國立交通大學

資訊科學與工程研究所

碩士論文

生物文獻中蛋白質交互關係抽取之研究

Protein-Protein Interaction Extraction from Biomedical Literature

研 究 生：施曉茹

指導教授：梁 婷　教授

中華民國 九十五 年六月

生物文獻中蛋白質交互關係抽取之研究

研究生：施曉茹　　　　指導教授：梁婷博士

國立交通大學資訊科學與工程研究所

摘要

在分子生物領域中，對於分子生物學家，若能從文獻中自動抽取出具有交互關係的蛋白質配對，將有助於生物知識庫的自動化。

過去，一些研究利用自然語言的處理技術，將文獻中的語句做語法分析，再進一步，利用設定好的規則，抽取關係，然而，語法的分析是複雜且耗時的。相反的，另一些研究，利用資料探勘的技術，從大量文獻或資料中找出有用的特徵，利用特徵抽取關係，雖然避免複雜的語句分析，但常因訓練資料不足而所限制。過去多數研究以句子為主，進行關係抽取，而本篇論文，是考慮整篇摘要後，再抽取關係，避免使用複雜的語句分析，進而解決跨句關係的抽取問題。

在本篇論文，我們利用文獻資訊、生物資料庫以及網路資源提出一套二階段的辨識程序。在第一階段，我們延用過去研究所使用的樣式，來抽取句中所含關係配對；在第二階段，建構了 Naïve Bayes 分類器，來處理跨句關係的抽取，除了考慮常被使用的特徵，如詞間距離、共現詞彙、以及共現頻率外，我們另外加入了蛋白質資料庫的資訊，利用分類器，進行二元分類。我們發現除了詞間距離、共現詞彙及頻率外，共同參考文獻的相似值在分類上也扮演重要的角色。我們分別在兩個測試語料上進行實驗，得出第一階段分別可達到 41%、32%的 F 分數，經由第二階段，F 分數分別可提升到 62%、61%。

Protein-Protein Interaction Extraction from Biomedical Literature

Student: Hsiao-Ju Shih Advisor: Tyne Liang

Institute of Computer Science and Engineering
National Chiao Tung University

Abstract

In biomedical domain, extracting protein-protein interaction relations automatically from literature is helpful for biological experts to automate knowledge databases.

Some related researches utilized NLP techniques to deal with relation extraction. However, the analysis of sentence structures is complex and time-consuming. On the contrary, some researches focused on using mining techniques to discover useful features from amount of literature or data. Complex sentence analysis is avoided, but the coverage which mining techniques could deal with is usually limited by the insufficiency of training data. Our method extracts relations by considering a whole abstract. It differentiates from most methods extracting relations by considering single-sentence. Besides, our method could extract relations across sentences.

In this thesis, we applied literature information, biological databases and web resources to construct a two-stage automatic relation identification procedure. In the first stage, we adopted the patterns discovered by other research to extract interaction pairs in a sentence. In the second stage, the classifier based on Naïve Bayes model was constructed to extract the residual relations among sentences in an abstract. In addition to the frequently used features, such as the distance of entities, co-occurring words, and co-occurrence, the information of protein databases was applied in our classifier to handle binary classification. We found that the reference similarity plays a critical role. Two corpora were tested in our experiment. The result showed that in the first stage, 41% and 32% F-scores were yielded in the two corpora, respectively. After the second stage, 62% and 61% F-scores were achieved.

ACKNOWLEDGEMENTS

I am glad for participating in the Information Retrieval Laboratory. My advisor Dr. Tyne Liang teaches and helps me a lot during the two years and gives me a guide to make plans for the future. Thank Dr. Tyne Liang for her encouragement and teaching, and then this thesis can be accomplished.

I also thank the members of information retrieval laboratory including Dian-Song Wu, Chuan-Jung Chu, Chien-Fu Cheng, Chun-Ling Chen, Cheng-I Liu, Tzu-Liang Kung, Lan-Chi Lin, I-Li Chen, I-Chia Wang, Li-Hung Huang, Chuan-Yao Su, Shou-I Cheng, Shan-Chun Pan, and Cheng-Hsing Tseng. They help me and provide many suggestions for me to accomplish the thesis.

Finally, I thank my family for their encouragement and support, especially my mother.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1 Introduction

With the rapid growth of the size and complexity of biological information, the necessity for automatic methods to discover knowledge and curate various databases is becoming more critical. For example, in recent ten years, the amount of biomedical citations available by PUBMED increases about 70% and the amount of proteins available in SWISS-PROT database increases about 280%[30]. It is more and more difficult for human experts to extract knowledge from huge amount of information and keep various databases up to date.

Because the full text of biomedical literature contains wealthy information which may not be completely and newly captured through manual curation, many automatic extraction systems like GENIS[11], SUSEKI[22] and MedScan[9], were proposed to deal with information overload. In biomedical domain, protein relation extraction is one essential task for biological experts to discover useful knowledge from literature. For example, proteins interact with each other to help accomplish biological process. Extracting information on such relations can help biological experts to understand biological process. Many approaches aimed at extracting protein relations from biomedical literature have been recently proposed, ranging from statistical methods to complex natural language processing (NLP) techniques.

Some researchers[3][9][11] emphasized analyzing the structure of a sentence and capturing as many subtleties in the sentence’s interpretation as possible. GENIS[11] is an example of such systems, utilizing a parser and a semantic grammar consisting of a large of set semantic patterns. In this system, a wide variety of different relations between biological molecules could be extracted by dealing with most frequently used sentence structures. However, in such a system, the grammar may be required to redesign completely in order to apply to a different domain. Unlike GENIES, Yakushiji et al.[3] separated NLP and information extraction into different modules. They used domain-independent parsers to perform full decomposition of a sentence structure and then used domain-specific frames to extract relations. Daraselia et al.[9] proposed a complete information extraction system, Medscan, which also separated NLP and information extraction into different modules. Because a sentence may have more than one alternative syntactic structures， how to decide correct sentence structures is helpful to extract correct relations. In order to understand the semantics of a sentence, Daraselia et al. adopted an ontology constructed by manually assigning senses to about 6000 words including biological terms and various relations between them. They utilized the ontology to serve as a filter to select correct sentence structures and then convert them into frame trees. Finally, the frame trees were transformed into the set of links between proteins and the relations could be extracted. Medscan is tested on the MEDLINE abstracts dated after 1988 and the testing result showed that the precision was 91% but the recall rate was 21%. Because such information extraction systems deal with the structure of an entire sentence, they are more accurate but time-consuming. In order to improve the efficiency and reduce the workload of processors, some researchers substituted full parsers for shallow parsers[6][7][8]. They identified certain phrases, like noun phrases and the phrases around a preposition, and extract dependencies between them, like subject-object relationship without considering the structure of an entire sentence. In [Pustejovsky et al, 2002], ‘inhibit’ relations are extracting from 56 MEDLINE articles and the result showed that the precision was 90% and the recall was 57% [7]. And in [Leroy et al., 2003] 26 MEDLINE abstracts were parsed by the shallow parser and the result showed that the precision was 90% and the recall was 62% [8]. Generally speaking, NLP-based methods are more complex and time-consuming. However, the analyses of sentence structures could help to achieve high precision.

Unlike NLP techniques, some researchers were interested in discovering as numerous amount of useful information as possible from a large number of literatures and avoiding complex and time-consuming NLP processing. For example, SUISEKI [22], employed a set of patterns which were predefined manually by filtering large amounts of text to find the most frequent constructions. These patterns implicated two protein names and expressed a direct or indirect interaction. The system was implemented to detect interactions from 100 MEDLINE abstracts and the experiment showed that the system correctly detected about 40% of the individual interaction instances with 45% precision. Huang et al. [18] used a dynamic programming algorithm to discover distinguishing patterns, like “protein1 interacts with protein2”, by aligning relevant sentences and key verbs describing protein interactions. They extracted the interactions between proteins by matching these discovered patterns and the recall and precision rate were 80% and 80.5%, respectively. However, the testing experiment was implemented with 180 sentences containing two proteins rather than a whole abstract. Oyama et al.[21] extracted 5241 features that characterize each protein appearing in the interactions from several databases, like SWISS-PROT and PIR, and mined the association rules from interaction-based transactions which were presented with these features. The 5241 features belong to seven types of protein characters, including YPD categories, EC numbers described in SWISS-PROT and PIR, SWISS-PROT/PIR keywords, PROSITE motifs, Bias of the amino acids, Segment clusters, and Amino acid patterns. Oyama et al. demonstrated that the method could detect already known rules. Ramani et al.[20] took an advantage of co-occurrence analysis to extract protein pairs from 750,000 Medline abstracts. They counted the number of abstracts citing a pair of proteins, and then calculated the probability of co-occurrence under a random model based on the hyper-geometric distribution. The end result showed that 31,609 interactions among 7,748 proteins were extracted. Co-occurrence played a useful role for interaction extraction.

Because of the rapid growth of biomedical citations, complex analysis of sentence structures is not practical enough for relation extraction. Some researchers, like Huang et al.[18], focused on using patterns to extract relation without analysis of sentence structures. However, the relations extracted by patterns are limited to a sentence. The relations between two proteins locating at different sentences are not considered. More features should be considered for interaction extraction to broaden the coverage of the relations which could be found. In this thesis, a two-stage method for extracting protein-protein interactions from scientific literature is proposed. In the first stage, patterns are utilized to match sentences containing interaction relation. In the second stage, a Naïve Bayes classifier is constructed. More features are investigated by the classifier, including surface features, co-occurrence, co-citations, and protein features. The features are extracted from MEDLINE abstracts, PUBMED, and SWISS-PROT database[27]. In order to avoid the complexity of sentence analyses, we present each protein as a vector containing co-occurring words with the protein in the same sentence. And then we count the cosine value of two protein vectors to serve as the surface feature. However, interaction relations may not be described explicitly in a sentence. We combine other inter-text information, such as the distance of a protein pair, and intra-text information, such as co-occurrence of a protein pair and co-references of a protein pair. Besides, we employ protein characters which are annotated in SWISS-PROT database to improve interaction extraction by considering domain knowledge. All protein pairs in an abstracts are considered and decided whether interaction relations exist or not by counting the probabilities of the protein pairs belonging to classes.

We use two corpora as our testing data. One is collected from MEDLINE abstracts, containing 155 abstracts, and the other is collected from the references for proving interactions in DIP[2], containing 100 abstracts. We use the interaction pairs from DIP to justify our extraction method. The result shows that our approach can yield 62% and 61% F-score in both corpora, respectively.

E S A
1896

Chapter 2 Related Work

2.1 Biomedical Resources

There are many available databases covering different aspects of protein, such as protein sequence information, protein-protein interaction, signaling pathways etc. We wanted to extract useful information from literature, databases, and web sources. We employed the three famous databases, including PUBMED, DIP, and SWISS-PROT as our information sources.

2.1.1 PUBMED and MEDLINE

PUBMED is one of the services provided by Entrez and was designed to provide accesses to citations from biomedical literature.

The primary component of PUBMED's database is MEDLINE which contains bibliographic citations and abstracts from more than 4,600 biomedical journals published in the United States and 70 other countries. The database contains over 12 million citations dating from mid-1960's.

2.1.2 DIP database

The Database of Interacting Proteins (DIP)[1][2] is a database that documents experimentally determined protein-protein interactions. The database is implemented as a relational database composed of four tables, including Protein Table, Interaction

Table, Method Table, and Reference Table. Protein Table lists proteins participating in an interaction within DIP. It provides, besides the DIP accession number, cross-reference to the three major sequence databases (SWISS-PROT, GeneBank, PIR) and additional information about the proteins such as keywords, localization and cellular function. Interaction Table records binary protein pairs that have interaction relations between them. Method Table entries describe the experimental technique that has been used to determine each interaction. Reference Table lists all the references to different articles that prove protein interactions and link them to the MEDLINE database. Currently, the database records 46,463 unique protein-protein interactions between 17,556 proteins.

Protein Table and Interaction Table were used to collect relevant articles from PUBMED. In DIP, there are more than 14,000 protein interactions between from 6,500 proteins recorded in SWISS-PROT database. These protein interactions were represented as binary protein pairs which are composed of the SWISS-PROT accession numbers of both proteins participating in each protein interaction. In this thesis, the abstracts of 1,459 articles containing DIP's binary protein pairs were extracted from PUBMED and they are used as our experimental corpus. Besides, we utilized the protein pairs in Interaction Table to justify our experiment.

2.1.3 SWISS-PROT database

SWISS-PROT is a protein sequence and knowledge database[27]. Each protein entry consists of the amino acid sequence, the protein name, taxonomic data and citation information. If further information on the protein is available, the entries contain detailed annotation such as the function(s) of the protein, similarities to other proteins etc.

SWISS-PROT version 1.0 contains 172,233 entries. Appendix A shows an example of a protein entry in SWISS-PROT database.

There are more than 6,500 SWISS-PROT entries that also appear in DIP and 25,337 PUBMED references related to these entries. We collect 1,459 references in such a way that all the references which contain at least one binary interaction protein pair in the same sentence. We extract titles and abstracts from the references as our training and testing corpus.

2.2 Relation Extraction Techniques

Recently, many approaches aimed at extracting protein relations from biomedical literature have been proposed, ranging from statistical methods to complex natural language processing (NLP) techniques. Some researchers emphasized analyzing the structure of a sentence and capturing as many subtleties in the sentence's interpretation as possible. However, some researchers were interested in discovering as numerous amount of useful information as possible from a large number of literatures and avoiding complex and time-consuming processing.

2.2.1 NLP techniques

Many NLP-based relation extraction approaches have been proposed in . They dealt with texts including part of speech tagging, disambiguation, grammars, simple phrase chunking and various parsing methods. For NLP techniques, researchers put an emphasis on parsing strategies. Parsing techniques perform decomposition of a sentence and extract local or global dependencies contained in an entire sentence or multiple sentences. With the help of an effective parser, those sentences containing protein interaction components can be identified. These parsing strategies can be classified into two categories: full parsing methods and shallow ones.

In general, full parsing procedures are slower than shallow ones. Yakushiji et al.[3] utilized a full parser to convert the varieties of sentences describing the same event into an argument structure regarding the verb. Then, they extracted the relations from argument structures by domain-specific mapping rules. In order to increase its speed, two preprocessors were employed to reduce the workload of the full parser and used to recognize noun chunks and reduce parts-of-speech ambiguity, respectively. Temkin and Gilter [5] used context free grammar (CFG) so as to reduce the NLP complexities of natural language processing by focusing on domain specific structures. The result showed that the recall and precision rates of the strategy were 63.9% and 70.2%, respectively. Park et al.[4] used a more specific approach with a bi-directional incremental parser based on Combinatory Categorial Grammar (CCG). With this grammar, verbs are expected to be surrounded by a particular sentence structure. They focused their parser on a few verbs of interest and localized the target verb as well as scanned the left and right neighborhood. Their experiment yielded high precision (80%) but lower recall (48%). Daraselia et al.[9] proposed a completely automated NLP-based information extraction system, named MedScan, and used the system to extract 2976 interactions between human proteins from MEDLINE abstracts dated after 1988. The precision of the extracted information was 91%, but the recall rate was lower than 21%. Generally speaking, full-parsing based extraction is complicated, time consuming, and highly dependent on the performance of parsers.

Leroy et al.[6] considered that elementary relations are based on prepositions. They tried to build templates around the two prepositions which are “by” and “of” and extract the information that fits templates. They tested the constructed parser on fifty unseen abstracts and found that more information was extracted without sacrificing precision. Pustejovsky et al.[7] used relational parsing as a shallow parser to extract protein inhibition interactions from abstracts. The lower recall (57%) was yielded by this strategy. Leroy et al.[8] proposed a shallow parser to capture generic relations between noun phrases automatically from free text and the precision rate was up to 90%. However, the recall is lower and about 62%.

The analyses of sentence structures could help to achieve high precision. However, the recall is lower. The considered extent of extracted proteins in text for NLP-based approaches is limited to a sentence or some dependent sentences. Besides, interaction protein pairs that are not described explicitly in text they cannot deal with.

2.2.2 Mining Techniques

Unlike limited domain for dealt by NLP-based extraction, the feature space available to mining is large, and it can include words, concepts, rules, patterns, formatting, authors, references and citations. Many researches[13-24] applied mining techniques to discover significant features useful for relation extraction.

Blaschke et al.[13] used 14 pre-defined verbs and some rules to indicate protein interactions. Ono et al.[14] used surface clues on word patterns to extract protein-protein interactions from those sentences related to yeast and E.coli proteins and achieved high recall and precision rates (average recall=84.65% and average precision=93.9%). However, the test corpus collected by Ono et al. was those sentences containing at least two protein names and one of the pre-defined keywords. Blaschke et al.[22] constructed SUISEKI system where more factors were taken into account to calculate the interaction's score. In addition to action keywords, negations and the distance between names were considered to decide the score in SUISEKI.

Their experiment showed that the system correctly detected 40 interaction instances with 45% precision.

To resolve manually predefined rules, researchers made efforts on the automation of rule generation. Ding et al.[15] discussed prominent sentence attributes to predict the existence of protein interactions. These chosen attributes were related to interactor, co-occurrence, terms order, terms separation and subject-object relationship. The experimental results revealed that co-occurrence turns to be important feature for getting higher precision. Ding et al. also showed that over 0.75 interaction descriptions had one protein occurrence as the subject and the other as the object of an interaction-indicating verb phrase. Chiang et al.[17] found that phrase patterns useful to detect the functions of gene products. They discovered these patterns within sentences by sentences aligning and then used the Navie Bayes classifier for sentence classification. Huang et al.[18] used a dynamic programming algorithm to discover distinguishing patterns by aligning relevant sentences and key verbs describing protein interactions [18]. They extracted the interactions between proteins by matching these discovered patterns and the recall and precision rate were 80% and 80.5%, respectively. Skounakis et al.[23] proposed an approach that is based on using hierarchical hidden Markov models to represent the grammatical structure of the sentences. They used a shallow parser to construct a multi-level representation of each sentence and trained hierarchical HMMs to capture the regularities of the parses for both positive and negative sentences being processed.

These methods described above dealt with the extraction of the relations explicit only. Oyama et al.[21] extracted 5241 features that characterize each protein appearing in the interactions from several public databases and mined the association rules from interaction-based transactions. Their method could detect already known rules efficiently. Ramani et al.[20] took an advantage of co-citation analysis to extract protein pairs from 750,000 Medline abstracts. They counted the number of abstracts citing a pair of proteins, and then calculated the probability of co-citation under a random model based on the hyper-geometric distribution. It was found that the co-citation probability had a relationship with the accuracy of protein interaction which is identified. Hu et al.[24] considered the characteristics of the scale-free network graph and finds the local clusters based on the local density of the vertex and its neighborhood vertices. The result showed that the clusters in the network graph represent some potential complexes, which are very important for biologist to study the protein functionality.

Chapter 3 The Proposed Relation Extraction

In this thesis, a two-stages method is proposed to extract the relations between tow proteins from texts. The proposed method takes advantage of the known protein pairs existing in DIP to collect related texts and further discover useful information from literature and databases. In this chapter, we describe our method in detail and show the analysis of the experiment implemented by using our method.

3.1 Two stage Extraction Flowchart

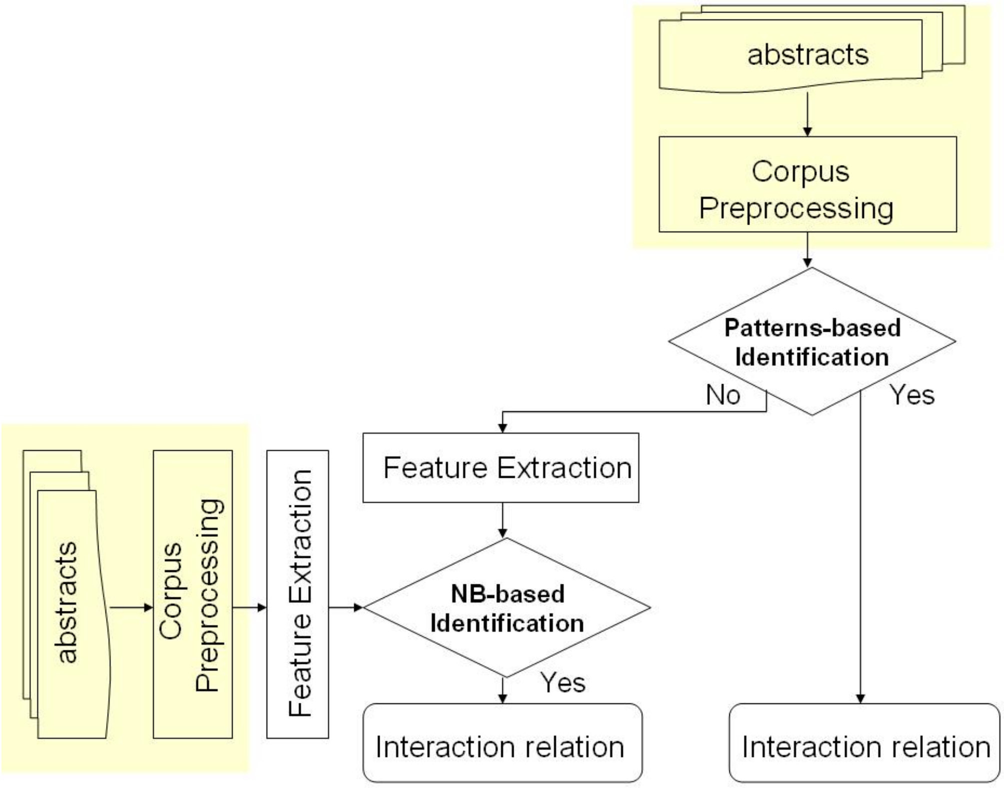


Figure 1: the flowchart of the two stages method

Figure 1 displays the extraction flowchart. The method is divided into two stages. When an abstract is inputed into our system, the entire abstract is segmented into sentences. Sentence boundaries are detected with three symbols: '.', '!', and '?'. The sentences are tokenized by converting words, numbers, and punctuation marks into separate tokens. We use SWISS-PROT database as our lexicon which contains 527,752 protein names. Then we use it to identify each protein entity in the abstract by maximum matching procedure. Besides, we tag the parts of speech of interaction-related words. Through corpus preprocessing, protein pairs are formed. And then, the protein pairs are processed by our two stage method. In the first stage, a set of predefined patterns is employed to extract relations between two proteins from the sentences. The set of patterns are adopted from the discovered patterns from [Hung, '04]. In the second stage, the classifier which is constructed by using the Naive Bayes model classifies the protein pairs into two classes: "yes" or "no" by using features which are verified with the Chi-Square test. The classifier is trained with our training data to serve to count the probability that an interaction relation is present between two proteins. Finally, the protein pairs which are not identified an interaction present between them in the first stage are again identified whether interaction relations are present between them or not by our classifier. The protein pairs which are identified as interaction pairs by our system are extracted. Due to the limit of interaction descriptions explicitly mentioned in an abstract, our system is expected to improve the coverage of the interaction pairs which can be found without sacrificing more precision.

3.2 Corpus Preprocessing

We collect 1549 MEDLINE abstracts which contain at least one interaction protein pair appearing in the same sentence. We divide the collected corpus into two parts: training set and testing set with 10-fold cross validation. There are 1,394 and

155 abstracts in the training and testing sets, respectively. Besides, we randomly select 100 abstracts that prove protein interactions referred by DIP as another testing corpus. Totally, we construct two corpora as our testing set, named ‘Corpus1’ and ‘Corpus2’, respectively. ‘Corpus1’ is collected from MEDLINE abstracts, containing 155 abstracts. And ‘Corpus2’ is collected from the references for proving interactions in DIP, containing 100 abstracts.

Figure 2: an example of how pairs are formed

There are three sentences (S1, S2, S3) and five proteins (P1, P2, P3, P4, P5) in text:
S1: P1, P2, P3
S2: P1, P4, P5
S3: P1
10 Pairs formed:
(P1,P2) (P1,P3) (P1,P4) (P1,P5) (P2,P3) (P2,P4) (P2,P5) (P3,P4) (P3,P5) (P4,P5)

In an abstract, every protein combines with another different protein to form a binary pair. By the way, $\frac{n \times (n-1)}{2}$ pairs are formed in the abstract which has n different proteins. Figure 2 shows an example of how pairs are formed. In the example, there are 10 unique pairs formed. Every abstract in corpora is transformed into a set of binary protein pairs. Table 1 shows the statistic of the corpora.

Table 1: the statistics of the corpora

	Training Corpus		**Testing Corpus**			
			Corpus1		**Corpus2**	
# abstracts	1394		155		100	
# sentences	12,373		1387		922	
# protein pairs	Yes	No	Yes	No	Yes	No
	1,773	29,543	180	3,327	529	3,245

3.3 Pattern Matching

[Huang, ‘04] tried to discover patterns that identify if an interaction is present in a sentence automatically from related sentences which were collected from PUBMED. More than five hundred patterns are adopted from [Huang ‘04] and divided into twenty-seven kinds of pattern forms as well as listed in Appendix B. Some pattern forms are described by using parts of speech and displayed in Table 2.

Table 2: some pattern forms

Pattern Form	Word Lists of pattern
P1 VBZ P2	*;modifies promotes inhibits actives mediates blocks enhances forms;*
P1 VBZ IN P2	*;interacts associates; with within ;*
P1 VBZ TO P2	*;binds; to;*
P1 VBN P2	*;linked modified promoted activated stimulated regulated enhanced;*
P1 VB IN P2	*;interact associate stimulate catalyze ; with in of;*
NNS IN P1 CCP2	interactions activations; with of between through from;*; and;*

Through the alignment between each pattern and sentence, all sentences which are intended to extract relation from are examined whether interactions are present in them. The same alignment strategy which was proposed by [Huang, ‘04] is utilized in our method. In the past, the general alignment strategy could find the longest subsequence which is matched in a sentence. However, one pattern possibly matches a sentence at different positions. The alignment strategy which was proposed by [Huang, ‘04] could be able to solve the problem and find out multiple matches in a sentence. Figure 3 shows the alignment strategy. A dynamic program is used in the alignment between each predefined pattern and sentence with the strategy. Formula (a) only allows matches to end when they score at least *T*. Threshold *T* is calculated with formula (c). The total score of all matches is obtained by adding an extra cell *F(n+1,0)* to the score matrix *F*. By tracing back from cell *(n+1,0)* to *(0,0)*, the multiple longest subsequences which are matched against the patterns in a sentence can be found.

Figure 3: the alignment strategy

For a sentence $X = (x_1, x_2, \ldots, x_n)$ and a pattern $P = (p_1, p_2, \ldots, p_m)$

$$F(0,0) = 0$$

$$F(i,0) = max\begin{cases} F(i-1,0) \\ F(i-1,j) - T, \ \ j = 1,2,\ldots,m \end{cases} \tag{a}$$

$$F(i,j) = max\begin{cases} F(i-1,j-1) + s(x_i, p_j) \\ F(i-1,j) + s(x_i, '-') \\ F(i,j-1) + s('-', p_j) \end{cases} \quad , s(x_i, p_j) = 1, s(x_i, '-') = 0, s('-', p_j) = 0 \tag{b}$$

$$T = 0.5 \times m \tag{c}$$

Following that, a filtering rule is used to filter the matched subsequences not met our requisition. If every word in the found subsequence appears in the corresponding position of the matched pattern, the subsequence is considered as the sequence representing an interaction relation. Finally, the corresponding protein pairs in the sequence are extracted and marked with class “Yes”. Figure 4 displays the whole algorithm of pattern matching.

Figure 4: pattern matching algorithm

```
Input : a pattern set $P=\{p_1,p_2,..,p_n\}$, a sequence X
Output: aligned result Set R
    for every pattern pi in P, do
    for X and the pattern pi
        1. build score matrix F using the alignment strategy;
        2. trace-back to find matches;
        3. suppose the result is Xi;
            check whether every word in Xi aligned to pi
            appears in the corresponding position of pi;
        4. if every word in Xi aligned to pi, add Xi to the result
            set R; if not, reject Xi.
      output R
```

3.4 Classification Model Constructing

All proteins in an abstract are combined with each other to form binary pairs. But two proteins which are the same are not combined with each other to form a pair. And then, all of the predefined features are extracted for each pair to form an instance in our training and testing data set. These features are tested through the Chi-square test to show that there exists correlation of each feature and classes with a high confidence degree (97.5%). The set of constructed instances contains all possible interaction pairs in the corpus. However, if a protein interacts with itself, it will be not found by using our classifier.

In this section, we explain how features are extracted for each protein pair in an abstract. Besides, in order to ensure our performance better, we select the best features for our classifier.

3.4.1 Features Extraction

Table 3: the features description

Feature	No	Description
Distance	1	The dice value of the frequencies of the protein pair in the same sentences
	2	The average of minimum distances of the protein pair in an abstract
Word	3	The cosine value of the protein pair which are presented as m-words vectors
Co-citation	4	The dice value of the frequencies of the protein pair in the same abstracts searched by the PUBMED.
	5	The maximum of reference similarities between each protein pair.
Topic	6	The similarity of the topic “function” in the SwissProt database.
	7	The similarity of the topic “similarity” in the SwissProt database.
	8	The similarity of the topic “subcellular location” in the SwissProt database.
	9	The similarity of the topic “subunit” in the SwissProt database.
	10	The similarity of the topic “catalytic activity” in the SwissProt database.

Several features are concerned in designing our classifier. Features, like distance and co-occurrence, have been verified to be useful to extract relation from text[16][20]. Features, like reference similarity, word and topic similarity, may be useful to identify whether an interaction relation exists between a protein pair or not. The predefined features are described in detail in Table 3.

For each protein pair, the values of the predefined features are extracted from the abstracts, the SWISS-PROT database and the PUBMED database.

For each protein pair (P_i, P_j), we compute feature values as follows:

Feature 1: The dice value of the protein pair co-occurring in the same sentence in the training corpus. The dice value is calculated by Equation 1.

$$dice\ value\left(P_i, P_j\right) = \frac{2\ \times \#\ of\ \ sentences\ of\ (P_i, P_j)\ co\text{-}occurring}{\#\ of\ \ sentences\ of\ P_i\ \ occurring\ +\ \#\ of\ \ sentences\ of\ P_j\ \ occurring} \qquad (1)$$

Feature 2: The average of minimum distance of the protein pair in an training abstract. The average value is calculated by Equation 2.

$$average\ value\left(P_i, P_j\right) = \frac{\sum_{i=1}^{n} the\ minimum\ sentence\ distance\ of\ (\ P_i, P_j\ )}{n} \qquad (2)$$

n : the number of abstracts P_i and P_j co - occurring

Feature 3: The cosine value of the protein pair presented as m-words vectors. Each protein is presented as m-vector. We remove stop words from the sentence where each protein locates. And then, all retained words are stemmed. The verbs, nouns and proteins co-occurring with the protein in the same sentence are served as elements of the vector and weighted by using Equation 3. The cosine value is calculated with Equation 4.

$$w_{ij} = f_{i,j} \times log\left(\frac{N}{n_i}\right) \qquad (3)$$

$w_{i,j}$: the weight of word w_i for protein P_j

N : the total number of proteins

$$f_{i,j} = \frac{freq_{i,j}}{max_l\ freq_{l,j}}$$

n_i : the number of proteins word w_i co-occurs

$freq_{i,j}$: the frequency of word w_i co-occurring with protein P_j

$f_{i,j}$: the normalization value of $freq_{i,j}$

$$sim\left(p_1, p_2\right) = \frac{\sum_{i=1}^{n} w_{i,1} \times w_{i,2}}{\sqrt{\sum_{i=1}^{n} w_{i,1}^2} \times \sqrt{\sum_{i=1}^{n} w_{i,2}^2}} \qquad (4)$$

Feature 4: The dice value of the protein pair co-occurring in the same abstracts searched by PUBMED. The dice value is calculated with Equation 5.

$$dice\ value\left(P_i,P_j\right)=\frac{2\ \times \#\ of\ \ abstracts\ (P_i,P_j\ )\ co\text{-}occur}{\#\ of\ \ abstracts\ of\ P_i\ \ appear\ +\ \#\ of\ \ abstracts\ of\ P_j\ \ appear} \qquad (5)$$

Feature 5: The reference similarity of P_i and P_j. Figure 5 shows the algorithm of calculating the similarities between the protein pair.

```
Count the similarity of the protein pair references:
   Input: $R_1\ \{r_{11},r_{12},...,r_{1p}\}$ is the reference set of $P_i$
         $R_2\ \{r_{21},r_{22},...,r_{2q}\}$ is the reference set of $P_j$
   Output: the maximum similarity between R1 and R2
    if $r_{1i}=r_{2j}$
         the maximum similarity=1
         count the ratio of (similarity=1)
             $the\ ratio\ of\ (similarity=1)\ =\ \frac{the\ number\ of\ reference\ similarity\ equaling\ 1}{p\times q}$
         output the maximum similarity and the ratio of (similarity=1)
    else
         $t_{1i}$ is the title of $r_{1i}$ , $t_{2j}$ is the title of $r_{2j}$ j
         remove the stop words from $t_{1i}$ and $t_{2j}$
             $w_{ij}=f_{i,j}\times log\left(\frac{N}{n_i}\right)$        N: the total number of all titles
             $f_{i,j}=\frac{freq_{i,j}}{max_l\ freq_{l,j}}$        ni: the number of titles that wi occurs in
         count the cosine value between $t_{1i}$ and $t_{2j}$ as the similarity of $r_{1i}$ and $r_{2j}$:
             $sim\left(t_{1i},t_{2j}\right)=\frac{\sum_{i=1}^{n}w_{i,1}\times w_{i,2}}{\sqrt{\sum_{i=1}^{n}w_{i,1}^2}\times\sqrt{\sum_{i=1}^{n}w_{i,2}^2}}$
         find the maximum similarity of $R_1$ and $R_2$:
             $the\ maximum\ similarity=max_k\ sim_{i,j}$
         Output the maximum similarity
```

Figure 5: the algorithm of reference similarity between a protein pair

Feature 6, 7, 8, 9, and 10: The similarities of all topics for each protein pair are calculated. We retain the topics whose similarities have obvious correlation with the classes as our features. As a result, five topics are retained including "function", "similarity", "subcellular location", "subunit", and "catalytic activity". Figure 6 displays the algorithm of extracting the values of these features.

Figure 6: the algorithm of extracting the database feature 6, 7, 8, 9, and 10.

Count the similarity of the protein pair topic description:

Input: D_1 is the topic description of P_i

D_2 is the topic description of P_j

Output: the similarity of D_1 and D_2

remove stop words from D_1 and D_2

represent D_1 and D_2 as m-vector using m words respectively

assign a weight value for each element in the two vectors:

$w_{ij} = f_{i,j} \times log\left(\frac{N}{n_i}\right)$ *N: the total number of all protein topic descriptions*

n_i: the number of topic descriptions that w_i occurs in

$f_{i,j} = \frac{freq_{i,j}}{max_l \ freq_{l,j}}$ *$freq_{i,j}$: the frequency of w_i appearing in the topic description$_j$*

count the cosine value between d_1 and d_2 as the similarity of D_1 and D_2:

$$sim(d_1, d_2) = \frac{\sum_{i=1}^{n} w_{i,1} \times w_{i,2}}{\sqrt{\sum_{i=1}^{n} w_{i,1}^2} \times \sqrt{\sum_{i=1}^{n} w_{i,2}^2}}$$

Output $sim(d_1,d_2)$

3.4.2 Chi-Square tests

In this thesis, we apply the Chi-Square test to verify whether there exists correlation between each feature and relation or not. For each predefined feature, the

Chi-Square value is calculated by using Equation 6. The Chi-Square tests of the predefined features with the training data set are described as follows.

$(v_1, v_2, ..., v_n)$ denote the values of the feature f_i. Table 4 represents the frequencies for every value of the feature and every class. T denotes the sum of R_i or the sum of C_j. Equation 6 is the formula to calculate the Chi-Square value.

Table 4: the frequencies for every feature value and every class

Class Feature	Yes	No	$\sum_{j \in Yes,No} N_{.j}$
v_1	N_{1Y}	N_{1N}	R_1
v_2	N_{2Y}	N_{2N}	R_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$
v_n	N_{nY}	N_{nN}	R_n
$\sum_{i=1}^{n} N_{i.}$	C_Y	C_N	T

$$\chi^2 = \sum_{i=1}^{n} \sum_{j \in Yes,No} \frac{\left(N_{ij} - e_{ij}\right)^2}{e_{ij}} \quad , \quad e_{ij} = \frac{R_i C_j}{T} \qquad (6)$$

For the two distance features and the word feature, it has a high confident level (97.5%) to show that both features have evident correlation with the classes.

Similar results show that the co-occurrence have evident correlation. For reference similarity, it is also observed that the distributions of the class ‘Yes’ and the class ‘No’ differentiate from each other.

For feature 6, 7, 8, 9, and 10, the distribution of the similarities of class “Yes” also differentiates evidently from the distribution of the similarities of class “No”. the distributions are showed in Appendix C.

3.4.3 Naïve Bayes Modeling

The Naïve Bayes classification model is applied to construct our classifier. Our training data is large, and the Naïve Bayes model is fit for large data. And we ensure that every feature is independent with each other. Identifying whether a protein pair is an interaction pair is considered as a binary classification task. P_i and P_j are represented as two proteins in one pair, respectively. $r(P_i, P_j)$ represents the relation between P_i and P_j. $r(P_i, P_j)$ is classified as positive or negative to show whether an interaction relation exists between P_i and P_j. For a given instance, the idea of the Naïve Bayes model is to estimate the probability that the instance belongs to a class and find the probable class where the instance has the highest probability.

For a given instance *I* and a set of classes *C*. The probability that the instance belongs to a class $c_j \in C$ is calculated. The target value output is selected by using Equation 7. F_I represents the set of the features which are extracted for the instance *I*. f_I represents each feature in the set F_I. We assume that the predefined features are independent and have no dependency among them.

$$\begin{aligned} v_{NB} &= \underset{c_j \in C}{arg\,max}\, Pr(c_j \mid I) \\ &= \underset{c_j \in C}{arg\,max} \frac{Pr(c_j)Pr(I \mid c_j)}{Pr(I)} \\ &= \underset{c_j \in C}{arg\,max}\, Pr(c_j)Pr(I \mid c_j) \\ &\approx \underset{c_j \in C}{arg\,max}\, Pr(c_j)\prod_{f_I \in F_I} Pr(f_I \mid c_j) \end{aligned} \tag{7}$$

$Pr(f_I \mid c_j)$, the probability that the feature f_I appears as the class c_j has happened is calculated from the training data. For an instance feature f_I which belongs to F_I, n values $(v_1, v_2, ..., v_n)$ are contained in the feature. Equation 8 is the formula for calculating the probability, $Pr(f_I \mid c_j)$. We use the estimate calculated by using Equation 5 to calculate the probability that each value in the feature f_I appears as the class c_j has happened. In Equation 9, $N(c_j)$ means the frequency that the class c_j totally appears in the training data and $N(v_i, c_j)$ means the frequency that the value v_i of the feature f_I also appears as the class c_j appears.

$$Pr(f_I \mid c_j) = \prod_{i=1}^{n} Pr(v_i \mid c_j) \qquad (8)$$

$$Pr(v_i \mid c_j) = \frac{N(v_i, c_j) + 1}{N(c_j) + n} \qquad (9)$$

For feature 5, the ratio of the reference pairs whose similarities are equal to 1 to all reference pairs is also considered when the maximum of reference similarities equals 1. Equation 10 is used to calculate $Pr(f_5 \mid c_j)$ when the reference similarity equals 1.

$$\begin{aligned} &Pr(f_5 \mid c_j) \\ &= Pr(similarity = 1 \mid c_j) Pr(the\ ratio\ of\ the\ similarities\ equaling\ to\ 1 \mid similarity = 1, c_j) \end{aligned} \qquad (10)$$

3.4.4 Feature Selection

In previous section, the ten selected features that have correlation with the classes with a high confidence level are all used in our classifier. However, the performance of the classifier with the ten features may not be the best. In order to improve the performance, the best combination of the features is found through the genetic algorithm. Because the genetic algorithm could be parallelized, it is fit for finding the best combination of the independent features.

The idea of the genetic algorithm is to find the best solution through choosing best candidate solutions and operating them (crossing-over, recombination) or changing the values of their parameters randomly to generate the next generation of candidate solutions. A fitness function is used to determine how good a candidate solution is. In each evolutionary step, the top-k candidate solutions are selected from the current generation by using the fitness function. And then, they are manipulated to form the next generation of candidate solutions. The process stops when new generations do not produce better solutions over a period of steps.

We apply the genetic algorithm to find the best features in the following ways. Each individual candidate solution is represented as a set of the values of the ten features with 0 or 1. The value of the feature which is 1 represents that the feature is used in our classifier. On the contrary, the value which is 0 represents that the feature is not used. In the initial state, ten individuals which differ in the values are selected randomly. We operate these selected individuals through exchanging the values among them or changing the values randomly to generate the next generation. Besides, in order to reduce the risk of running into local maxima, extra ten individuals are initiated randomly and also added to the next generation. Maximal F-score is used to evaluate each individual and top ten individuals are chosen for the next generation. The process stops when new generations do not produce better individuals over a period of steps. Figure 7 shows the flowchart of feature selection.

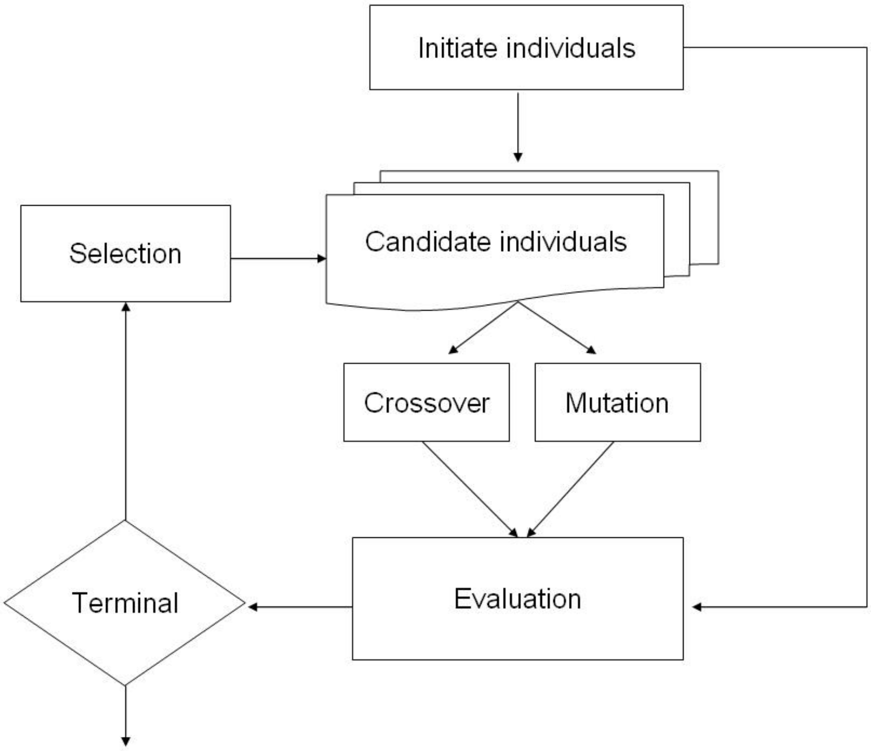


Figure 7: the flowchart of feature selection

ES A 1896

Figure 8 shows the maximal F-score for all generations. From the result, we know that the maximal F-score is a little more than 74% and the best combination of the features contain feature 1, 2, 3, 4, 5, 6, 7, and 9.

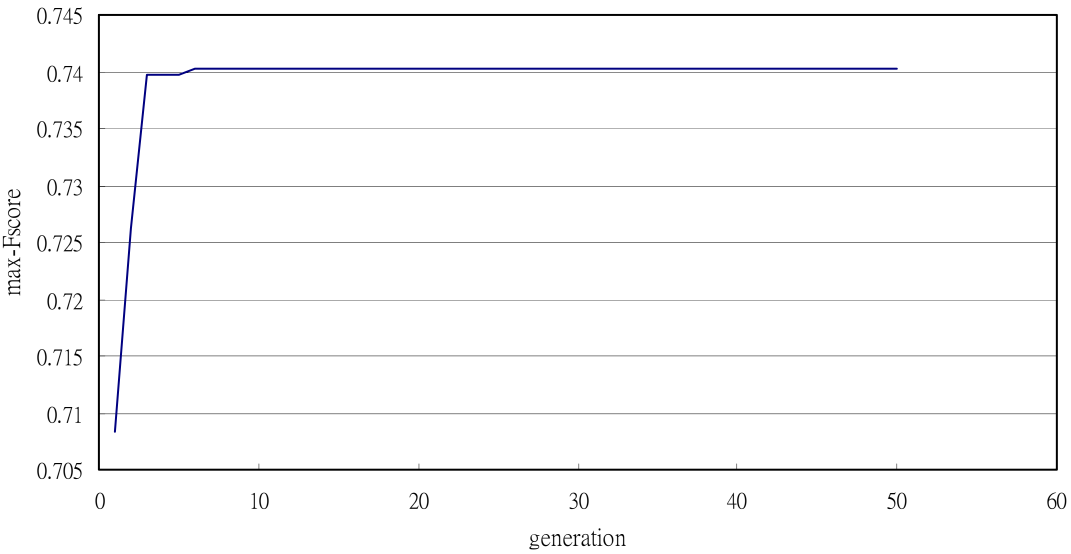


Figure 8: the maximal F-score of all generations

3.5 Experimental Results and Analysis

The experiment is implemented with the training and testing sets. The classifier is constructed by applying the Naïve Bayes model with the best features which are selected through the genetic algorithm. In order to understand how fit the classifier is, we test the training set. Table 5 shows the result of the experiment which was implemented with the training set. The F-score is a little more than 74% and about 3% increment in F-score is produced against the result of experiment with all features.

Table 5: feature selection experimental results with training corpus

Feature	TP+FP	TP	TP+FN	Precision	Recall	F-score
Total features	1,930	1,330	1,773	68.91%	75.01%	71.83%
Genetic features all-{f8,f10}	1,847	1,340	1,773	72.55%	75.58%	74.49%

Table 6 shows the impact of each feature in the training data.

Feature 5 has the most effect on the F-score. The reference similarity of each protein pair plays a critical role for interaction extraction. The distributions of the reference similarities of the class “Yes” and the class “No” differentiate from each other evidently. From the result, we know that the obvious difference between the distributions of class “Yes” and class “No” is important information for relation extraction. Some features do not have obvious impacts on the performance. Feature 8 which means the topic “subcellular location” similarity of each protein pair does not have a positive impact on the F-score. Besides, Feature 10 has little impact and it means the topic “catalytic activity” similarity of each protein pair.

Table 6: the feature impact on the training data

	features	tp+fp	tp+fn	tp	precision	recall	F-score	Diff.
Training Set	All	1930	1330	1773	68.91%	75.01%	71.83%	
	All-f1	2013	1324	1773	65.77%	74.68%	69.94%	-1.89%
	All-f2	1982	1308	1773	65.99%	73.77%	69.67%	-2.17%
	All-f3	2083	1356	1773	65.10%	76.48%	70.33%	-1.50%
	All-f4	1840	1267	1773	68.86%	71.46%	70.14%	-1.70%
	All-f5	1794	1073	1773	59.81%	60.52%	60.16%	-11.67%
	All-f6	2021	1326	1773	65.61%	74.79%	69.90%	-1.93%
	All-f7	1973	1334	1773	67.61%	75.24%	71.22%	-0.61%
	All-f8	1924	1338	1773	69.54%	75.47%	72.38%	0.55%
	All-f9	2017	1320	1773	65.44%	74.45%	69.66%	-2.18%
	All-f10	1846	1295	1773	70.15%	73.04%	71.57%	-0.27%

Table 7: relation identification results on test Corpus1

Corpus1						
	TP+FP	**TP**	**TP+FN**	**Precision**	**Recall**	**F-score**
First Stage	90	55	180	61.11%	**30.56%**	40.74%
Second Stage	141	72	125	51.06%	57.60%	54.13%
Total	231	127	180	54.98%	**70.56%**	**61.80%**

Table 8: relation identification results on test Corpus2

Corpus2						
	TP+FP	**TP**	**TP+FN**	**Precision**	**Recall**	**F-score**
First Stage	189	114	529	60.32%	**21.55%**	31.75%
Second Stage	422	235	415	55.69%	56.63%	56.15%
Total	611	349	529	57.12%	**65.97%**	**61.30%**

We use the genetic features and test the two corpora: “Corpus1” and “Corpus2” to show the performance of the two-stages method which is proposed in the thesis. The experiment result is displayed in Table 7 and Table 8, respectively. For Corpus1, in the first stage, the precision is 60.32% but recall is 30.56%. It reveals that larger percentage of the total interaction pairs still are not found by patterns matching only. In the second stage, the classifier is utilized to identify the residual pairs not extracted in the first stage whether they are interaction pairs or not. Finally, the higher recall (70.56%) could be achieved through the second stage. However, the precision is lower and about 55%. For Corpus2, through combining the first and second stages, the higher recall (65.97%) could be achieved without sacrificing more precision. In both corpora, the F-score yields about 61%.

The experiment shows that it is useful to use a statistic method to extract the protein pairs which are interaction pairs. The classifier constructed in the thesis could make up the shortage of patterns matching. The performance of the two stage method is better than both the performances of patterns matching and the classifier.

Chapter 4 Conclusion and Future Work

In this thesis, we combine the literature information, well-annotated database information, and the information stored in web resources to construct a two-stages method which could be able to identify interaction proteins in text through adding a statistic model after pattern matching. Because the description of protein interaction is variant and implicit information such as anaphor and pronoun exists in text, it is necessary to analyze the structures of sentences. However, the parsing strategies often time-consuming and it is not practical to utilize a parser to extract relations from a huge amount of literature. In order to solve the problem mentioned above, we consider all protein pairs in text and identify interaction proteins from them by using pre-defined features which are extracted from literature, well-annotated database and web resources.

We implement our system by using two corpora which are collected from SWISS-PROT references and the related articles which prove interactions in DIP, respectively and compare with the method which only uses pattern matching. The experiment result shows that the proposed system could be able to avoid the complexity of sentences structures analyses and effectively broad the coverage of interaction proteins which could be found from literature. The F-score yield 61% in the two corpora.

However, there are still several suggestions to improve:

1. Numerous and available training corpus :

   For a statistic model, available and information-rich training data is important to accomplish this kind of information extraction with a good effect. Numerous and available training data could help to broad the coverage and improve the precision.

2. Quality identification of protein name:

   In the thesis, we only use lexicon to identify proteins in text. We know that some proteins in text could not be identified by only lexicon. Besides, the variants of protein name could result in identification errors. So, it is helpful to utilize a good protein tagger to improve our system.

3. More features:

   For a statistic method, adding more features is the way to solve the problem of data sparseness. Well-annotated database information is important for our system to extract relations from text. However, only the information of SWISS-PROT database is utilized in our system. In the future, more information of other well-annotated databases could be applied, such as Gene-Ontology.

4. Patterns optimization:

   In order to avoid the complexity of sentence structures analyses, some researches focused on using pattern to extract relations from text. However, the accuracy of pattern matching is less than parsing methods. Recently, some researches found that pattern optimization could be helpful to improve the accuracy. They also considered the distances of words in a pattern and found the best condition for these distances.

5. Threshold determination:

   Some features are numeral values in our system. In order to avoid data sparseness, we divided the values into several divisions. The numeral values are transformed into the numbers of divisions. However, the division may not be optimized. To determine the threshold to divide the numeral values could be helpful for our system.

References

[1] Salwinski,L., Miller,C.S., Smith,A.J., Pettit,F.K., Bowie,J.U., Eisenberg,D. The database of interacting proteins: 2004 update. NAR 32 Database issue: D449-51.

[2] Xenarios, I., Rice,D.W., Salwinski L., Baron,M.K., Marcotte,E.M., Eisenberg.D. (2000) DIP: The database of interaction proteins. NAR 28, 289-91.

[3] Yakushiji, A., Tateisi, Y., Miyao, Y. and Tsujii, J. I.. Event Extraction From Biomedical Papers Using a full Parser. 2001.

[4] Park, J.C., Kim, H. S., Kim, J. J. Bidirectional incremental parsing for automatic pathway identification with combinatory categorical grammar. In *Proceedings of the Pacific Symposium Bio-computing*. 2001. pages 396-407.

[5] Temkin, J. M. and Gilder, M. R.. Extraction of protein interaction information from unstructured text using a context-free grammar. Bioinformatics. vol. 19. pages 2046-2053. 2003.

[6] Leroy, G. and Chen, H.. Filling Preposition-based templates to capture information from medical abstracts. In Pacific Symposium Bio-computing 2002. vol. 7. pages 350-361.

[7] Pustejovsky, J., Castano, J., Zhang, J.. Robust Relational Parsing over Biomedical Literature: Extracting Inhibit Relations. In *the Proceedings of the seventh Pacific Symposium Bio-computing* . pages 362-373. 2002

[8] Leroy, G., Chen, H., and Martinez, J. D.. A shallow parser based on closed-class words to capture relations in biomedical text. Journal of Biomedical Informatics Vol. 36. pages 145-158. 2003.

[9] Daraselia, N., Yuryev, A., Egorov, S., Novichkova, S., Niktin A. and Mazo, I..

Extracting human protein interactions from MEDLINE using a full-sentence parser. Bioinformatics Vol. 20. no. 5. pages 604-611. 2004.

[10] Pyysalo, S., Ginter, F., Pahikkala, T., Koivula, J., Boberg, J., Järvinen J. and Salakoski, T.. Analysis of Link Grammar on Biomedical Dependency Corpus Targeted at Protein-Protein Interactions.

[11] Friedman, C., Kra, P., Yu, H., Krauthammer, M., and Rzhetsky, A. Genies: a natural-language processing system for the extraction of molecular pathways from journal articles. Bioinformatics. Vol. 17. suppl.1: 74-82. 2001.

[12] Cohen, A. M. and Hersh, W. R.. A survey of current work in biomedical text mining. Briefings in Bioinformatics. Vol 6. no 1. pages 57-71. March 2005.

[13] Blaschke, C., Andrade, M. A., Ouzounis, C. and Valencia, A.. Automatic extraction of biological information from scientific text: protein-protein interactions. *American Association for Artificial Intelligence*. 1999.

[14] Ono, T., Hishigaki, H., Tanigami, A., and Takagi, T. Automatic extraction of information on protein-protein interactions from the biological literature. Bioinformatics. Vol. 17. No.2. pages 155-161. 2001.

[15] Ding, J., Berleant, D., Nettleton, D. and Wurtele, E.. Mining MEDLINE: abstracts, sentences, or phrases? Pacific Symposium on Bio-computing Vol. 7. pages 326-337. 2002.

[16] Berleant, D., Ding, J., Nettleton, D.. Corpus Properties of Protein Interaction Descriptions.

[17] Chiang J. H. and Yu H. C.. MeKE: discovering the functions of gene products from biomedical literature via sentence alignment. Bioinformatics. Vol. 19. No. 11. pages 1417-1422. 2003.

[18] Huang, M. L., Zhu, X. Y., Hao, Y., Payan, D. G., Qu, K. B., and Li, M.. Discovering patterns to extract protein-protein interactions from full texts.

Bioinformatics. Vol. 20 No.18. December 2004.

[19] Hirschman, L., Park, J. C., Tsujii, J., Wong, L. and Wu, C.H.. Accomplishments and challenges in literature data mining for biology. Bioinformatics. Vol. 18. pages 1553-1561. 2002.

[20] Ramani, A., Marcotte, E., Bunescu, R. and Mooney, R.. Using Biomedical Literature Mining to Consolidate the Set of Known Human Protein-Protein Interactions. *Proceedings of the ACL-ISMB Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics*. pages 46-53. June 2005.

[21] Oyama, T., Kitano, K., Satou, K. and Ito, T.. Extraction of knowledge on protein-protein interaction by association rule discovery. Bioinformatics. Vol. 18. No. 5. pages 705-714. 2002.

[22] Blaschke, C. and Valencia, A.. The Frame-Based Module of the SUISEKI Information Extraction System. IEEE INTELLIGENT SYSTEMS. 2002.

[23] Skounakis, M., Craven, M. and Ray, S.. Hierarchical Hidden Markov Models for Information Extraction. *Proceedings of the 18th International Joint Conference on Artificial Intelligence*. 2003.

[24] Hu, X., Yoo, I., Song, I. Y., Song, M., Han, J. and Lechner, M.. Extracting and Mining Protein-Protein Interaction Network from Biomedical Literature. IEEE. 2004.

[25] Bunescu, R., Ge, R., Kate, R. J., Marcotte, E. M., Mooney, R. J., Ramani, A. K. and Wong, Y. W.. “Comparative Experiments on Learning Information Extractors for Proteins and their Interactions” Special Issue in the Journal Artificial Intelligence in Medicine on Summarization and Information Extraction from Medical Documents. 2003.

[26] Kaneta, Y., Ahaduxxaman, M., Ohkawa, T.. A method of extracting sentences

related to protein interaction from literature using a structure database. In *Proceedings of the second European Workshop on Data Mining and Text Mining in Bioinformatics.*

[27] Boeckmann, B., Bairoch, A., Apweiler, R., Blatter, M. C., Estreicher, A., Gasteiger, E., Martin, M. J., Michoud, K., O'Donovan, C., Phan, I., Pilbout, S. and Schneider, M.. The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003. Nucleic Acids Research. Vol. 31. No.1. 2003

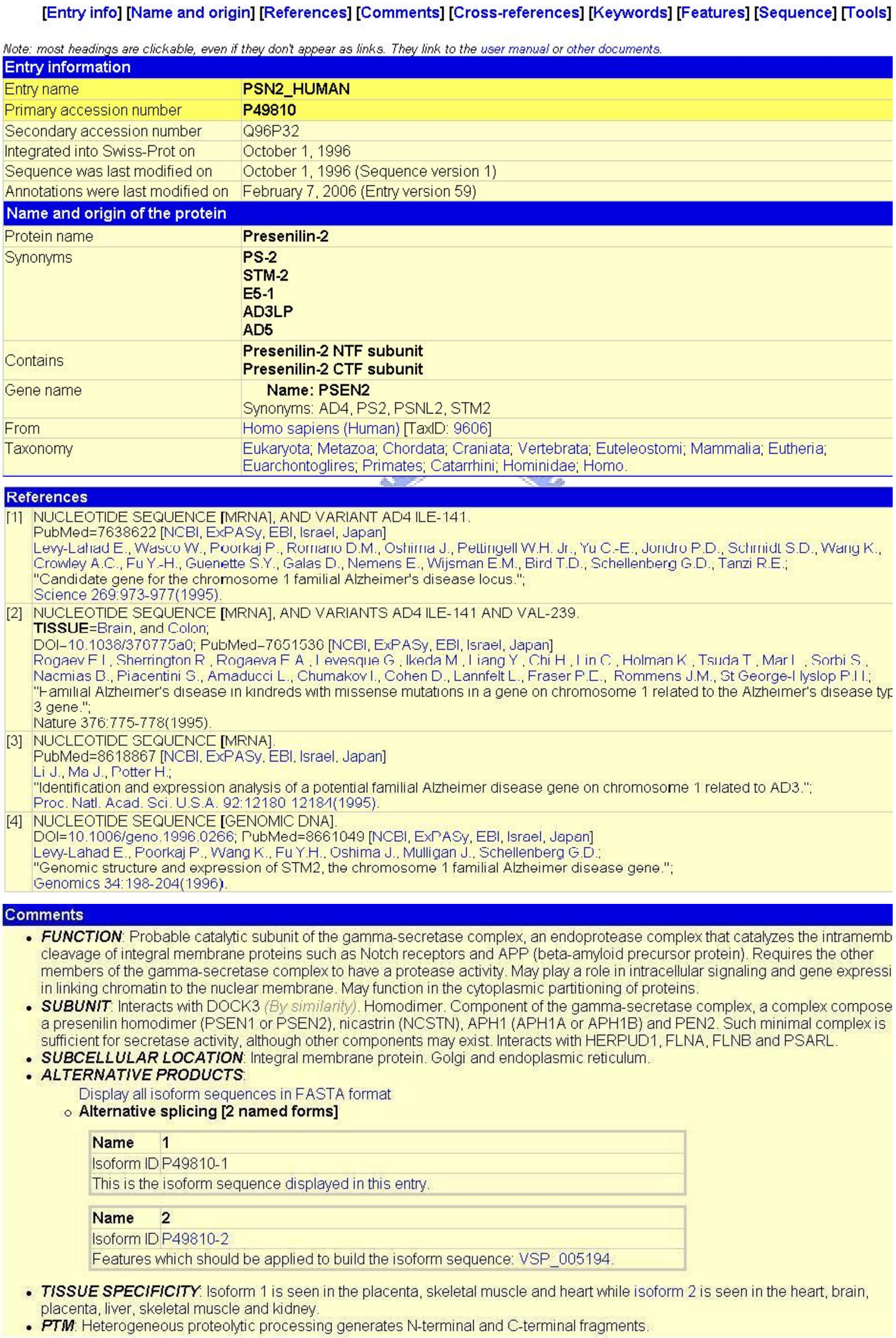
[28] Mitchell, T.. Machine Learning. McGraw Hill, New York. 1997.

[29] O`Donovan, C., Martin, M.J., Gattiker, A., Gasteiger, E., Bairoch, A. and Apweiler, R. High-quality protein knowledge resource: SWISS-PROT and TrEMBL. Brief. Bioinform.Vol. 3 pages 275-284. 2002.

[30] Shih, Ping-Ke. Automatic Protein Entities Recognition from PubMed Corpus. 2004.

Appendix A

An Example of SWISS-PROT Entry



[Entry info] [Name and origin] [References] [Comments] [Cross-references] [Keywords] [Features] [Sequence] [Tools]

Note: most headings are clickable, even if they don't appear as links. They link to the user manual or other documents.

Entry information	
Entry name	**PSN2_HUMAN**
Primary accession number	**P49810**
Secondary accession number	Q96P32
Integrated into Swiss-Prot on	October 1, 1996
Sequence was last modified on	October 1, 1996 (Sequence version 1)
Annotations were last modified on	February 7, 2006 (Entry version 59)

Name and origin of the protein	
Protein name	**Presenilin-2**
Synonyms	**PS-2** **STM-2** **E5-1** **AD3LP** **AD5**
Contains	**Presenilin-2 NTF subunit** **Presenilin-2 CTF subunit**
Gene name	**Name: PSEN2** Synonyms: AD4, PS2, PSNL2, STM2
From	Homo sapiens (Human) [TaxID: 9606]
Taxonomy	Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi; Mammalia; Eutheria; Euarchontoglires; Primates; Catarrhini; Hominidae; Homo.

References

[1] NUCLEOTIDE SEQUENCE [MRNA], AND VARIANT AD4 ILE-141.
PubMed=7638622 [NCBI, ExPASy, EBI, Israel, Japan]
Levy-Lahad E., Wasco W., Poorkaj P., Romano D.M., Oshima J., Pettingell W.H. Jr., Yu C.-E., Jondro P.D., Schmidt S.D., Wang K., Crowley A.C., Fu Y.-H., Guenette S.Y., Galas D., Nemens E., Wijsman E.M., Bird T.D., Schellenberg G.D., Tanzi R.E.;
"Candidate gene for the chromosome 1 familial Alzheimer's disease locus.";
Science 269:973-977(1995).

[2] NUCLEOTIDE SEQUENCE [MRNA], AND VARIANTS AD4 ILE-141 AND VAL-239.
TISSUE=Brain, and Colon;
DOI=10.1038/376775a0; PubMed=7651536 [NCBI, ExPASy, EBI, Israel, Japan]
Rogaev E.I., Sherrington R., Rogaeva E.A., Levesque G., Ikeda M., Liang Y., Chi H., Lin C., Holman K., Tsuda T., Mar L., Sorbi S., Nacmias B., Piacentini S., Amaducci L., Chumakov I., Cohen D., Lannfelt L., Fraser P.E., Rommens J.M., St George-Hyslop P.H.;
"Familial Alzheimer's disease in kindreds with missense mutations in a gene on chromosome 1 related to the Alzheimer's disease typ
3 gene.";
Nature 376:775-778(1995).

[3] NUCLEOTIDE SEQUENCE [MRNA].
PubMed=8618867 [NCBI, ExPASy, EBI, Israel, Japan]
Li J., Ma J., Potter H.;
"Identification and expression analysis of a potential familial Alzheimer disease gene on chromosome 1 related to AD3.";
Proc. Natl. Acad. Sci. U.S.A. 92:12180-12184(1995).

[4] NUCLEOTIDE SEQUENCE [GENOMIC DNA].
DOI=10.1006/geno.1996.0266; PubMed=8661049 [NCBI, ExPASy, EBI, Israel, Japan]
Levy-Lahad E., Poorkaj P., Wang K., Fu Y.H., Oshima J., Mulligan J., Schellenberg G.D.;
"Genomic structure and expression of STM2, the chromosome 1 familial Alzheimer disease gene.";
Genomics 34:198-204(1996).

Comments

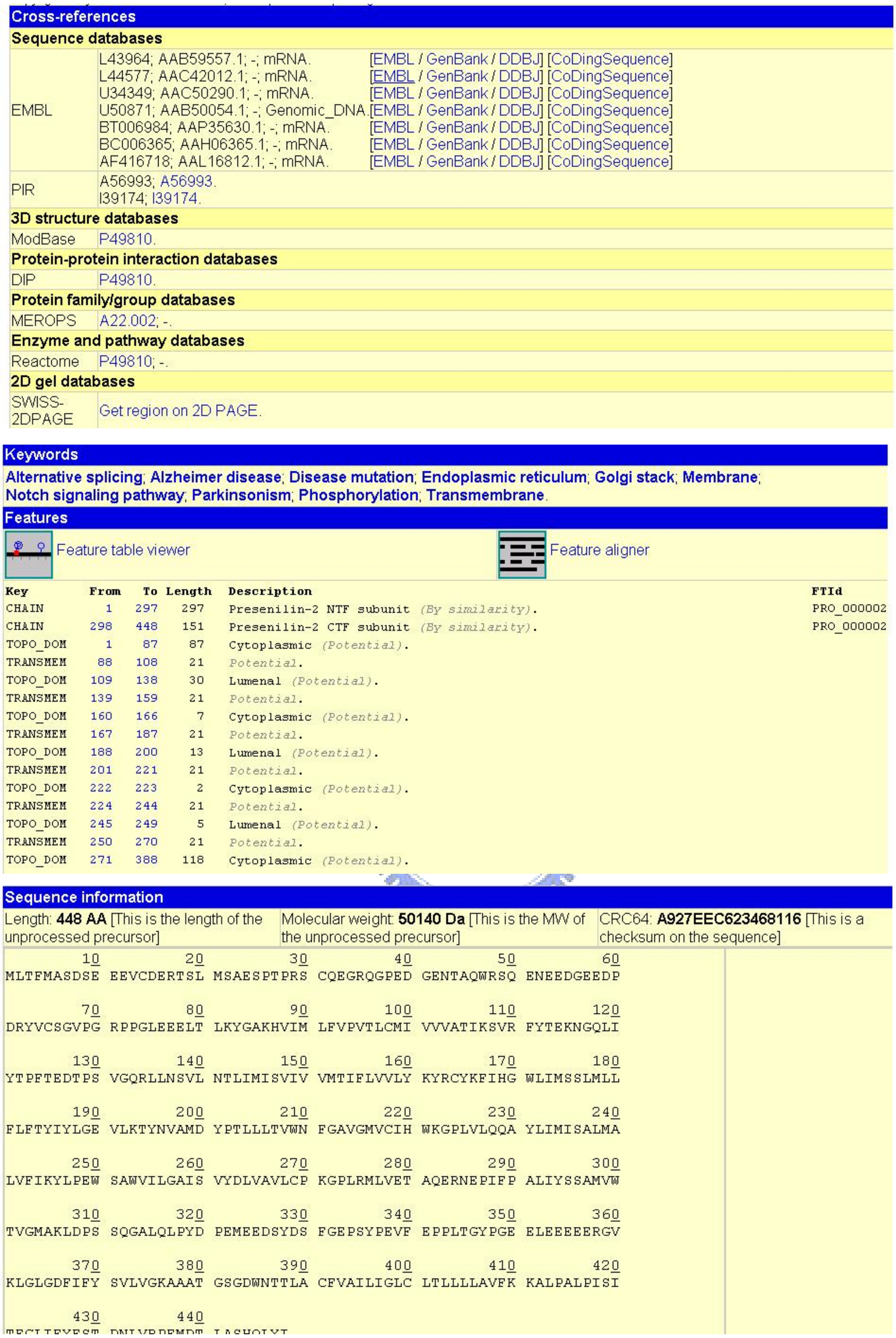
- ***FUNCTION***: Probable catalytic subunit of the gamma-secretase complex, an endoprotease complex that catalyzes the intramemb cleavage of integral membrane proteins such as Notch receptors and APP (beta-amyloid precursor protein). Requires the other members of the gamma-secretase complex to have a protease activity. May play a role in intracellular signaling and gene expressi in linking chromatin to the nuclear membrane. May function in the cytoplasmic partitioning of proteins.
- ***SUBUNIT***: Interacts with DOCK3 *(By similarity)*. Homodimer. Component of the gamma-secretase complex, a complex compose a presenilin homodimer (PSEN1 or PSEN2), nicastrin (NCSTN), APH1 (APH1A or APH1B) and PEN2. Such minimal complex is sufficient for secretase activity, although other components may exist. Interacts with HERPUD1, FLNA, FLNB and PSARL.
- ***SUBCELLULAR LOCATION***: Integral membrane protein. Golgi and endoplasmic reticulum.
- ***ALTERNATIVE PRODUCTS***:

  Display all isoform sequences in FASTA format
  - **Alternative splicing [2 named forms]**

Name	**1**
Isoform ID	P49810-1
This is the isoform sequence displayed in this entry.	

Name	**2**
Isoform ID	P49810-2
Features which should be applied to build the isoform sequence: VSP_005194.	

- ***TISSUE SPECIFICITY***: Isoform 1 is seen in the placenta, skeletal muscle and heart while isoform 2 is seen in the heart, brain, placenta, liver, skeletal muscle and kidney.
- ***PTM***: Heterogeneous proteolytic processing generates N-terminal and C-terminal fragments.



Cross-references

Sequence databases		
EMBL	L43964; AAB59557.1; -; mRNA. L44577; AAC42012.1; -; mRNA. U34349; AAC50290.1; -; mRNA. U50871; AAB50054.1; -; Genomic_DNA. BT006984; AAP35630.1; -; mRNA. BC006365; AAH06365.1; -; mRNA. AF416718; AAL16812.1; -; mRNA.	[EMBL / GenBank / DDBJ] [CoDingSequence] [EMBL / GenBank / DDBJ] [CoDingSequence] [EMBL / GenBank / DDBJ] [CoDingSequence] [EMBL / GenBank / DDBJ] [CoDingSequence] [EMBL / GenBank / DDBJ] [CoDingSequence] [EMBL / GenBank / DDBJ] [CoDingSequence] [EMBL / GenBank / DDBJ] [CoDingSequence]
PIR	A56993; A56993. I39174; I39174.	
3D structure databases		
ModBase	P49810.	
Protein-protein interaction databases		
DIP	P49810.	
Protein family/group databases		
MEROPS	A22.002; -.	
Enzyme and pathway databases		
Reactome	P49810; -.	
2D gel databases		
SWISS-2DPAGE	Get region on 2D PAGE.	

Keywords

Alternative splicing; **Alzheimer disease**; **Disease mutation**; **Endoplasmic reticulum**; **Golgi stack**; **Membrane**; **Notch signaling pathway**; **Parkinsonism**; **Phosphorylation**; **Transmembrane**.

Features

Feature table viewer

Feature aligner

Key	From	To	Length	Description	FTId
CHAIN	1	297	297	Presenilin-2 NTF subunit *(By similarity)*.	PRO_000002
CHAIN	298	448	151	Presenilin-2 CTF subunit *(By similarity)*.	PRO_000002
TOPO_DOM	1	87	87	Cytoplasmic *(Potential)*.	
TRANSMEM	88	108	21	*Potential*.	
TOPO_DOM	109	138	30	Lumenal *(Potential)*.	
TRANSMEM	139	159	21	*Potential*.	
TOPO_DOM	160	166	7	Cytoplasmic *(Potential)*.	
TRANSMEM	167	187	21	*Potential*.	
TOPO_DOM	188	200	13	Lumenal *(Potential)*.	
TRANSMEM	201	221	21	*Potential*.	
TOPO_DOM	222	223	2	Cytoplasmic *(Potential)*.	
TRANSMEM	224	244	21	*Potential*.	
TOPO_DOM	245	249	5	Lumenal *(Potential)*.	
TRANSMEM	250	270	21	*Potential*.	
TOPO_DOM	271	388	118	Cytoplasmic *(Potential)*.	

Sequence information

Length: **448 AA** [This is the length of the unprocessed precursor]	Molecular weight: **50140 Da** [This is the MW of the unprocessed precursor]	CRC64: **A927EEC623468116** [This is a checksum on the sequence]

```
        10         20         30         40         50         60
MLTFMASDSE EEVCDERTSL MSAESPTPRS CQEGRQGPED GENTAQWRSQ ENEEDGEEDP

        70         80         90        100        110        120
DRYVCSGVPG RPPGLEEELT LKYGAKHVIM LFVPVTLCMI VVVATIKSVR FYTEKNGQLI

       130        140        150        160        170        180
YTPFTEDTPS VGQRLLNSVL NTLIMISVIV VMTIFLVVLY KYRCYKFIHG WLIMSSLMLL

       190        200        210        220        230        240
FLFTYIYLGE VLKTYNVAMD YPTLLLTVWN FGAVGMVCIH WKGPLVLQQA YLIMISALMA

       250        260        270        280        290        300
LVFIKYLPEW SAWVILGAIS VYDLVAVLCP KGPLRMLVET AQERNEPIFP ALIYSSAMVW

       310        320        330        340        350        360
TVGMAKLDPS SQGALQLPYD PEMEEDSYDS FGEPSYPEVF EPPLTGYPGE ELEEEEERGV

       370        380        390        400        410        420
KLGLGDFIFY SVLVGKAAAT GSGDWNTTLA CFVAILIGLC LTLLLLAVFK KALPALPISI

       430        440
TFGLIFYFST DNLVRPFMDT LASHQLYI
```

Appendix B

List of Patterns

PTN PTN activation between PTN
PTN PTN assembly
PTN PTN association via PTN
PTN PTN complex
PTN PTN conjugation
PTN PTN conjugation with PTN
PTN PTN interaction
PTN PTN interaction of PTN
PTN PTN modification by PTN
PTN PTN producing
PTN PTN repair
PTN PTN transfer
PTN abolished PTN PTN
PTN activate PTN
PTN activate PTN PTN
PTN activated PTN
PTN activated PTN PTN
PTN activated by PTN
PTN activates PTN
PTN activates PTN PTN
PTN activates PTN and PTN
PTN activates PTN by PTN
PTN activates PTN on PTN
PTN activates PTN via PTN
PTN activation of PTN and PTN
PTN and PTN associate with PTN
PTN and PTN interact of PTN
PTN and PTN interact with PTN
PTN and PTN regulate of PTN

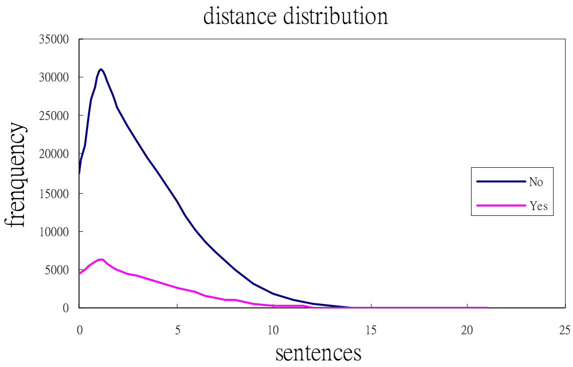
PTN assembles PTN and PTN
PTN assembles PTN of PTN
PTN assembles PTN upon PTN
PTN assembles in PTN and PTN
PTN assembles of PTN PTN
PTN assembles of PTN and PTN
PTN assembles with PTN PTN
PTN associate in PTN
PTN associate through PTN
PTN associate with PTN
PTN associated PTN
PTN associated by PTN
PTN associated via PTN
PTN associates from PTN but PTN
PTN associates through PTN PTN
PTN associates with PTN but PTN
PTN associates within PTN
PTN associates within PTN PTN
PTN association between PTN
PTN association between PTN and PTN
PTN association between PTN or PTN
PTN association within PTN
PTN augments PTN PTN
PTN binding with PTN but PTN
PTN binding with PTN or PTN
PTN binds PTN and PTN
PTN binds PTN from PTN
PTN binds PTN in PTN
PTN binds to PTN
PTN blocked PTN

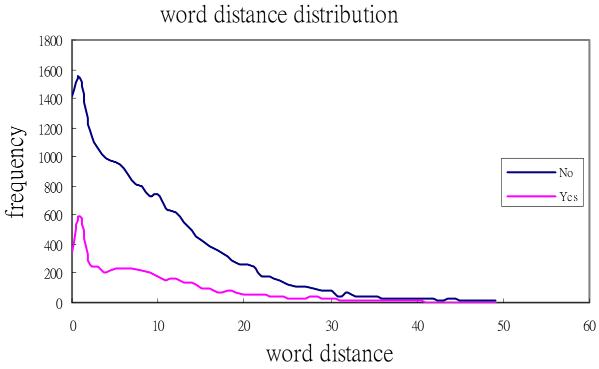
PTN blocked PTN PTN
PTN blocks PTN
PTN catalyze PTN
PTN catalyze PTN PTN
PTN catalyze of PTN
PTN co-localized by PTN
PTN co-localized of PTN
PTN complex of PTN
PTN complex on PTN
PTN conjugation by PTN
PTN conjugation in PTN
PTN conjugation on PTN
PTN conjugation with PTN or PTN
PTN degradation between PTN or PTN
PTN disrupts PTN and PTN
PTN encodes PTN through PTN
PTN encodes PTN within PTN
PTN enhanced PTN
PTN enhancement by PTN but PTN
PTN enhances PTN
PTN expressed of PTN
PTN expressed upon PTN
PTN form PTN PTN
PTN forms PTN
PTN induce PTN PTN
PTN induced PTN
PTN induced by PTN
PTN induced of PTN
PTN induces PTN between PTN
PTN induces PTN with PTN
PTN induction by PTN but PTN
PTN inhibit PTN PTN
PTN inhibits PTN
PTN inhibits PTN PTN
PTN interact with PTN
PTN interacted via PTN
PTN interacted with PTN
PTN interaction of PTN
PTN interaction with PTN
PTN interacts by PTN PTN
PTN interacts in PTN PTN
PTN interacts in PTN and PTN
PTN interacts with PTN
PTN interacts with PTN but PTN
PTN linked PTN
PTN localized upon PTN
PTN mediated of PTN
PTN mediates PTN
PTN modification in PTN
PTN modification of PTN and PTN
PTN modification within PTN
PTN modified PTN
PTN modified PTN PTN
PTN modified with PTN
PTN modifies PTN
PTN modifies PTN PTN
PTN modifies PTN by PTN
PTN modifies PTN with PTN
PTN modify PTN
PTN phosphorylated with PTN

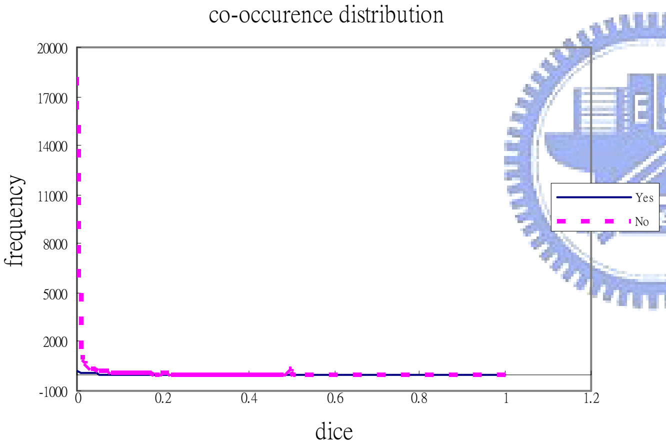
PTN phosphorylates PTN and PTN
PTN phosphorylates PTN but PTN
PTN phosphorylates PTN in PTN
PTN phosphorylates PTN on PTN
PTN prevents PTN but PTN
PTN promoted PTN
PTN promoted on PTN
PTN promoted with PTN
PTN promotes PTN
PTN promotes PTN PTN
PTN recognize PTN
PTN recognize PTN PTN
PTN recognizes PTN between PTN
PTN recognizes PTN through PTN
PTN reduces PTN PTN
PTN regulated PTN
PTN regulated by PTN
PTN regulated on PTN
PTN stimulate in PTN
PTN stimulate of PTN
PTN stimulated PTN
PTN stimulated upon PTN
PTN suppress PTN PTN
PTN ubiquitinated on PTN
PTN ubiquitinates PTN and PTN
PTN ubiquitinates PTN but PTN
PTN ubiquitinates PTN from PTN
PTN ubiquitinates PTN upon PTN
activation of PTN or PTN
activation of PTN to PTN
activations between PTN and PTN
activations of PTN and PTN
activations through PTN and PTN
association of PTN with PTN
association of PTN within PTN
association with PTN and PTN
association with PTN or PTN
associations from PTN and PTN
associations through PTN and PTN
binding of PTN through PTN
binding of PTN to PTN
binding of PTN via PTN
conjugation of PTN to PTN
degradation by PTN but PTN
interaction between PTN and PTN
interaction of PTN via PTN
interaction of PTN within PTN
interactions of PTN and PTN
interactions with PTN and PTN
modification from PTN but PTN
modification from PTN or PTN

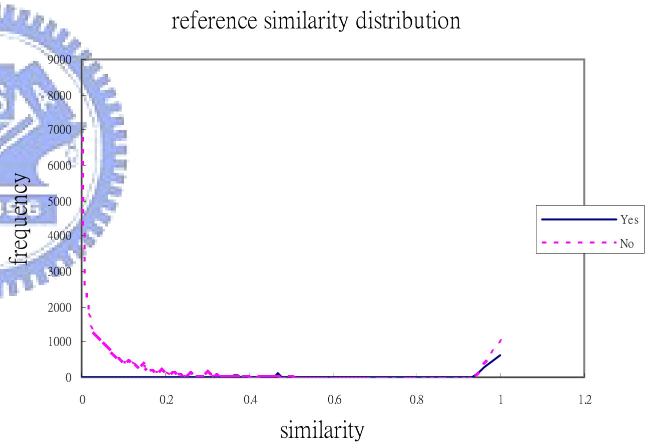
Appendix C

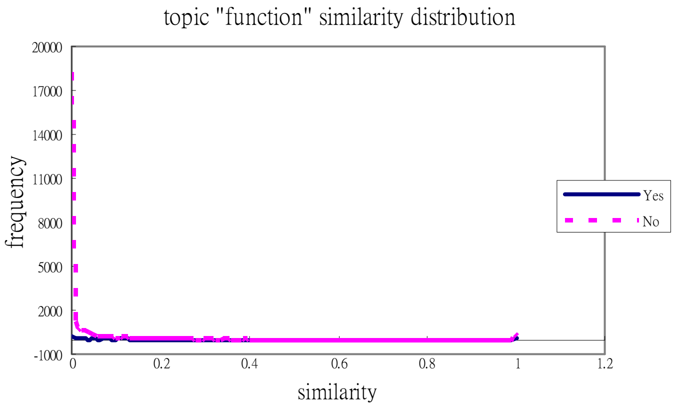
The distributions of all features with the classes: “Yes” and “No”

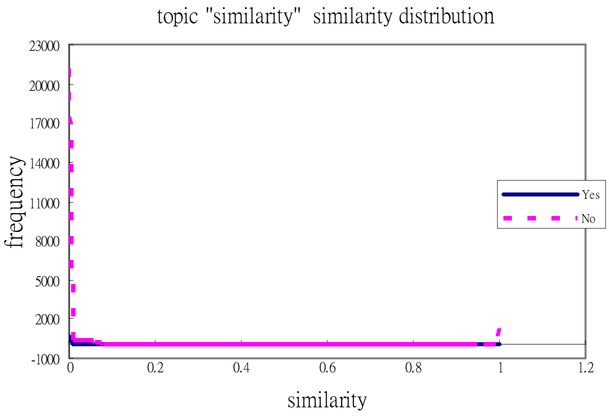


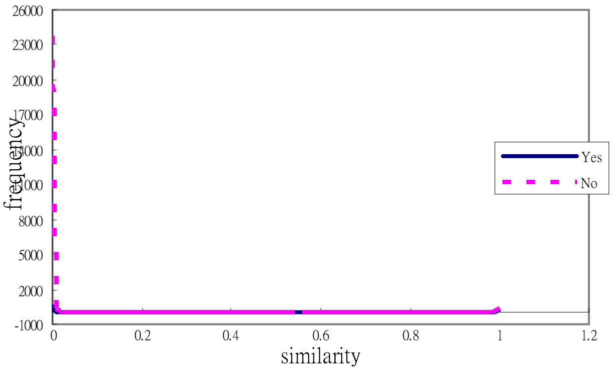


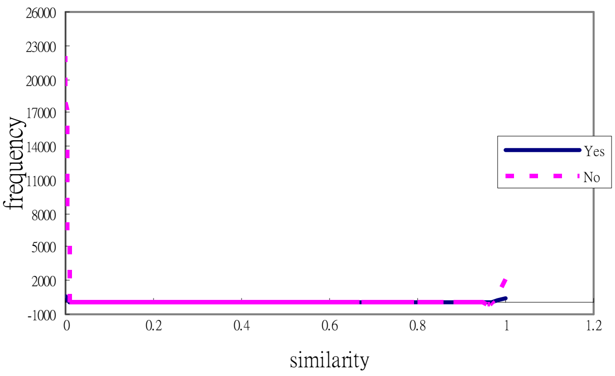


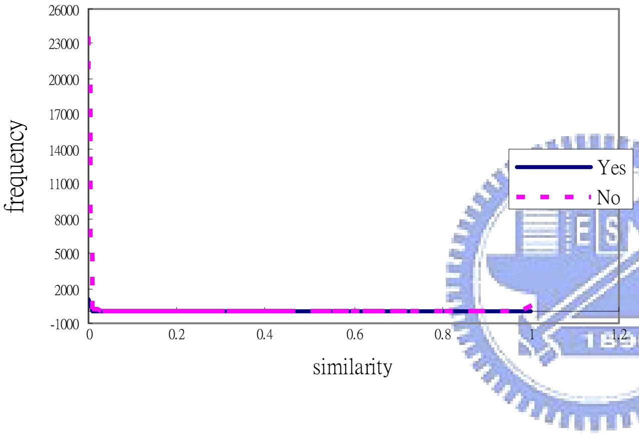







topic "subunit" similarity distribution


topic "subcellular location" similarity distribution


topic "catalytic activity" similarity distribution