Automatic Extraction of Proteins and Their Interactions from Biological Text*

Kiho Hong[1], Junhyung Park[2], Jihoon Yang[2], and Eunok Paek[3]

[1] IT Agent Research Lab, LSIS R&D Center
Hogae-dong, Dongsan-Gu, Anyang-Shi, Kyungki-Do 431-080, Korea
khhong1@lsis.biz
[2] Department of Computer Science and
Interdisciplinary Program of Integrated Biotechnology, Sogang University
1 Shinsoo-Dong, Mapo-Ku, Seoul 121-742, Korea
jhpark@mllab.sogang.ac.kr, yangjh@sogang.ac.kr
[3] Department of Mechanical and Information Engineering, The University of Seoul
90 Jeonnong-Dong, Dongdaemun-Gu, Seoul 130-743, Korea
paek@uos.ac.kr

Abstract. Text mining techniques have been proposed for extracting protein names and their interactions from biological text. First, we have made improvements on existing methods for handling single word protein names consisting of characters, special symbols, and numbers. Second, compound word protein names are also extracted using conditional probabilities of the occurrences of neighboring words. Third, interactions are extracted based on Bayes theorem over discriminating verbs that represent the interactions of proteins. Experimental results demonstrate the feasibility of our approach with improved performance in terms of accuracy and F-measure, requiring significantly less amount of computational time.

1 Introduction

In biologically significant applications such as developing a new drug and curing an inveterate disease, understanding the mutual effects of proteins (or genes which will be used interchangeably in the paper) are essential [1]. For instance, in order to develop a medicine for the breast cancer, we need to figure out the proteins related to the disease, and understand the mechanism how they work together in the course of the development of the breast cancer. In order to achieve the goal, extracting gene names must be proceeded. However, results by some of the existing methods leave much to be desired (e.g. extraction of multiple protein names, handling of negative and compound sentences and special characters). [2, 3]. Motivated by this background, we propose a new approach to extracting gene names and their relations. Section 2 and 3 describe the extraction of gene names and interactions between them. Section 4 shows experimental results in comparison with other approaches, followed by concluding remarks in Section 5.

* This work was supported by grant No. R01-2004-000-10689-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

2 Extraction of Protein Names

A Protein is named either as a single word (i.e. singular protein name (SPN)) or multiple words (i.e. multiple protein name (MPN)). We describe extracting methods for each case.

2.1 SPN Extraction

A SPN is extracted by two steps:

1. Word Class Tagging
   First, we used the Brill's tagger for tagging the text [4]. We added a word class GENE and prepared a list of the words in the class. GenBank[4] database is used for making the list. To define lexicon rules and context rules during the tagger's learning stage, we used GENIA CORPUS [2, 5].
2. SPN Extraction
   Generally, the protein names in biological literature are usually irregular and ambiguous. Even though there exist some rules for protein naming (some can be found at Nature Genetics site [3]), it is hard to apply the rules to existing protein names. Also as the rules are not generalized, some of the special characters are used frequently (e.g. hyphens, Greek letters, digits and Roman letters). In our lexicon, about 37% of these special characters are contained in the text. For this reason, processing them plays a great role for the whole efficiency. The HMM(Hidden Markov Model) with the Viterbi algorithm is applied for SPN extraction [6]. In addition to the algorithm, in order to handle the special characters, a substitution method was considered (e.g. & for digits and ? for roman letters). Substring matching was applied to the substituted protein names. However, there could be a collision in substring matching. For instance, 'gap1' will be substituted to 'gap&' which can be confused with 'gap' since 'gap' and 'gap&' has the same prototype 'gap'. Therefore, a set of words that can be confused in this fashion has been reserved as stopwords, which are ignored.

2.2 MPN Extraction

Usually an SPN makes up an MPN with near (or neighboring) words. However, an MPN not including an SPN should be considered as well (e.g. tumor necrosis factor). Based on the technique used in TagGeN [3], we developed an enhanced probability model. First, if GENE tag is included, the range of an MPN is determined by expanding words in bidirection (i.e. right and left). If an MPN does not include any GENE word, we use SEED word (e.g. the words appearing in MPNs frequently) for MPN determination. In our experiment, about 80 SEED words were used. To determine the range of an MPN, it is needed to expand the search from a GENE word or a SEED word, considering the following probability:

$$P(W_{next}|W_{current}, M_{current} = 1) \quad (1)$$

[4] http://www.ncbi.nlm.nih.gov/Genbank/index.html

where W_i represents a word occurring at position i, and M_i is binary value which represents whether the word at position i belongs to GENE word class or not. Some of the examples of W_i are illustrated in Table 1.

Table 1. Examples of W_i's Used in Probabilistic Models

Left Direction	
Set	*Example*
NN(Noun Class)	single-chain/**NN** fv/GENE
JJ(Adjective Class)	human/**JJ** GM-CSF/**GENE** gene/NN
CD(Number Class)	3/**CD** alpha/NN HSD/GENE
GENE(Gene Class)	human/JJ GM-CSF/**GENE** gene/NN
...ase	phospholip**ase**
Roman, Greek Character	type **II** IL-1R
Word Set(i.e. protein, gene, factor, etc.)	**protein** tyrosine kinase
Right Direction	
Set	*Example*
reporter	beta-globin **reporter**
product	start-1 gene **product**
single character	c-erb **A**
Numerals	IFN-stimulated gene factor **3**
...ed	C5a induc**ed** kappa-B
...like	Proximal c-jun TRE-**like** promoter element
...ing	IRF-1 GAS-bind**ing** complex

Initially, only M values of SEED words have 1 and all the others have 0. From the SEED word in the middle of the MPN, we move bidirectionally (i.e. to the right and to the left). By calculating the probability in (1), we calculate M values which represent whether the word is included in the MPN or not. Generally, the left-hand side words of an MPN have diverse word classes than the right ones, and the right-hand side words of an MPN consist of Greek letters, Roman letters and digits. This bidirectional expansion of words is expected to generate a more accurate model than that by TagGeN [3]. In order to make a probabilistic model for MPN extraction, we used 600 documents which are arbitrarily chosen from GENIA corpus and pre-tagged by domain experts. Figure 1 illustrates the probabilistic model used for tagging MPN from documents.

3 Extraction of Protein Interactions

This section describes the method for protein interactions. For example, there could be a pattern like *'Protein(A)-Type(interaction)-Protein(B)'* [4]. We define the verbs for the interactions and extract events from these predefined patterns.

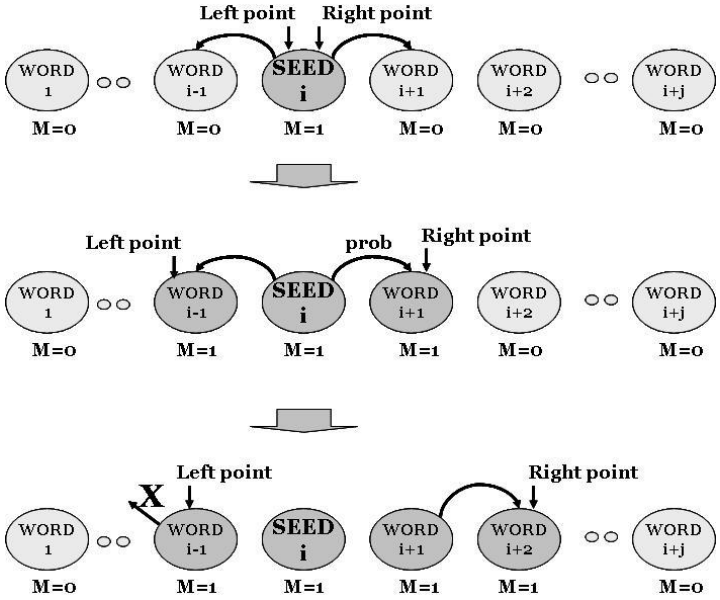

Fig. 1. Probabilistic Model for MPN Tagging

Then we are able to know that entity A has a relation with B. We first extract the discriminating verbs and then extract the associated protein interactions.

3.1 Discriminating Verb Extraction

A discriminating verb is extracted as follows:

1. Pre-processing
   The set of types (i.e. interactions) we are interested in would be the discriminating verb set. To define the set, pre-processing for extracting verbs from the text is needed. This can be done easily as Brill's tagger tags verbs as VB(verb, base form) including VBN(verb, past participle), and VBZ(verb, 3rd person singular present) that we can extract and stem.
2. P-Score Estimation
   We design a Bayesian probabilistic model for estimating the P-Score of each verb in the document. Then, we determine the set of discriminating verbs based on the P-Scores. The P-Score exhibits how well a verb describes the interaction between proteins. This was proposed for extracting a word set to classify documents by Marcotte [7]. We applied the method for extraction of discriminating verbs and calculate the following probability:

$$P(n|N,f) \approx e^{-Nf}\frac{(Nf)^n}{n!} \tag{2}$$

where n means how many times a verb is used as a protein interaction, N is total number of words in a document, and f is the total occurrences of each verb. The Poisson distribution can be an alternative for $P(n|N, f)$ while N is big enough and f is fairy small.

3. Discriminating Verb Selection
   Calculate the P-Score for every word, and then choose a set of arbitrary number of words with the highest P-Scores. 80 words (e.g. inhibit, indicate, etc.) were used in our experiment.

3.2 Protein Mutual Effect Extraction

To extract an interaction between genes from a sentence, there should be more than two gene names and one verb which describes their relation. However, due to an ambiguity of natural language, it is hard to recognize the structure well. We introduce a simple method to decrease the ambiguity of natural language structures. The steps of extracting protein interaction by using discriminating verbs and events are as follows:

1. Complex Sentence Processing
   To handle the ambiguity in a sentence, we used Toshihide Ono's method [1]. The method diminishes the ambiguity by converting a complex sentence into simple sentences and a negative sentence into a positive one.
2. Interaction Extraction
   If there is a pattern like *'Protein(A)-Type(Verb)- Protein(B)'* and a discriminating verb in a sentence, we calculate $Confidence$ of the sentence and then add the sentence into the *event* (protein interaction) set.

   The *Confidence* is calculated as follows:

$$Confidence = s + \frac{1}{sd} \tag{3}$$

where s is a binary value which represents whether the pattern is included in the sentence or not, and sd is sum of distances from proteins to a verb in the sentence. The distance is a number of words from a verb to proteins in a sentence. For example, 'IL-10 inhibits IFN-gamma-induced ICAM-1 expression in monocytes.' has distance 2 as *IL-10* and *inhibit* have distance 1 and *inhibit* and *IFN-gamma-induced ICAM-1 expression* have distance 1, too.

A sentence with no discriminating verb is also added to the candidate event set. We re-calculate $Confidence$ with $Frequency$ (how many times protein(A) and (B) are found in documents).

4 Experiments

We obtained the following extraction results of proteins and their interactions. Data used for the experiments are 600 papers from the GENIA Corpus. Our results are compared with those by ABGene and TagGeN [2, 3] in following tables.

– SPN Extraction
To observe the results while a data set size is changing, we experimented on 100 to 600 documents. Table 2 exhibits comparable accuracies among the approaches, with no conspicuous differences in performance for various sizes of data. Due to the substring matching method our system showed 2% low accuracy than that of ABGene, while it produced high recall and F-measure as shown in Table 3 and Table 4. In addition, our system was order of magnitude faster due to protein name hashing and simplified tagging process, as shown in Table 5.

Table 2. Accuracy of SPN Extraction

System \ Dataset	100	200	300	400	500	600	Average
Our system	83.28	85.17	84.97	85.10	85.58	85.88	85.00(%)
ABGene	87.40	87.12	87.13	87.19	86.12	87.10	87.01(%)
TagGeN	80.17	82.24	83.51	84.09	84.50	84.91	83.24(%)

Table 3. Recall of SPN Extraction

System \ Dataset	100	200	300	400	500	600	Average
Our system	95.06	95.99	96.39	97.00	96.89	96.33	96.27(%)
ABGene	50.15	57.75	49.16	54.12	60.02	61.12	55.22(%)
TagGeN	68.75	75.49	78.32	77.16	78.82	79.09	76.27(%)

Table 4. F-measure of SPN Extraction

System \ Dataset	100	200	300	400	500	600	Average
Our system	88.78	90.26	90.32	90.66	90.88	90.80	90.28(%)
ABGene	63.74	69.02	62.86	66.79	70.74	71.83	67.56(%)
TagGeN	74.02	78.72	80.83	80.48	81.56	81.90	79.56(%)

Table 5. Processing Time of SPN Extraction

System \ Dataset	100	200	300	400	500	600
Our system	2.81	3.50	4.23	4.85	5.46	6.23(sec)
ABGene	19.01	39.28	56.12	74.31	94.11	113.00(sec)
TagGeN	5913	11925	18777	24970	30979	36324(sec)

– MPN Extraction

'Exact' means the case every words in an MPN is extracted correctly. When some range of an MPN is partially extracted, it is named as 'Partial'. As shown in Table 6, our approach outperformed TagGeN in MPN extraction.

Table 6. Performance of MPN Extraction

	Recall(%)	Precision(%)	F-measure(%)
Our system(exact/partial)	84.25/91.56	86.65/91.35	84.84/91.84
TagGeN(exact/partial)	80.23/86.51	87.81/91.15	83.84/88.77

– Protein Interaction Extraction

We used 80 discriminating verbs in order of high P-Score. Selected 100 sentences including 14 negative, 8 compound sentence structures, and 121 protein interactions were used. From the sentences, we got 139 protein interactions. The number of interactions obtained by only discriminating verbs were 89, and 50 relations were added from the sentences in the candidate event set. We obtained F-measure over 80% as shown in Table 7.

Table 7. Performance of Interaction Extraction

Precision(%)	Recall(%)	F-measure(%)
76.58	92.70	83.87

5 Conclusion

We developed an extraction system for proteins and their interactions. Our protein name substring matching method and more abundant lexicon improved overall system performance. We also defined discriminating verbs and extracted them using a probabilistic model. We extracted 80 discriminating verbs by Poisson distribution. Finally, we defined events, and by their confidence values extracted their interactions. We observed improved performance in experiments with biological data.

Some of future research directions include: First, current simple substring matching method might cause low precision, which can be improved; Second, current algorithm includes ad hoc steps, and a more systematic algorithm for interaction extraction can be devised; Third, a thoughtful consideration for natural language processing is needed for more enhanced information extraction; Finally, more experiments with additional data will help verify our system.

References

1. Ono, T.: Automated extraction of information on protein-protein interactions from the biological literature. Bioinformatics **17** (2001) 155–161
2. J. D. Kim, T. Ohta, Y.T., Tsujii, J.: Genia corpus - a semantically annotated for bio-textmining. Bioinformatics **19** (2002) 180–192
3. Tanabe, L., Wilbur, W.J.: Tagging gene and protein names in full text article. In: Proceedings of Association for Computational Linguistics. (2004) 9–13
4. Brill, E.: Some advances in transformation-based part of speech tagging. In: AAAI. (1994)
5. Rinaldi, F.: Mining relations in the genia corpus. In: Proceedings of the Second European Workshop and Text mining for Bioinformatics. (2004)
6. Duda., R.O.: Pattern Classification. second edition edn. Wiley-interscience. Inc. (2000)
7. M.Marcotte, E.: Mining literature for protein-protein interactions. Bioinformatics **17** (2002) 359–363