-glycolipid alpha-glucosyltransferase activity
1722 dolichyl-phosphate-mannose-glycolipid alpha-mannosyltransferase activity
1723 ornithine carbamoyltransferase activity
1724 ornithine decarboxylase activity
```

Figure 5-2: Extract the dataset 2

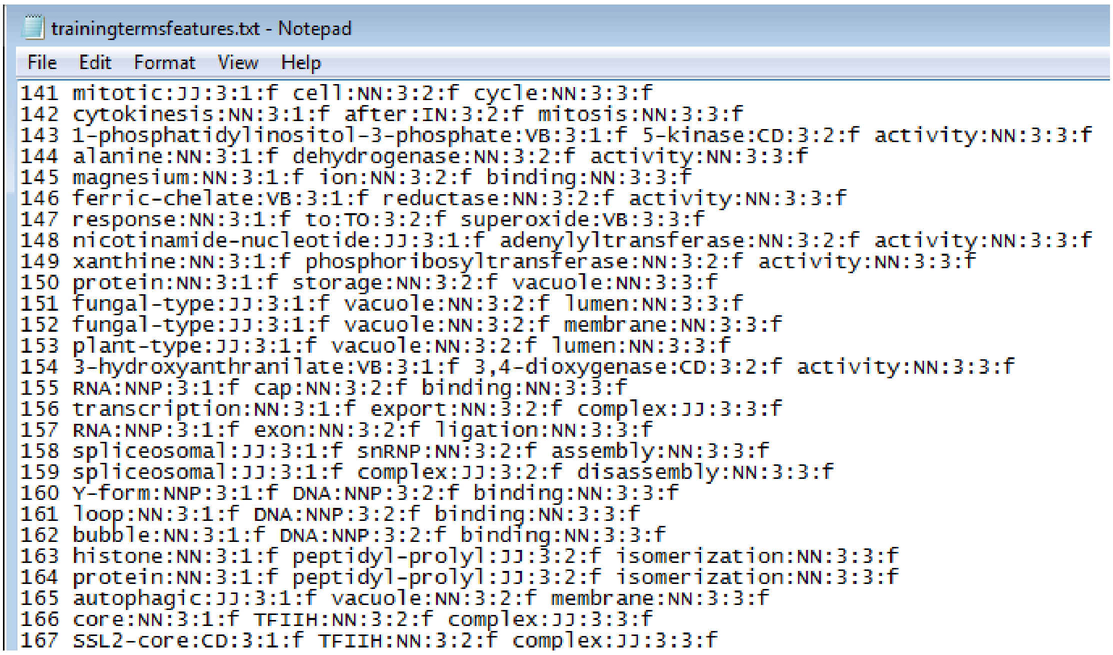
Figure 5-1 and figure 5-2 shows a small subset of data set which contains 4165 GO terms we choose. In machine learning theory, the quality of dataset has the directly effect on the final results. Thus, our dataset should cover all the lengths of terms according to features we choose. As the figure 5-2 shown, we also can know the same token probably have the different positions in different terms. For example, as figure 5-2 shown, for term 1696 ‘nitric-oxide synthase activity’, the token ‘activity’ position is 3 , as figure 5-1 shown, for term 313 ‘lactase activity’, the token ‘activity’ position is 2.

The figure 5-3 shows a small subset of the data set with features. This result is shown we have succeeded extracting all the features we used: token, POS, length, position, and root. For example, as the figure 5-3 shown, for number 1 term ‘reproduction:NN:1:1:T’, NN means reproduction is Noun, the first ‘1’ means the term has one token, the second ‘1’ means the token reproduction’s position is 1. T means ‘reproduction’ was marked root.



```
1 reproduction:NN:1:1:t
2 thioredoxin:NN:1:1:t
3 repairosome:JJ:1:1:t
4 flocculation:NN:1:1:t
5 polarisome:JJ:1:1:t
6 exocyst:NN:1:1:t
7 leptotene:NN:1:1:t
8 zygotene:NN:1:1:t
9 pachytene:NN:1:1:t
10 diplotene:NN:1:1:t
11 diakinesis:NN:1:1:t
12 macromitophagy:NN:1:1:t
13 micromitophagy:NN:1:1:t
14 macropexophagy:NN:1:1:t
15 micropexophagy:NN:1:1:t
16 karyogamy:NN:1:1:t
17 conjugation:NN:1:1:t
18 cytogamy:NN:1:1:t
19 kinetochore:NN:1:1:t
20 chromatin:NN:1:1:t
21 nucleosome:JJ:1:1:t
22 euchromatin:NN:1:1:t
23 heterochromatin:NN:1:1:t
24 cytokinesis:NN:1:1:t
25 ossification:NN:1:1:t
26 fibrillin:NN:1:1:t
27 globin:NN:1:1:t
28 angiogenesis:NN:1:1:t
```

Figure 5-3: Extract the dataset 1 with features



```
141 mitotic:JJ:3:1:f cell:NN:3:2:f cycle:NN:3:3:f
142 cytokinesis:NN:3:1:f after:IN:3:2:f mitosis:NN:3:3:f
143 1-phosphatidylinositol-3-phosphate:VB:3:1:f 5-kinase:CD:3:2:f activity:NN:3:3:f
144 alanine:NN:3:1:f dehydrogenase:NN:3:2:f activity:NN:3:3:f
145 magnesium:NN:3:1:f ion:NN:3:2:f binding:NN:3:3:f
146 ferric-chelate:VB:3:1:f reductase:NN:3:2:f activity:NN:3:3:f
147 response:NN:3:1:f to:TO:3:2:f superoxide:VB:3:3:f
148 nicotinamide-nucleotide:JJ:3:1:f adenylyltransferase:NN:3:2:f activity:NN:3:3:f
149 xanthine:NN:3:1:f phosphoribosyltransferase:NN:3:2:f activity:NN:3:3:f
150 protein:NN:3:1:f storage:NN:3:2:f vacuole:NN:3:3:f
151 fungal-type:JJ:3:1:f vacuole:NN:3:2:f lumen:NN:3:3:f
152 fungal-type:JJ:3:1:f vacuole:NN:3:2:f membrane:NN:3:3:f
153 plant-type:JJ:3:1:f vacuole:NN:3:2:f lumen:NN:3:3:f
154 3-hydroxyanthranilate:VB:3:1:f 3,4-dioxygenase:CD:3:2:f activity:NN:3:3:f
155 RNA:NNP:3:1:f cap:NN:3:2:f binding:NN:3:3:f
156 transcription:NN:3:1:f export:NN:3:2:f complex:JJ:3:3:f
157 RNA:NNP:3:1:f exon:NN:3:2:f ligation:NN:3:3:f
158 spliceosomal:JJ:3:1:f snRNP:NN:3:2:f assembly:NN:3:3:f
159 spliceosomal:JJ:3:1:f complex:JJ:3:2:f disassembly:NN:3:3:f
160 Y-form:NNP:3:1:f DNA:NNP:3:2:f binding:NN:3:3:f
161 loop:NN:3:1:f DNA:NNP:3:2:f binding:NN:3:3:f
162 bubble:NN:3:1:f DNA:NNP:3:2:f binding:NN:3:3:f
163 histone:NN:3:1:f peptidyl-prolyl:JJ:3:2:f isomerization:NN:3:3:f
164 protein:NN:3:1:f peptidyl-prolyl:JJ:3:2:f isomerization:NN:3:3:f
165 autophagic:JJ:3:1:f vacuole:NN:3:2:f membrane:NN:3:3:f
166 core:NN:3:1:f TFIIH:NN:3:2:f complex:JJ:3:3:f
167 SSL2-core:CD:3:1:f TFIIH:NN:3:2:f complex:JJ:3:3:f
```

Figure 5-4: Extract the dataset 2 with features

The figure 5-4 is shown a small subset of training dataset without annotations. As the figure 5-4 shown, the training dataset have not annotated yet. All the terms (length is bigger than 1)'s roots were marked into “f”.

The figure 5-6 is shown our training data set with the annotations. According to the supervised machine learning method, the class should be pre-defined. After extracting the training data set shown in figure 5-4, we classified the training data set and got the training data set with annotations shown in figure 5-6.

As the figure 5-6 shown, for number 101 term 'histidine:NN:3:1:f biosynthetic:JJ:3:2:f process:NN:3:3:t', token 'histidine' NN means POS was marked NN(noun), '3' means this term has three tokens, '1' means the token 'histidine' is in the first position, 'f' means the token 'histidine' was not marked root.

There are 453 terms in our training data set. Then we divide the rest data set into 9 test data set and each test data set has 405 terms. Finally we get 18 test data sets results, 9 test data sets with possibility and the other 9 test data sets without possibility. We also can see the accuracy for each test data set from the result. The accuracy is defined as the proportion that the right numbers of terms marked as roots in test data set. The accuracy is calculated in following equation 5-5:

$$\text{Accuracy} = \frac{\text{The right numbers of terms marked as roots}}{\text{numbers of terms in each test dataset}}$$

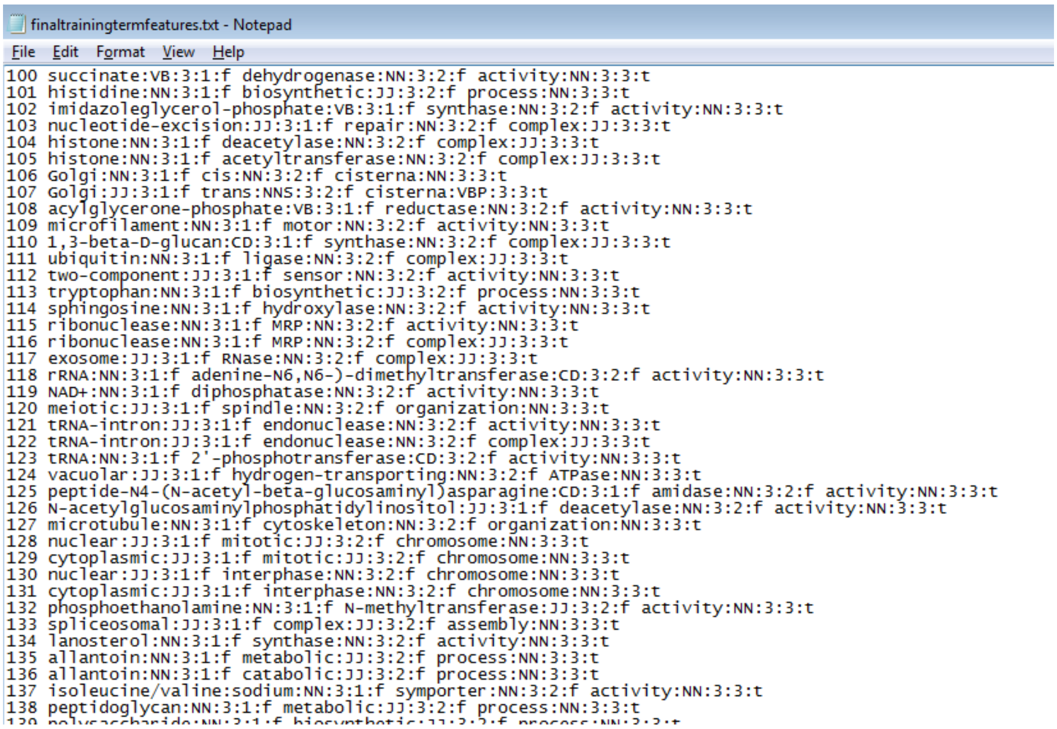
Equation 5-5: The accuracy calculation formula

Figure 5-7 showed test dataset 1 with probability. For example, in figure 5-7, for term 'radial: 4.749670099285197E-10 spokehead: 3.9752693418534165E-9 ROOT: spokehead', spokehead: 3.9752693418534165E-9 means the probability that token 'spkoehead' was recognized the roots for this term was 3.9752693418534165E-9. It is bigger than 4.749670099285197E-10. So this token 'transport' is the root for this term.

Figure 5-8 showed the test dataset 1 without probability. For example, in figure 5-8, for number 114 term 'guanine nucleotide transport', the root is transport.

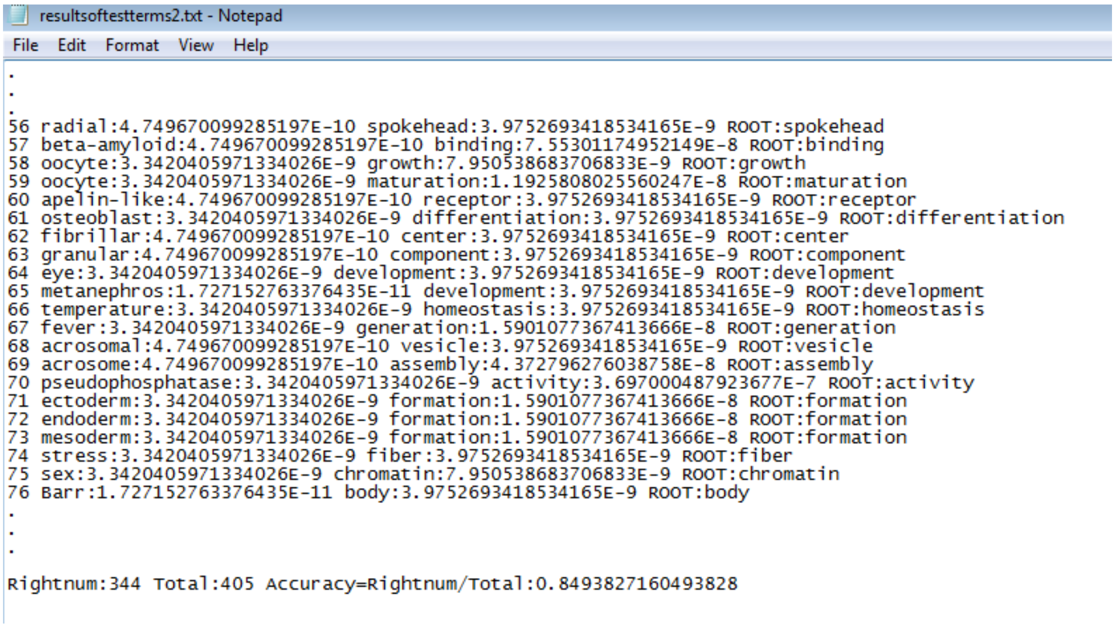
There are some terms were marked wrong roots. For example, in figure 5-8, for number 115 term 'skeletal system development', we marked the token 'system' into root, but the real root should be 'development'. Our results not only show the roots of terms effectively, but also can show the Exact Root of terms was marked wrong.

The table 5-9 shows the accuracy of 9 sets of test data set. We got 9 test data files without possibility and 9 test data files with possibility and got the 88.25% average accuracy.



```
100 succinate:VB:3:1:f dehydrogenase:NN:3:2:f activity:NN:3:3:t
101 histidine:NN:3:1:f biosynthetic:JJ:3:2:f process:NN:3:3:t
102 imidazoleglycerol-phosphate:VB:3:1:f synthase:NN:3:2:f activity:NN:3:3:t
103 nucleotide-excision:JJ:3:1:f repair:NN:3:2:f complex:JJ:3:3:t
104 histone:NN:3:1:f deacetylase:NN:3:2:f complex:JJ:3:3:t
105 histone:NN:3:1:f acetyltransferase:NN:3:2:f complex:JJ:3:3:t
106 Golgi:NN:3:1:f cis:NN:3:2:f cisterna:NN:3:3:t
107 Golgi:JJ:3:1:f trans:NNS:3:2:f cisterna:VBP:3:3:t
108 acylglycerone-phosphate:VB:3:1:f reductase:NN:3:2:f activity:NN:3:3:t
109 microfilament:NN:3:1:f motor:NN:3:2:f activity:NN:3:3:t
110 1,3-beta-D-glucan:CD:3:1:f synthase:NN:3:2:f complex:JJ:3:3:t
111 ubiquitin:NN:3:1:f ligase:NN:3:2:f complex:JJ:3:3:t
112 two-component:JJ:3:1:f sensor:NN:3:2:f activity:NN:3:3:t
113 tryptophan:NN:3:1:f biosynthetic:JJ:3:2:f process:NN:3:3:t
114 sphingosine:NN:3:1:f hydroxylase:NN:3:2:f activity:NN:3:3:t
115 ribonuclease:NN:3:1:f MRP:NN:3:2:f activity:NN:3:3:t
116 ribonuclease:NN:3:1:f MRP:NN:3:2:f complex:JJ:3:3:t
117 exosome:JJ:3:1:f RNase:NN:3:2:f complex:JJ:3:3:t
118 rRNA:NN:3:1:f adenine-N6,N6-)-dimethyltransferase:CD:3:2:f activity:NN:3:3:t
119 NAD+:NN:3:1:f diphosphatase:NN:3:2:f activity:NN:3:3:t
120 meiotic:JJ:3:1:f spindle:NN:3:2:f organization:NN:3:3:t
121 tRNA-intron:JJ:3:1:f endonuclease:NN:3:2:f activity:NN:3:3:t
122 tRNA-intron:JJ:3:1:f endonuclease:NN:3:2:f complex:JJ:3:3:t
123 tRNA:NN:3:1:f 2'-phosphotransferase:CD:3:2:f activity:NN:3:3:t
124 vacuolar:JJ:3:1:f hydrogen-transporting:NN:3:2:f ATPase:NN:3:3:t
125 peptide-N4-(N-acetyl-beta-glucosaminyl)asparagine:CD:3:1:f amidase:NN:3:2:f activity:NN:3:3:t
126 N-acetylglucosaminylphosphatidylinositol:JJ:3:1:f deacetylase:NN:3:2:f activity:NN:3:3:t
127 microtubule:NN:3:1:f cytoskeleton:NN:3:2:f organization:NN:3:3:t
128 nuclear:JJ:3:1:f mitotic:JJ:3:2:f chromosome:NN:3:3:t
129 cytoplasmic:JJ:3:1:f mitotic:JJ:3:2:f chromosome:NN:3:3:t
130 nuclear:JJ:3:1:f interphase:NN:3:2:f chromosome:NN:3:3:t
131 cytoplasmic:JJ:3:1:f interphase:NN:3:2:f chromosome:NN:3:3:t
132 phosphoethanolamine:NN:3:1:f N-methyltransferase:JJ:3:2:f activity:NN:3:3:t
133 spliceosomal:JJ:3:1:f complex:JJ:3:2:f assembly:NN:3:3:t
134 lanosterol:NN:3:1:f synthase:NN:3:2:f activity:NN:3:3:t
135 allantoin:NN:3:1:f metabolic:JJ:3:2:f process:NN:3:3:t
136 allantoin:NN:3:1:f catabolic:JJ:3:2:f process:NN:3:3:t
137 isoleucine/valine:sodium:NN:3:1:f symporter:NN:3:2:f activity:NN:3:3:t
138 peptidoglycan:NN:3:1:f metabolic:JJ:3:2:f process:NN:3:3:t
139 polysaccharide:NN:3:1:f biosynthetic:JJ:3:2:f process:NN:3:3:t
```

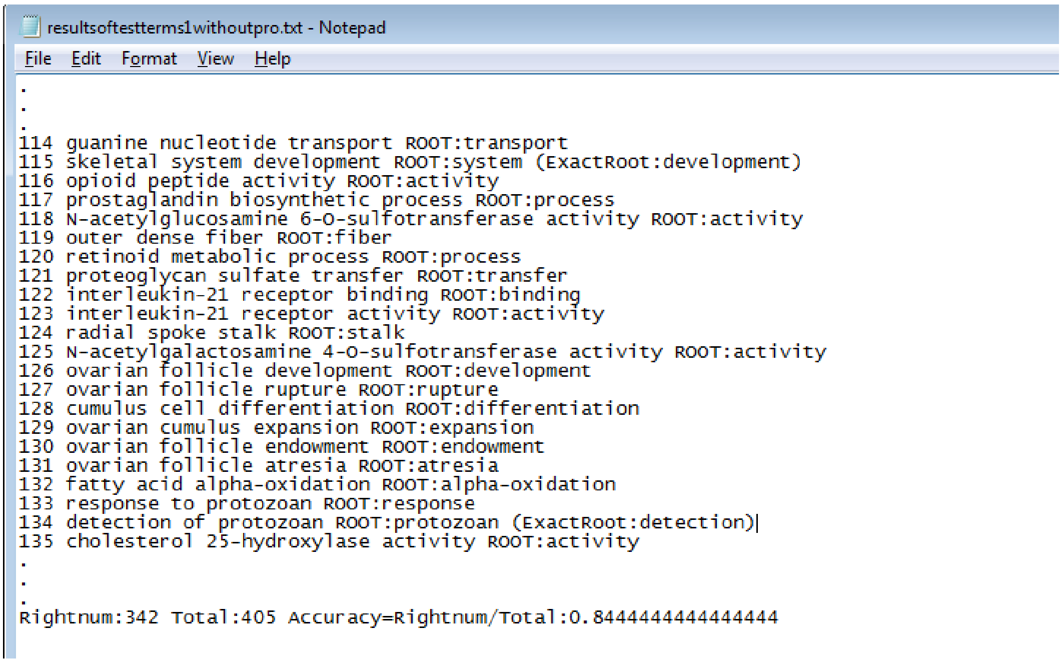
Figure 5-6: The training dataset with classified



```
.
.
.
56 radial:4.749670099285197E-10 spokehead:3.9752693418534165E-9 ROOT:spokehead
57 beta-amyloid:4.749670099285197E-10 binding:7.55301174952149E-8 ROOT:binding
58 oocyte:3.3420405971334026E-9 growth:7.950538683706833E-9 ROOT:growth
59 oocyte:3.3420405971334026E-9 maturation:1.1925808025560247E-8 ROOT:maturation
60 apelin-like:4.749670099285197E-10 receptor:3.9752693418534165E-9 ROOT:receptor
61 osteoblast:3.3420405971334026E-9 differentiation:3.9752693418534165E-9 ROOT:differentiation
62 fibrillar:4.749670099285197E-10 center:3.9752693418534165E-9 ROOT:center
63 granular:4.749670099285197E-10 component:3.9752693418534165E-9 ROOT:component
64 eye:3.3420405971334026E-9 development:3.9752693418534165E-9 ROOT:development
65 metanephros:1.727152763376435E-11 development:3.9752693418534165E-9 ROOT:development
66 temperature:3.3420405971334026E-9 homeostasis:3.9752693418534165E-9 ROOT:homeostasis
67 fever:3.3420405971334026E-9 generation:1.5901077367413666E-8 ROOT:generation
68 acrosomal:4.749670099285197E-10 vesicle:3.9752693418534165E-9 ROOT:vesicle
69 acrosome:4.749670099285197E-10 assembly:4.372796276038758E-8 ROOT:assembly
70 pseudophosphatase:3.3420405971334026E-9 activity:3.697000487923677E-7 ROOT:activity
71 ectoderm:3.3420405971334026E-9 formation:1.5901077367413666E-8 ROOT:formation
72 endoderm:3.3420405971334026E-9 formation:1.5901077367413666E-8 ROOT:formation
73 mesoderm:3.3420405971334026E-9 formation:1.5901077367413666E-8 ROOT:formation
74 stress:3.3420405971334026E-9 fiber:3.9752693418534165E-9 ROOT:fiber
75 sex:3.3420405971334026E-9 chromatin:7.950538683706833E-9 ROOT:chromatin
76 Barr:1.727152763376435E-11 body:3.9752693418534165E-9 ROOT:body
.
.
.

Rightnum:344 Total:405 Accuracy=Rightnum/Total:0.8493827160493828
```

Figure 5-7: The test dataset1 with probability



```
.
.
.
114 guanine nucleotide transport ROOT:transport
115 skeletal system development ROOT:system (ExactRoot:development)
116 opioid peptide activity ROOT:activity
117 prostaglandin biosynthetic process ROOT:process
118 N-acetylglucosamine 6-O-sulfotransferase activity ROOT:activity
119 outer dense fiber ROOT:fiber
120 retinoid metabolic process ROOT:process
121 proteoglycan sulfate transfer ROOT:transfer
122 interleukin-21 receptor binding ROOT:binding
123 interleukin-21 receptor activity ROOT:activity
124 radial spoke stalk ROOT:stalk
125 N-acetylgalactosamine 4-O-sulfotransferase activity ROOT:activity
126 ovarian follicle development ROOT:development
127 ovarian follicle rupture ROOT:rupture
128 cumulus cell differentiation ROOT:differentiation
129 ovarian cumulus expansion ROOT:expansion
130 ovarian follicle endowment ROOT:endowment
131 ovarian follicle atresia ROOT:atresia
132 fatty acid alpha-oxidation ROOT:alpha-oxidation
133 response to protozoan ROOT:response
134 detection of protozoan ROOT:protozoan (ExactRoot:detection)|
135 cholesterol 25-hydroxylase activity ROOT:activity
.
.
.
Rightnum:342 Total:405 Accuracy=Rightnum/Total:0.8444444444444444
```

Figure 5-8: The final test dataset 1 without possibility

Test dataset	The numbers of terms marked right	Total number	Accuracy
Test dataset 1	342	405	0.8444
Test dataset 2	349	405	0.8617
Test dataset 3	356	405	0.8790
Test dataset 4	354	405	0.8740
Test dataset 5	337	405	0.8321
Test dataset 6	357	405	0.8814
Test dataset 7	385	405	0.9506
Test dataset 8	373	405	0.9209
Test dataset 9	364	405	0.8987

Table 5-9: The accuracy of 9 sets of test dataset

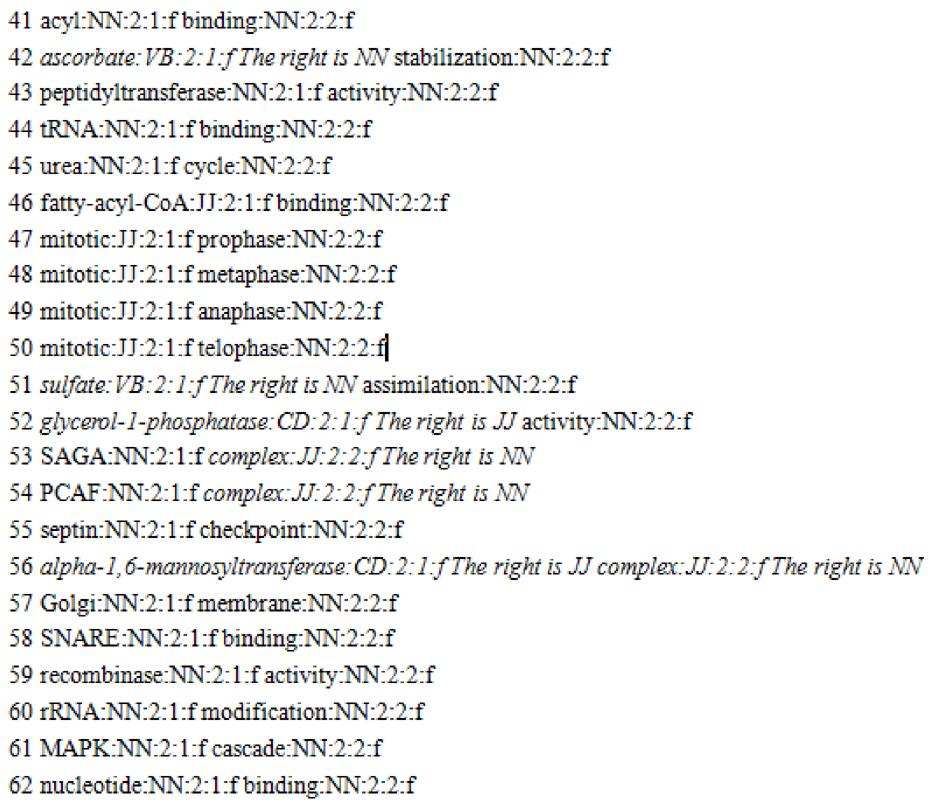
6 Analysis and discussion

6.1 Analysis the results based on the method for extracting the features

In machine learning methods, the method for extracting the features has the effected on the final results.

For feature tokenization, we used space method in java for tokenization. This method cannot avoid some deviation. It has limitation that avoiding all the special symbols of terms.

For feature POS, we used improved Brill Tagger method. However, this method is not perfect, we cannot avoid some mistakes in using this method. We calculated the numbers of tokens which were marked wrong POS. By accounting, for the training data set we chose, there are 2434 tokens, the numbers of tokens were marked wrong POS were 249, the accuracy is about 0.8971. This accuracy is defined as the proportion that the tokens marked POS rightly in the whole tokens.



41 acyl:NN:2:1:f binding:NN:2:2:f

42 *ascorbate:VB:2:1:f The right is NN* stabilization:NN:2:2:f

43 peptidyltransferase:NN:2:1:f activity:NN:2:2:f

44 tRNA:NN:2:1:f binding:NN:2:2:f

45 urea:NN:2:1:f cycle:NN:2:2:f

46 fatty-acyl-CoA:JJ:2:1:f binding:NN:2:2:f

47 mitotic:JJ:2:1:f prophase:NN:2:2:f

48 mitotic:JJ:2:1:f metaphase:NN:2:2:f

49 mitotic:JJ:2:1:f anaphase:NN:2:2:f

50 mitotic:JJ:2:1:f telophase:NN:2:2:f

51 *sulfate:VB:2:1:f The right is NN* assimilation:NN:2:2:f

52 *glycerol-1-phosphatase:CD:2:1:f The right is JJ* activity:NN:2:2:f

53 SAGA:NN:2:1:f *complex:JJ:2:2:f The right is NN*

54 PCAF:NN:2:1:f *complex:JJ:2:2:f The right is NN*

55 septin:NN:2:1:f checkpoint:NN:2:2:f

56 *alpha-1,6-mannosyltransferase:CD:2:1:f The right is JJ complex:JJ:2:2:f The right is NN*

57 Golgi:NN:2:1:f membrane:NN:2:2:f

58 SNARE:NN:2:1:f binding:NN:2:2:f

59 recombinase:NN:2:1:f activity:NN:2:2:f

60 rRNA:NN:2:1:f modification:NN:2:2:f

61 MAPK:NN:2:1:f cascade:NN:2:2:f

62 nucleotide:NN:2:1:f binding:NN:2:2:f

Figure 6-1: The training data set with POS

As the figure 6-1 shown, the italic tokens mean which were marked wrong POS. For example, for term 53 'SAGA complex', the token 'complex' was marked JJ and its real POS should be NN.

The quality of method used for tokenization will decide how many tokens we can get in the dataset. For the feature length, it is based on how many tokens we can get. Thus, the method's quality of tokenization will also have effect on the feature length. For position, we used the array to define the tokens' position. The array's length is the numbers of tokens in each term. It is based on how many tokens we got according to the tokenization. Thus, the quality of tokenization also has the effect on the feature position.

Therefore, the method for extracting the features has the effect on the quality and accuracy of classification.

6.2 Analysis the results based on the size of training data set

As the table 6-2 shown, the training data set has 453 terms. It covers different features mentioned in classification model. If we changed the training data set from 453 terms into 200 terms and the new training data set not covered all the different features, we found the new test data set results have low accuracy. This is indicated that the quality and chose of training data set is important and has a directly effect on the quality of classification model.

Test dataset	The numbers of terms marked right	Total number	Accuracy
Test dataset 1	242	405	0.5975
Test dataset 2	249	405	0.6148
Test dataset 3	256	405	0.6321
Test dataset 4	254	405	0.6272
Test dataset 5	237	405	0.5852
Test dataset 6	257	405	0.6346
Test dataset 7	285	405	0.7037

Test dataset 8	273	405	0.6741
Test dataset 9	264	405	0.6519

Table 6-2: The test data set result with training data with 200 terms

6.3 Analyse the results based on the final test data result

As the table 5-9 shown, the test data set has got an effective average accuracy 88.25%. But there are still some terms were marked into wrong roots. For test data set 1, there are 63 terms were marked into wrong. By accounting, there are 14 terms were marked into wrong may because the tokens were not in the training dataset. For example, for term ‘protein: 6.964703613691954E-6 deneddylation: 3.96570213123213E-5 ROOT: protein’, the exact root is deneddylation. It may because the token deneddylation is not in the training data set and the training dataset has the limitation of size that it cannot cover all types of terms.

There are 25 terms were marked into wrong may because the tokens were marked into wrong POS. For example, for term ‘transition: 2.894340388125493E-5 pore: 1.894340388125493E-5 complex: 1.79734039127896E-5 ROOT: transition’, the exact root is complex. We marked the JJ for token ‘complex’, the real POS is NN. Therefore we got the wrong root.

There are 24 terms were marked into wrong because of model error. Our classification model for identify the roots is not perfect, it cannot avoid the error. For example, for term ‘urogenital: 0.0 system: 2.0657645123091753E-5 development: 3.45789213212354E-5 ROOT: system’, we checked each token’s POS is right and the exact root is ‘development’. However the token ‘development’ probability is bigger than token ‘system’, but we marked ‘system’ into root. It may because of system error.

Test data set	The numbers of terms marked wrong	The numbers of terms marked wrong (Model error maybe)	The numbers of terms marked wrong (POS error maybe)	The numbers of terms marked wrong (Training set error maybe)
Test data set 1	63	14	25	24

Test data set 2	56	11	23	22
Test data set 3	49	9	21	19
Test data set 4	51	10	21	20
Test data set 5	68	17	28	23
Test data set 6	48	10	20	18
Test data set 7	20	5	8	7
Test data set 8	32	8	13	11
Test data set 9	41	11	16	14

Table 6-3: The terms marked wrong in three ways

As the table 6-3 shown, there are three causes that the roots were marked into wrong. Most terms were marked into wrong probably because the wrong POS and training dataset error.

7 Conclusion and Future work

The first research question “What is the method for identification of the roots of terms” was finely answered in chapter 2, 4 as well as second question “What general framework is needed regarding support of the method”.

In this master thesis, we determined the method for classification. According to classification model, we divide two categories, one is that token was recognized as roots and the other one is that token was not recognized as roots. Based on the four features, we calculate and compare the probabilities for each token. Finally we identify the roots of each term. This is our theoretical process for how to identify the roots of terms.

This master thesis can be seen as a step towards to analysis of the compositional structure of terms in biomedical field. The proposed general classification model which aims to identify the roots of terms in biomedical field was implemented in GO. The proposed framework of supervised machine learning can be tailored to any biomedical ontology.

Our classification model got the satisfactory results. In particular, we develop an effective POS method based on Brill tagging method. We have succeed in generating a general method for identify the roots of terms in biomedical and implemented in GO.

Our method is depending on the quality of classification model and algorithm for calculating the probability. Following our result, future work could possibly improve our classification model and try to use other algorithm. In our project, we only identify the roots of terms. We will analyse the compositional structure of terms not just roots in the further work. For POS, we will improve our rule-based POS method in future research.

In short, we have succeed in generating a classification model that judging which token is root in GO used the Naïve Bayes algorithm. The model got the effective result. In future work, we will try to use the other field’s data for implemented and use the other algorithms for calculating the probabilities. Then compare the difference to the Naïve Bayes algorithm. Finally we will do the term parsing according to this master thesis.

8 References

[1]Mulligan, K., Language, Truth and Ontology (Philosophical Studies Series), Dordrecht/Boston/London: Kluwer, 1992.

[2]Ringwald M., GXD: a gene expression database for the laboratory mouse current status and recent enhancements. Nuckeic Acids Res.28 115-119, 2000.

[3]David G., John H., “Genetic Algorithms and Machine Learning”, pp95-99, October 1988.

[4]Staab S., Knowledge Processes and Ontologies, IEEE Intelligent Systems,vol.16, pp26–34, no. 1, 2001.

[5]Smith, B., Parts and Moments. Studies in Logic and Formal Ontology, Munich: Philosophia, 1982.

[6]Tim B.L., James H., and Ora L. The semantic web. Scientic American, pp34-43, 2001.

[7]Tan P.N., Michael S., Vipin K., Introduction of Data Mining, Beijing: People’s Posts and Telecommunication Press, 2007.

[8]Nat G., Gene Ontology: tool for the unification of biology, pp25–29, May 2000.

[9]Mewes H.W., MIPS: a database for genomes and protein sequences. Nucleic Acids Res. 28, pp37–40, 2000.

[10]Mitchell T. M., Machine Learning, New York: McGraw-Hill, 1997.

[11]Clark J., The Gene Ontology database and informatics resource, pp11-7, 2004.

[12]Sebastian F., Machine learning in automated text categorization. ACM Computing Surveys, 34(1), pp1-47, 2002.

[13]Rogati M. and Yang Y., High-performing feature selection for text classificationACM CIKM, 2002.

[14]Boutsidis C., Mahoney M.W., Drineas P., Unsupervised feature selection for principal components analysis. In Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, pp61-69, 2008.

[15]Christopher D.M., Prabhakar R., Hinrich S., Introduction to Information Retrieval, chapter 14, Text classification and Naive Bayes, Cambridge University Press, 2008.

[16]McCallum M., A comparison of Events of Models for Naïve Bayes Text Classification, Carnegie Mellon University Press, 2008.

[17]Domingos P. and Pazzani M., Beyond independence: Conditions for the optimality of the simple bayesian classifier. In Proceedings of the Thirteenth International Conference on Machine Learning (ICML-96), pp105–112, July 1996.

[18]Yin J.L., Wang H.L., Research on the POS method. Institute of Scientificand Technical Information of China, Beijing 100038, China.

[19]Hemut S., "Probabilistic Part of speech Tagging using decision trees", Institute fur machine learning, University Stuttgart, October 2007.

[20]Efron B., "Estimating the error rate of a prediction rule: improvement on cross-validation", Journal of the American Statistical Association 78(382), 316-330, 1993.

[21]Eric B., Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging. *Computational Linguistics*, December 1995.

[22]Ghauri P.N., Gronhaug K., Research methods in business studies: a practical guide, Prentice Hall, 2005.

[23]Hevner A.R., Park J., Design Science in informatics systems research, Journal of Management information systems, Volume 24, Number 3, March 2004

[24]Bostrom R.P. and Heinen J.S., MIS problems and failures: A socio-technical perspective: Part I: The causes. MIS Quarterly, 1(3), pp17-32.

[25]Mumford E., Designing Human Systems for New Technology, The ETHICS method, Manchester Business School, Manchester.

[26]Langefors B., Theoretical Analysis of Information Systems, Studentlitteratur, Sweden, Lund.

[27]Alan R., Design science in information systems research, Journal MIS Quarterly Volume 28 Issue 1, March 2004.

[28]Lundeberg M. Goldkuhl G. and Nilsson A., Systemering, Studentlitteratur, Sweden, Lund.

[29]Koppenhagen N., Gass O., Müller B. and Maedche A.,Design Science Research In Action –Anatomy Of Success Critical Activities For Rigor And Relevance, Proceedings of the 20th European Conference on Information Systems (ECIS 2012) Poster Presentation, Barcelona, Spain, 2012.

[30]Rittel H. and Webber , Planning problems are wicked problems, in Developments in Design Methodology, N., John Wiley & Sons, New York, pp135–144.

[31]Simon H., The Sciences of Artificial, 3rd edn., MIT Press, Cambridge, MA, 1996.

[32]Hevner A., Chatterjee S., Design Research in Information Systems: Theory and Practice Integrated Series in Information Systems. Vol 22, pp9-20, 2010.

[33]Peffers K., Tuunanen T., Rothenberger M., and Chatterjee S., A design science research methodology for information systems research, Journal of Management Information Systems 24 (3), pp45–77, 2008.

[34]Takeda H., Veerkamp P., Tomiyama T., and Yoshikawa H., Modeling Design Processes. AI Magazine 11, 4, pp37-48, 1990.

[35]Sein M. K., Henfridsson O., Purao S., Rossi M. and Lindgren, R., Action Design Research. MIS Quarterly, (35:1), pp37-56, 2011.

[36]Walls J. G., Widmeyer G. R. and El Sawy O. A., Building an Information System Design Theory for Vigilant EIS. Information Systems Research, 3(1), pp36-59, 1992.

[37]Gregory R.W., Design Science Research and the Grounded Theory Method: Characteristics, Differences, and Complementary Uses. Proceedings of the 18th European Conference on Information System (ECIS 2010), Pretoria, South Africa.

[38]Vaishnavi V.K., Kuechler Jr. W., Design Science Research Methods and Patterns: Innovating Information and Communication Technology, Auerbach Publications, Taylor and Francis Group, New York, USA, 2008.

[39]Shareef M. I., Rawi A.W., The Customized Database Fragmentation Technique in Distributed Database Systems: A Case Study, Jönköping University, School of Engineering, JTH, Computer and Electrical Engineering, 2012.

[40]Sochacki G., Evaluation of Software Projects, a recommendation for implementation: The iterating evaluation model. A master thesis in: Blekinge Institute of Technology, Sweden, 2002.

[41]Tan H. Rajaram K. and Nirupama B., Ontology-Driven Construction of Domain Corpus with Frame Semantics Annotations. Institutionen för datavetenskap, Linköpings universitet, Sweden. Institution for medicines technique, Linkoping's university, Sweden.

[42]Langley P., Selection of revelant features in machine learning, USA, Oct 1994.

[43]Hen R, Learning a classification model for segmentation, Computer. Sci. Div, California University, USA, Oct 2003.

9 Appendix

9.1 The part of training data set's POS evaluation results, the italic means the tokens were marked wrong

1 reproduction:NN:1:1:t
2 thioredoxin:NN:1:1:t
3 *repairosome:JJ:1:1:t   The right is NN*
4 flocculation:NN:1:1:t
5 *polarisome:JJ:1:1:t   The right is NN*
6 exocyst:NN:1:1:t
7 leptotene:NN:1:1:t
8 zygotene:NN:1:1:t
9 pachytene:NN:1:1:t
10 diplotene:NN:1:1:t
11 diakinesis:NN:1:1:t
12 macromitophagy:NN:1:1:t
13 micromitophagy:NN:1:1:t
14 macropexophagy:NN:1:1:t
15 micropexophagy:NN:1:1:t
16 karyogamy:NN:1:1:t
17 conjugation:NN:1:1:t
18 cytogamy:NN:1:1:t
19 kinetochore:NN:1:1:t
20 chromatin:NN:1:1:t
21 *nucleosome:JJ:1:1:t   The right is NN*
22 euchromatin:NN:1:1:t
23 heterochromatin:NN:1:1:t
24 cytokinesis:NN:1:1:t
25 ossification:NN:1:1:t
26 fibrillin:NN:1:1:t
27 globin:NN:1:1:t
28 angiogenesis:NN:1:1:t
29 microfibril:NN:1:1:t
30 elastin:NN:1:1:t
31 elastin:NN:1:1:t
32 mitochondrion:NN:2:1:f inheritance:NN:2:2:f
33 *alpha-1,6-mannosyltransferase:CD:2:1:f   The right is JJ* activity:NN:2:2:f
34 trans-hexaprenyltranstransferase:JJ:2:1:f activity:NN:2:2:f
35 vacuole:NN:2:1:f inheritance:NN:2:2:f
36 lactase:NN:2:1:f activity:NN:2:2:f
37 alpha-glucoside:JJ:2:1:f transport:NN:2:2:f
38 *alpha-1,2-mannosyltransferase:CD:2:1:f   The right is JJ* activity:NN:2:2:f
39 mannosyltransferase:NN:2:1:f activity:NN:2:2:f
40 *alpha-1,3-mannosyltransferase:CD:2:1:f   The right is JJ* activity:NN:2:2:f
41 acyl:NN:2:1:f binding:NN:2:2:f
42 *ascorbate:VB:2:1:f The right is NN* stabilization:NN:2:2:f
43 peptidyltransferase:NN:2:1:f activity:NN:2:2:f

44 tRNA:NN:2:1:f binding:NN:2:2:f
45 urea:NN:2:1:f cycle:NN:2:2:f
46 fatty-acyl-CoA:JJ:2:1:f binding:NN:2:2:f
47 mitotic:JJ:2:1:f prophase:NN:2:2:f
48 mitotic:JJ:2:1:f metaphase:NN:2:2:f
49 mitotic:JJ:2:1:f anaphase:NN:2:2:f
50 mitotic:JJ:2:1:f telophase:NN:2:2:f
51 *sulfate:VB:2:1:f The right is NN* assimilation:NN:2:2:f
52 *glycerol-1-phosphatase:CD:2:1:f The right is JJ* activity:NN:2:2:f
53 SAGA:NN:2:1:f *complex:JJ:2:2:f The right is NN*
54 PCAF:NN:2:1:f *complex:JJ:2:2:f The right is NN*
55 septin:NN:2:1:f checkpoint:NN:2:2:f
56 *alpha-1,6-mannosyltransferase:CD:2:1:f The right is JJ complex:JJ:2:2:f The right is NN*
57 Golgi:NN:2:1:f membrane:NN:2:2:f
58 SNARE:NN:2:1:f binding:NN:2:2:f
59 recombinase:NN:2:1:f activity:NN:2:2:f
60 rRNA:NN:2:1:f modification:NN:2:2:f
61 MAPK:NN:2:1:f cascade:NN:2:2:f
62 nucleotide:NN:2:1:f binding:NN:2:2:f
63 *3'-5'-exoribonuclease:CD:2:1:f The right is JJ* activity:NN:2:2:f
64 rDNA:NN:2:1:f binding:NN:2:2:f
65 protein:NN:2:1:f polyubiquitination:NN:2:2:f
66 nuclear:JJ:2:1:f chromosome:NN:2:2:f
67 cytoplasmic:JJ:2:1:f chromosome:NN:2:2:f
68 astral:JJ:2:1:f microtubule:NN:2:2:f
69 mitotic:JJ:2:1:f prometaphase:NN:2:2:f
70 pericentriolar:JJ:2:1:f material:NN:2:2:f
71 commitment:NN:2:1:f *complex:JJ:2:2:f The right is NN*
72 nitrilase:NN:2:1:f activity:NN:2:2:f
73 mitochondrial:JJ:2:1:f chromosome:NN:2:2:f
74 mitochondrial:JJ:2:1:f fission:NN:2:2:f
75 cell:NN:2:1:f fraction:NN:2:2:f
76 M:NNP:2:1:f phase:NN:2:2:f
77 nuclear:JJ:2:1:f division:NN:2:2:f
78 shmoo:NN:2:1:f orientation:NN:2:2:f
79 mitochondrial:JJ:3:1:f genome:NN:3:2:f maintenance:NN:3:3:f
80 ribosomal:JJ:3:1:f chaperone:NN:3:2:f activity:NN:3:3:f
81 *phosphopyruvate:VB:3:1:f The right is NN* hydratase:NN:3:2:f *complex:JJ:3:3:f The right is NN*
82 mitotic:JJ:3:1:f spindle:NN:3:2:f elongation:NN:3:3:f
83 maltose:NN:3:1:f metabolic:JJ:3:2:f process:NN:3:3:f
84 maltose:NN:3:1:f biosynthetic:JJ:3:2:f process:NN:3:3:f
85 maltose:NN:3:1:f catabolic:JJ:3:2:f process:NN:3:3:f
86 *mannosylphosphate:VB:3:1:f The right is NN* transferase:NN:3:2:f activity:NN:3:3:f
87 adenine:NN:3:1:f deaminase:NN:3:2:f activity:NN:3:3:f
88 autophagic:JJ:3:1:f vacuole:NN:3:2:f assembly:NN:3:3:f
89 autophagic:JJ:3:1:f vacuole:NN:3:2:f fusion:NN:3:3:f
90 Rieske:NN:3:1:f iron-sulfur:JJ:3:2:f protein:NN:3:3:f
91 citrulline:NN:3:1:f metabolic:JJ:3:2:f process:NN:3:3:f
92 *argininosuccinate:VB:3:1:f The right is NN* metabolic:JJ:3:2:f process:NN:3:3:f

93 mitochondrial:JJ:3:1:f ornithine:NN:3:2:f transport:NN:3:3:f
94 cell:NN:3:1:f cycle:NN:3:2:f checkpoint:NN:3:3:f
95 DNA:NNP:3:1:f replication:NN:3:2:f checkpoint:NN:3:3:f
96 DNA:NNP:3:1:f damage:NN:3:2:f checkpoint:NN:3:3:f
97 cell:NN:3:1:f shape:NN:3:2:f checkpoint:NN:3:3:f
98 mitotic:JJ:3:1:f anaphase:NN:3:2:f *A:DT:3:3:f The right is LS*
99 mitotic:JJ:3:1:f anaphase:NNP:3:2:f *B:NNP:3:3:f The right is LS*
100 *succinate:VB:3:1:f The right is NN* dehydrogenase:NN:3:2:f activity:NN:3:3:f
101 histidine:NN:3:1:f biosynthetic:JJ:3:2:f process:NN:3:3:f
102 *imidazoleglycerol-phosphate:VB:3:1:f The right is JJ* synthase:NN:3:2:f activity:NN:3:3:f
103 nucleotide-excision:JJ:3:1:f repair:NN:3:2:f *complex:JJ:3:3:f The right is NN*
104 histone:NN:3:1:f deacetylase:NN:3:2:f *complex:JJ:3:3:f The right is NN*
105 histone:NN:3:1:f acetyltransferase:NN:3:2:f *complex:JJ:3:3:f The right is NN*
106 Golgi:NN:3:1:f cis:NN:3:2:f cisterna:NN:3:3:f
.
.
.
2434 *Golgi:JJ:3:1:f The right is NN trans:NNS:3:2:f The right is NN cisterna:VBP:3:3:f The right is NN*