구문관계에 기반한 유전자 상호작용 인식

김 미 영†

요 약

단백질이나 유전자들 간의 상호작용 인식은 생물학적 현상의 기술에 있어서 필수적이고, 이러한 상호작용의 네트웍 파악은 생물학 접근의 시작이라고 할 수 있다. 최근에, 대량의 생물학 관련 문서로부터 자연언어처리 기술을 사용하여 이러한 정보를 추출하려는 연구들이 많이 등장했다. 또한 이전 연구들은 언어학적 정보가 문서로부터 유전자 상호작용을 자동으로 추출하는 데 있어서 유용하다고 주장하고 있다. 하지만 기존의 방법들은 정확률에 비해 재현율이 많이 낮아서 성능이 그다지 좋지 못했다. 정확률의 감소 없이 재현율의 성능향상을 위해, 이 논문은 생물학관련 문서에서 구문관계에 기반하여 유전자 상호작용을 인식하는 방법을 제안한다. 생물학 도메인에 관련된 전문지식 없이, 우리의 방법은 단지 적은 양의 학습데이터를 사용하여 효과적인 성능을 보인다. LLL05(ICML05 Workshop on Learning Language in Logic)에서 제공한 데이터 포맷을 그대로 사용하여, 상호작용하는 두 유전자 중 작용의 주체가 되는 유전자를 에이전트라 하고 상호 작용의 대상이 되는 유전자를 타겟이라 한다. 본 논문에서 제안하는 첫 단계에서, 에이전트와 타겟 유전자에 대한 유전자-전이 구문관계를 인식한다. 두 번째 단계에서, 유전자 간의 상호작용이 있음을 암시하는 용언리스트를 구축한다. 마지막 단계에서, 상호작용하는 것으로 인식된 두 유전자 중 어느 것이 에이전트이고 타겟인지를 판단하기 위해 구문관계의 방향 정보를 학습한다. LLL05 데이터를 사용한 실험결과에서, 본 논문에서 제안한 방법이 학습 데이터에 대해서는 88%의 F-measure 성능을 보였고, 테스트 데이터에 대해서는 70.4%의 F-measure 성능을 보였다. 이 결과는 기존의 방법들보다 훨씬 더 좋은 성능이다. 우리는 성능에 대한 각 단계의 공헌도를 실험하여, 첫 단계는 재현율 향상에 기여를 하고 두 번째와 세 번째 단계는 정확률 향상에 기여했음을 보인다.

키워드 : 유전자 상호작용, 바이오인포매틱스, 구문 관계, 정보 추출, 텍스트 마이닝

Detection of Gene Interactions based on Syntactic Relations

Mi-Young Kim†

ABSTRACT

Interactions between proteins and genes are often considered essential in the description of biomolecular phenomena, and networks of interactions are considered as an entre for a Systems Biology approach. Recently, many works try to extract information by analyzing biomolecular text using natural language processing technology. Previous researches insist that linguistic information is useful to improve the performance in detecting gene interactions. However, previous systems do not show reasonable performance because of low recall. To improve recall without sacrificing precision, this paper proposes a new method for detection of gene interactions based on syntactic relations. Without biomolecular knowledge, our method shows reasonable performance using only small size of training data. Using the format of *LLL05(ICML05 Workshop on Learning Language in Logic)* data, we detect the agent gene and its target gene that interact with each other. In the 1st phase, we detect encapsulation types for each agent and target candidate. In the 2nd phase, we construct verb lists that indicate the interaction information between two genes. In the last phase, to detect which of two genes is an agent or a target, we learn direction information. In the experimental results using LLL05 data, our proposed method showed F-measure of 88% for training data, and 70.4% for test data. This performance significantly outperformed previous methods. We also describe the contribution rate of each phase to the performance, and demonstrate that the first phase contributes to the improvement of recall and the second and last phases contribute to the improvement of precision.

Key Words : Gene Interaction, Bioinformatics, Syntactic Relation, Information Extraction

1. 서 론

생물학 관련 문서로부터의 정보 추출은 최근 중요하게 부각되고 있는 분야이다. 대량의 문서데이터로부터 주로 단백질 및 유전자의 상호작용정보, 단백질의 세포 내 위치나 단백질명의 자동 인식과 관련하여 정보를 추출하는 데 초점을 두고 있고, 단순한 자질들을 사용한 기계학습 방법보다는 자연언어처리 방법을 이용하여 언어학적 정보를 사용한 정보추출 방법들이 최근 많이 소개되고 있다. 또한 이전 연구

※ 이 논문은 2007년도 성신여자대학교 학술연구조성비 지원에 의하여 연구되었음.
† 정 회 원 : 성신여자대학교 컴퓨터정보학부 전임강사
논문접수 : 2007년 4월 30일, 심사완료 : 2007년 8월 20일

들은 언어학적 정보가 문서로부터 유전자 상호작용을 자동으로 추출하는 데 있어서 유용하다고 주장하고 있다. 하지만 실제 언어학적 정보가 표기된 생물학 관련 학습데이터가 많이 제공되지 않고 있고, 다만 LLL05(ICML05 Workshop on Learning Language in Logic)에서 품사정보와 구문관계 정보를 표시한 소량의 데이터를 제공하고 있다. LLL05는 상호작용하는 유전자 쌍의 자동 추출을 위하여 학습데이터와 실험데이터를 제공하여 실험결과를 비교하는 대회로서, 각각의 방법에 대한 객관적인 실험결과 비교가 가능하나, 학습데이터가 적어서 이 데이터를 대상으로 한 논문들의 성능은 그다지 좋지 않았다. 따라서 본 논문은 소규모 데이터의 학습에도 효과적일 수 있도록, 구문관계에 기반하여 유전자 상호작용을 자동으로 인식하는 3단계 방법을 제안한다. 실험데이터로는 LLL05의 데이터를 그대로 사용하여, 기존 결과들과 객관적인 비교를 해본다. 본 논문은 다음과 같이 구성되어 있다. 2장에서는 유전자 상호작용에 관련된 기존 연구를 설명한다. 3장에서는 유전자 상호작용을 위한 3단계를 상세히 설명한다. 4장에서 실험에 사용되는 데이터에 대한 설명을 하고, 본 논문에서 제안한 3단계 방법이 유전자 상호작용을 결정하는 데 효과적임을 실험결과를 통해 보인다. 마지막으로 결론이 이어진다.

2. 기존연구

단백질이나 유전자들 간의 상호작용 인식은 생물학적 현상의 기술에 있어서 필수적이고, 이러한 상호작용의 네트웍 파악은 생물학 접근의 시작이라고 할 수 있다[17]. 따라서 텍스트로부터 유전자의 상호작용 정보를 자동추출하는 문제와 관련하여 많은 연구가 있어왔다. 현재 연구는 단백질-단백질 상호작용[5, 6], 세포 내 위치[7], 질병 위치[8]와 그 외 특정 다른 타입들을 인식하는 것을 하고 있다. 현재, 연속적 모델링에 기반한 시스템들과 패턴과 규칙에 기반한 추출방법이 단백질-단백질 상호작용 정보를 추출하는 데 가장 좋은 성능을 보이고 있다[5, 9, 10].

단백질/유전자 상호작용을 위하여, 많은 연구들이 언어학적 정보를 사용하여 성능을 높이고자 하였다. 정확률을 희생하지 않고 재현율을 높이기 위해, Otasek[1]은 더 많은 학습을 통하여 파서에 의해 인식가능한 문장 구조를 다양하게 확장했다. 하지만 그들이 사용한 파서 정보는 문장이 상호작용 정보를 담고 있는지 아닌지를 판단하는 데만 이용되는데 그쳤다. Park[2]에서 또한 재현율을 높이기 위해 구문분석 정보를 사용할 것을 제안하고 있다. 실제 양방향 점진적 파싱을 사용한 방법에서 좋은 성능을 보였고, 그 외에도 대등접속문의 동격처리, 복합명사, 긍정/부정 서술어 학습 등 많은 언어학적 처리를 수행했다. 하지만 두 유전자가 상호작용을 나타내는 동사와 구문관계를 가질 때, 동격과 대등접속 이외에도 다양한 구문관계를 거쳐서 구문관계를 가질 수 있으므로, 특정 구문에 대한 전처리에만 의존하는 방법으로는 재현율을 크게 향상시킬 수 없다.

LLL05 실험 데이터를 이용한 논문들은 다음과 같다. J. Hakenberg 등[4]은 문장정렬과 유한상태 오토마타를 사용하여 실험하였다. 그들은 LLL05데이터에서 제공한 언어학적 정보를 사용하지 않았다. 이 실험에서는 상호작용하는 두 유전자를 제대로 찾은 후 두 개의 유전자 사이에 에이전트와 타겟을 반대로 인식하는 경우가 많았다. 언어학적 정보는 이러한 타입의 오류를 수정하는 데 필요할 것이다.

Greenwood[11]는 MINIPAR[16]의 의존구문트리에 기반하여 패턴을 추출하였다. 그들은 에이전트와 타겟 유전자들의 구문트리에서의 위치를 파악하여, 모든 가능한 구문상의 위치를 학습데이터로부터 추출하여 패턴으로 구축하였다. 실제학습은 LLL05 데이터 중 대용어참조가 없는 학습데이터를 대상으로 하였고, 테스트 성능은 14.8%로 좋지 않았다. 학습데이터의 양이 적기 때문에 실험데이터에서는 학습데이터와 다른 구문위치에 에이전트와 타겟이 충분히 존재할 수 있으므로, 이러한 단순한 구문트리에서의 위치 정보 추출은 성능향상에 기여하지 못했다. 또한, 성능이 안 좋은 이유 중 하나로 MINIPAR 구문분석기의 의존트리 결과 오류를 들 수 있다.

Goadrich[12]는 Brill 태거와 CRF(Conditional Random Fields) 기반의 단문분석, 그리고 스테머(stemmer)를 사용했다. 구문 구조에 의존하지 않고 에이전트에 가장 가까운 단백질/유전자를 단순히 타겟으로 결정함으로써 문제를 일으키는 경우가 많았다. Popelinsky[13]는 간단한 예제를 대상으로 한 학습과 모든 예제를 대상으로 한 학습을 분리하여 2단계 방법을 제안했고, 언어학적 정보를 위해 Brill 태거와 워드넷을 사용하였다.

LLL05 데이터를 사용한 실험 중 가장 좋은 성능을 보인 것은 S. Riedel과 E. Klein[15]이었고, 그들은 구문체인을 사용하였다. 에이전트와 타겟 유전자 사이에 구문관계의 체인이 있다고 가정하고 두 개의 유전자 사이에 구문체인에 의존하여 절의 셋을 만들었다. 이 실험은 52.6%의 f-measure 성능을 보이면서, 구문정보를 사용하면 성능이 급격하게 향상된다고 주장했다. 그들은 에이전트와 타겟 간의 구문체인에 대한 조건을 규칙으로 생성하여 사용했다. 실제 에이전트와 타겟이 구문체인으로 연결되기 위한 구문조건을 파악한 결과 정확률 향상에 크게 기여를 하였으나, 학습데이터가 작았던 이유로 재현율을 높이기 위한 방법은 좀 더 연구될 필요가 있다.

LLL05 데이터를 이용하지 않고 GENIA와 ATCR 데이터를 이용한 연구로서, Rinaldi [18] 또한 언어학적 접근을 시도하였다. 그들은 상호작용동사와 주어 또는 목적어 구문관계로 연결된 에이전트와 타겟을 찾는다. 따라서, 단지 주어와 목적어 구문관계만을 고려하고 있고, 두 유전자가 상호작용동사와 직접적인 구문관계로 연결된 경우만을 인식한다는 단점이 있다.

기존의 논문들은 상호작용하는 유전자 쌍의 인식을 위해 구문정보가 도움이 된다는 공통된 정보를 제공하고 있으나, 구문관계의 일부만을 사용하거나 상호작용동사와 직접적인

관련이 있는 구문관계만을 고려하고 있고, 학습데이터에서 두 유전자가 등장한 구문패턴의 위치 자체를 그대로 규칙으로 사용하여 재현율 향상에는 크게 기여를 못하고 있다.

따라서, 기존의 논문들의 주장에 의거하여 상호작용하는 유전자 쌍을 인식하는 데 구문정보가 중요한 키가 된다는 것은 가정하에, 우리는 상호작용하는 두 유전자 사이의 구문관계 사슬 정보를 충실히 이용하면서 정확률과 재현율을 둘 다 높일 수 있는 3단계 방법을 제안한다.

상호작용하는 유전자 쌍을 알기 위해서는 상호작용을 나타내는 동사를 먼저 인식해야 한다. 상호작용을 나타내는 동사를 이 논문에서 '상호작용동사'라고 하겠다. 하지만 일반적으로 상호작용동사가 유전자 쌍과 직접적으로 구문관계를 가지고 연결되어 있지는 않다. 실제 상호작용동사와 직접적인 구문관계가 있는 용어는 따로 존재하고, 이 용어 노드로까지 유전자 노드가 여러 구문관계를 통하여 전이되어야 한다. 이와 같이 유전자 노드의 전이를 발생시키는 구문관계를 유전자-전이 구문관계라 부르겠다. 이 유전자-전이 구문관계는 재현율 향상에 기여하는 주된 요소가 된다.

상호작용하는 유전자 쌍 인식을 위해 본 논문에서 제안하는 단계는 다음과 같다. 첫 번째 단계로 상호작용동사와 직접적인 구문관계를 가지는 노드로까지 유전자 노드를 전이시키는 구문관계(유전자-전이 구문관계) 리스트를 추출하고, 두 번째 단계로 상호작용을 나타내는 동사리스트를 추출한다. 위의 두 단계를 거치면 상호작용하는 유전자 쌍의 인식이 가능하다. 마지막 단계로 상호작용하는 유전자 쌍에서 에이전트와 타겟이 각각 무엇인지 결정하기 위해 구문관계의 방향성을 학습한다. 이 3단계를 3장에서 자세히 설명하도록 한다.

3. 유전자 상호작용 인식 과정

실험에 사용하는 LLL05 데이터에 대해 먼저 설명하도록 한다. LLL05 데이터는 고초균(Bacillus subtilis) 관련 문장 내에서 유전자 상호작용의 정보추출에 초점을 맞추고 있다. LLL05 학습데이터의 예제는 (그림 1)과 같고, (그림 1)의 각 정보에 대한 상세한 설명은 (그림 2)에 나와 있다.

(그림 1)에서는 'In this mutant, expression of the spoIIG gene, whose transcription depends on both sigma(A) and the phosphorylated Spo0A protein, Spo0A~P, a major transcription factor during early stages of sporulation, was greatly reduced at 43 degrees C.' 라는 문장에 대한 단어, 구문, 상호작용 정보를 표기한 것이다.

LLL05의 학습 데이터셋은 크게 두 종류로 나뉜다. 첫 번째 학습 데이터셋은 대용어나 생략을 포함하지 않은 문장으로 구성되어 있고, 두 번째 학습 데이터셋은 대용어나 생략 등이 포함되어 있는 문장들이다. 실제 LLL05에서 제공하는 구문정보는 대용어나 생략 등의 정보는 알려주지 않으므로, 첫 번째 학습데이터만을 대상으로 실험한다. 이 학습데이터 내에는 유전자 상호작용 정보가 106개 들어 있다. 즉,

```
ID          11064201-3
sentence    In this mutant, expression of the spoIIG gene, whose tran-
            scription depends on both sigma(A) and the phosphory-
            lated Spo0A protein, Spo0A~P, a major transcription
            factor during early stages of sporulation, was greatly
            reduced at 43 degrees C.

words
word(0,'In',0,1) word(1,'this',3,6) word(2,'mutant',8,13)
word(3,'expression',16,25) word(4,'of',27,28) word(5,'the',30,32)
word(6,'spoIIG',34,39) word(7,'gene',41,44) word(8,'whose',47,51)
word(9,'transcription',53,65) word(10,'depends',67,73)
word(11,'on',75,76) word(12,'both',78,81) word(13,'sigma(A)',83,90)
word(14,'and',92,94) word(15,'the',96,98)
word(16,'phosphorylated',100,113) word(17,'Spo0A',115,119)
word(18,'protein',121,127) word(19,'Spo0A~P',130,136)
word(20,'a',139,139) word(21,'major',141,145)
word(22,'transcription',147,159) word(23,'factor',161,166)
word(24, 'during',168,173) word(25,'early',175,179)
word(26,'stages',181,186) word(27,'of',188,189)
word(28,'sporulation',191,201) word(29,'was',204,206)
word(30,'greatly',208,214) word(31,'reduced',216,222)
word(32,'at',224,225) word(33,'43',227,228) word(34,'de-
            grees',230,236)
word(35,'C',238,238)

lemmas
lemma(0,'in') lemma(1,'this') lemma(2,'mutant') lemma(3,'ex-
            pression')
lemma(4,'of') lemma(5,'the') lemma(6,'spoIIG') lemma(7,'gene')
lemma(8,'whose') lemma(9,'transcription') lemma(10,'depend')
lemma(11,'on') lemma(12,'both') lemma(13,'sigA') lemma(14,'and')
lemma(15,'the') lemma(16,'phosphorylated') lemma(17,'spo0A')
lemma(18,'protein') lemma(19,'Spo0A~P') lemma(20,'a')
lemma(21,'major') lemma(22,'transcription') lemma(23,'factor')
lemma(24,'during') lemma(25,'early') lemma(26,'stage') lem-
            ma(27,'of')
lemma(28,'sporulation') lemma(29,'be') lemma(30,'greatly')
lemma(31,'reduce') lemma(32,'at') lemma(33,'43') lemma(34,'de-
            gree')
lemma(35,'C')

syntactic_relations
relation('subj:V_PASS-N',31,3) relation('mod_att:N-N',7,6)
relation('mod_att:N-ADJ',34,33) relation('comp_during:N-N',23,26)
relation('comp_of:N-N',26,28) relation('comp_on:V-N',10,13)
relation('mod_att:N-N',23,22) relation('mod_att:N-ADJ',18,16)
relation('mod:V_PASS-ADV',31,30) rela-
            tion('mod_att:N-ADJ',26,25)
relation('mod_att:N-ADJ',23,21) relation('mod_att:N-N',18,17)
relation('comp_on:V-N',10,18) relation('appos',19,23)
relation('subj:V-N',10,9) relation('appos',18,19)
relation('comp_of:N-N',3,7) relation('comp_in:V-N',31,2)
relation('comp_of:N-N',9,7) relation('comp_at:V_PASS-N',31,34)
relation('mod_att:N-N',34,35)

agents   agent(13) agent(17)
targets  target(6)
genic_interactions   genic_interaction(13,6) genic_interaction(17,6)
```

(그림 1) LLL05 학습 데이터의 예

70개의 행동에 관련된 예제와 30개의 바인딩과 촉매에 관련된 예제, 6개의 regulon에 관련된 예제가 있다. 구문 관계는 텍스트의 구조를 인식하는 데 중요한 언어학적 정보로서, (그림 3)은 상호작용하는 두 유전자 사이에 존재하는 구문관계 사슬의 하나의 예를 보여준다. LLL05에서 제공된 구문관계는 relation(rel_i, w_i, w_j)의 형태다. 여기서 rel_i 는 LLL 구문분석기에 의해 할당된 w_i 와 w_j 사이의 구문카테고리의

ID : 각 문장의 유일한 아이디
sentence : 원래 문장
words : 문장 내의 단어들
표기법- word(단어id,단어어휘,단어시작위치,단어끝위치)

lemmas : 문장 내 단어들의 표준형(어근) 리스트
(동사의 기본형, 명사, 대명사, 대용어 등의 단수형 등)
표기법- lemma(단어 id, 단어의 기본형)

syntactic_relations : 단어들의 구문관계 표시
표기법- relation(구문카테고리, 지배소단어의 아이디, 의존소단어의 아이디)

agents : 문장 내에 존재하는 상호작용 유전자 쌍 중 에이전트 모음
표기법 : agent(단어 id)
targets : 문장 내에 존재하는 상호작용 유전자 쌍 중 타겟 모음
표기법 : target(단어 id)
genic_interactions : 서로 작용하는 유전자 쌍
표기법 : genic_interaction(에이전트가 되는 단어 id, 타겟이 되는 단어 id)

자세한 것은 http://genome.jouy.inra.fr/texte/LLLchallenge/ 참조바람.

(그림 2) LLL05 학습데이터의 설명

고정된 셋 중의 하나이다. 우리는 이러한 구문관계 정보를 이용하여 상호작용하는 유전자 쌍을 인식하고자 한다. 문장 내에서 상호작용동사가 에이전트와 타겟을 연결하는 기능을 한다고 하면, 실제 상호작용동사는 에이전트와 타겟에 대해 구문관계를 가질 것이고, [에이전트] → (상호작용동사) → [타겟] 과 같이 구문관계 사슬을 간단히 나타낼 수 있다. 여기서 [에이전트]는 실제 에이전트에 해당하는 유전자 그대로일 수도 있고, 이것의 동격형이나 다른 명사의 수식어가 된 형태 등으로 다른 용어로의 전이가 일어날 수도 있다. [타겟] 또한 마찬가지다. (그림 3)의 예를 보자. (그림 3)에서 상호작용하는 유전자쌍 (에이전트, 타겟) = (Spo0A, spoIIG)이다. Spo0A로부터 spoIIG까지의 구문사슬을 따라가보면, 에이전트와 타겟을 매개해 주는 동사는 'depend(V)'이다. 하지만 실제 'depend(V)'는 에이전트(Spo0A)와 타겟(spoIIG)들과는 직접적인 구문관계가 있지 않고, protein(N), transcription(N)과 직접 구문관계를 맺고 있다. 즉, protein(N)은 Spo0A가 여러 구문관계를 거쳐 전이되어 포장된 형태라고 볼 수 있다. Spo0A가 protein(N)으로 전이될 때 사용된 구문관계는 mod_att:N-N이다. 타겟인 spoIIG 또한 여러 구문관계를 거쳐 전이되면서 transcription(N)으로 포장되어 depend(V)와 구문관계를 맺고 있다고 볼 수 있다.

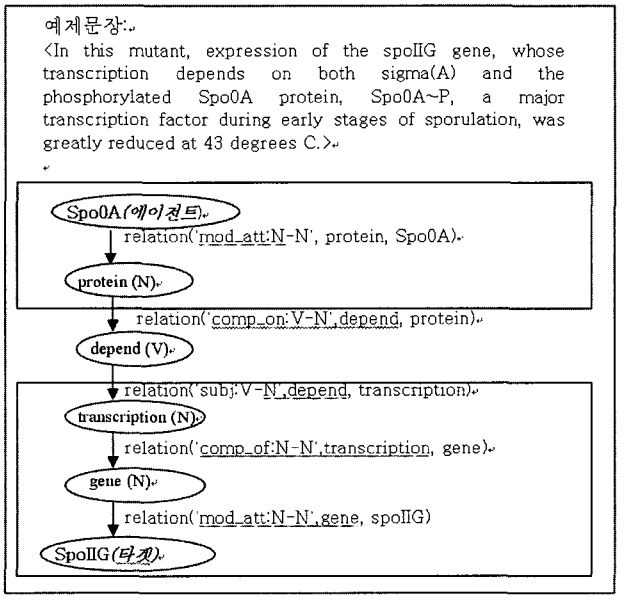
따라서 우리는 첫 번째 단계로 상호작용동사 노드와 직접적인 관계를 맺고 있는 노드로까지 에이전트와 타겟 노드가 전이되도록 하는 구문관계(유전자-전이 구문관계)의 타입을 정하고, 두 번째 단계로 에이전트와 타겟을 매개해 주는 동사들을 추출한다. 이 두 단계를 거치면 상호작용하는 유전자 쌍을 인식할 수는 있겠으나, 두 유전자 중 어느 것이 에이전트이고 타겟인지를 알 수 있는 정보는 없다. 따라서, 에이전트부터 타겟까지 연결된 구문사슬의 방향을 학습하는 세 번째 단계가 필요하다. 상세한 3단계를 아래에서 설명한다.

3.1 단계 1: 에이전트와 타겟 후보를 전이시키는 유전자-전이 구문관계 추출

먼저 구문관계를 바탕으로 하여 에이전트로부터 타겟에까지 이르는 가장 짧은 구문사슬을 만든다. 구문사슬 내에는 에이전트와 타겟을 연결하는 동사가 그 두 개의 유전자 사이에 상호작용을 나타내면서 존재하게 된다.

에이전트로부터 타겟까지의 구문사슬에서, 상호작용동사 바로 앞에서 상호작용동사와 직접 구문관계를 맺으며 에이전트를 포장하는 노드를 '메타에이전트'라고 부르고, 상호작용동사 뒤에서 직접 구문관계를 맺으며 타겟을 포장하고 있는 노드를 '메타타겟'이라 부르도록 한다. (그림 3)에서 protein(N)은 메타에이전트이고 transcription(N)은 메타타겟이다. 이 단계에서는 에이전트로부터 메타에이전트까지, 타겟으로부터 메타타겟까지 전이를 돕는 구문관계(유전자-전이 구문관계) 리스트를 추출하는 것이 목적이다.

실제 학습데이터로부터 형성된 구문사슬에서 상호작용동사가 어떤 것인지 자동적으로 알 수가 없으므로, 단지 하나의 동사가 포함된 구문사슬만을 대상으로 하여 그 동사를



(그림 3) 구문관계 사슬의 예

-mod_att:N-N
-mod_pred:N-N
-comp_of:N-N
-comp_in:N-N
-mod:ADJ-N
-mod_att:N-ADJ
-mod_att:N-ADJ
-appos
-comp_within:N-N
-comp_from:N-N
-comp_by:V_PASS-N
-comp_at:V-N

(그림 4) 유전자-전이 구문관계

상호 작용동사라 가정하고 유전자-전이 구문관계를 수집했다. (그림 4)는 획득된 유전자-전이 구문관계를 보여준다.

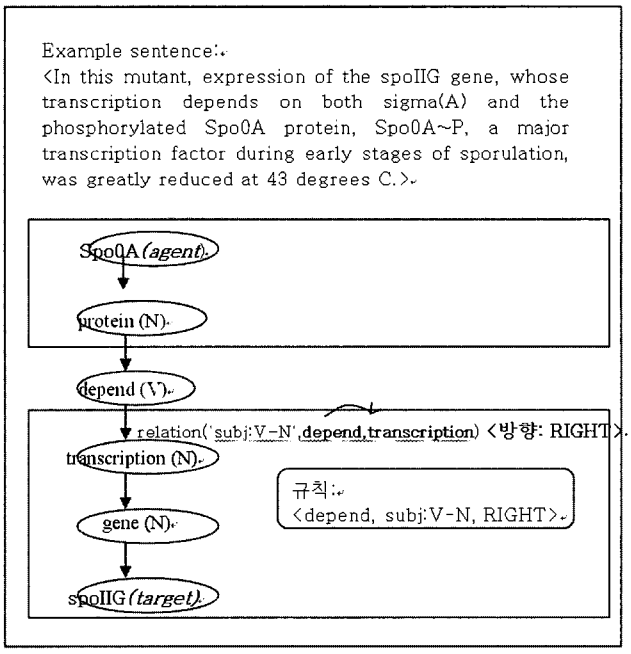
3.2 단계 2: 에이전트와 타겟을 연계하는 상호작용동사 리스트 추출

실제 학습데이터에서 유전자 상호작용을 나타내는 상호작용동사들을 자동으로 추출한다. 이 동사리스트는 에이전트에서 타겟에 이르는 가장 짧은 구문사슬 내에 존재하는 모든 동사들을 추출한 것이다. 그리하여 67개의 동사가 추출되었다.

3.3 단계 3: 상호작용하는 두 유전자로부터 에이전트와 타겟을 각각 인식하기 위한 구문사슬의 방향 정보 학습

이전 연구에서, 상호작용하는 양쪽 유전자 중 어느 것이 에이전트이고 타겟인지를 인식하는 과정 동안 많은 오류가 발생했다. 에이전트와 타겟의 잘못된 인식은 낮은 정확률을 초래한다. 따라서, 두 개의 유전자 사이에 에이전트와 타겟을 정확히 인식하는 과정이 요구된다. 인식을 수행하기 위하여, 에이전트로부터 타겟까지의 구문사슬에서 구문관계의 방향을 학습한다. 만약 우리가 반대 방향을 허용하지 않으면, 에이전트와 타겟은 반대로 인식되지 않을 것이므로 정확률을 높일 수 있다.

구문관계에 대하여 방향은 RIGHT, LEFT, ANY 등 세 가지 태그가 주어지며, 각각은 다음과 같이 정의된다.



(그림 5) 그림3의 구문사슬 중 상호작용동사에 대한 방향정보

1. RIGHT는 구문관계가 *relation(구문카테고리, 현재노드, 다음노드)*일 때 주어지는 방향태그로서, 다음노드가 현재노드의 오른쪽에 적혀 있음을 의미한다.
2. LEFT는 구문관계가 *relation(구문카테고리, 다음노드, 현재노드)*일 때 주어지는 방향태그로서, 다음 노드가 현재노드의 왼쪽에 적혀 있음을 의미한다.
3. ANY는 RIGHT나 LEFT 어느 방향도 허용한다는 태그로서, 실제 방향규칙을 획득할 때 사용되는 태그다.

에이전트 노드로부터 타겟 노드 방향으로 이어지는 구문사슬의 방향정보를 얻기 위하여, 구문 사슬의 노드들 중 2단계에서 얻었던 상호작용동사 노드를 기준으로 구문카테고리, 현재 노드의 어휘, 방향태그를 습득하여 <어휘, 구문카테고리, 방향> 규칙 셋을 구축한다. (그림 3)의 구문사슬 중 상호작용동사 노드에 관련된 <어휘, 구문카테고리, 방향> 규칙이 (그림 5)에 나타나 있다. 위의 규칙 셋을 바탕으로 하여 방향규칙이 학습된다. 방향규칙 학습 알고리즘은 (그림 6)에 나와 있고, 이를 설명하면 다음과 같다.

두 유전자 중 어느 것이 에이전트이고 타겟인지를 정확히 인식하기 위하여 두 종류의 규칙 셋을 구축한다. 하나는 에이전트로부터 시작하여 타겟까지의 구문사슬의 방향을 학습함으로써 얻어지는 긍정규칙 셋이고, 다른 하나는 반대방향으로 타겟으로부터 에이전트로의 구문사슬의 방향을 학습함으로써 얻어진 부정규칙 셋이다.

(그림 7)은 (그림 3)의 문장을 대상으로 타겟부터 시작하여 에이전트까지의 역방향 구문사슬을 보여주고 있다. (그림 3)의 문장에 대한 긍정규칙과 부정규칙은 <표 1>에 나와 있다.

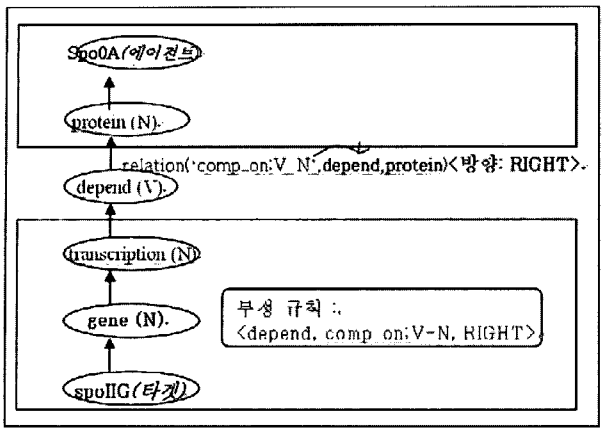
1> 긍정규칙 셋 정렬
(1.1) 에이전트로부터 타겟에 이르는 가장 짧은 구문사슬로부터 <어휘 A, 관계 B, 방향 C> 자료를 추출하여 긍정규칙 셋을 만든다.
(1.2) 임의의 어휘 A와 구문카테고리 B에 대해, 만약 긍정규칙 셋 내에 <A, B, RIGHT> 와 <A, B, LEFT>가 둘 다 존재하면, 두 규칙을 삭제하고 새로운 규칙 <A, B, ANY>를 추가한다.
2> 부정규칙 셋 정렬
(2.1) 타겟으로부터 에이전트에 이르는 가장 짧은 구문사슬로부터 <어휘 A, 관계 B, 방향 C> 자료를 추출하여 부정규칙 셋을 만든다.
(2.2) 임의의 어휘 A와 구문카테고리 B에 대해, 만약 부정규칙 셋 내에 <A, B, RIGHT>와 <A, B, LEFT>가 둘 다 존재하면, 두 규칙을 삭제하고 새로운 규칙 <A, B, ANY>를 추가한다.
3> 방향규칙 구축
긍정규칙 셋의 각 규칙 <A, B, C>에 대해
(3.1) 만약 부정규칙 셋에 <A, B, C>가 존재한다면, **방향규칙 <A, B, ANY>를 얻는다.**
(3.2) 만약 부정규칙 셋에 <A, B, C의 역방향>이 존재한다면, **방향규칙 <A, B, C>를 얻는다.**
(3.3) 그 외의 경우, 즉 부정규칙 셋에 어휘 A, 구문카테고리 B에 대한 정보가 없으면, **방향 규칙 <A, B, C>를 얻는다.**

방향 규칙이 더 이상 생성되지 않을 때까지 위의 과정을 반복한다.

(그림 6) 방향규칙 구축 알고리즘

<표 1> (그림 3)의 문장에 대한 긍정규칙과 부정규칙

긍정규칙	<depend, subj:V-N, RIGHT>
부정규칙	<depend, comp_on:V-N, RIGHT>



(그림 7) 부정규칙 구축을 위한 (그림 3)의 역방향 구문사슬

방향규칙을 얻기 위해, 우선 긍정규칙과 부정규칙 셋을 정렬하여 모순되는 정보들을 제거한다. 그 후, 두 규칙으로부터 방향규칙을 최종적으로 얻는다. 위 두 단계를 아래의 장에서 각각 설명하도록 한다.

3.3.1 긍정/부정 규칙 셋의 정렬

첫 번째로, 긍정규칙과 부정규칙 각각으로부터 모순된 정보들을 찾아내어 제거함으로써 각 규칙 셋을 정렬한다.

임의의 어휘 A와 구문카테고리 B에 대해, 만약 긍정규칙 셋 내에 <A, B, RIGHT>와 <A,B,LEFT>가 모두 존재한다면, 서로 다른 방향정보가 존재하여 신뢰성이 떨어지므로 두 개의 규칙을 삭제하고 수정된 하나의 새로운 규칙 <A, B, ANY>를 추가한다. 어느 방향도 허용한다는 의미로 'ANY'라는 태그를 사용하였다. 부정규칙 셋에서 또한 마찬가지의 과정을 거친다. 위의 과정은 (그림 6)의 1>과 2>에 나타나 있다.

3.3.2 긍정/부정 규칙 셋으로부터 방향규칙 습득

위의 과정에서 설명한 대로 긍정/부정 규칙을 정렬한 후, 두 규칙 셋으로부터 최종적으로 방향규칙을 얻는다. 긍정규칙 내의 각 <어휘 A, 구문카테고리 B, 방향 C>에 대하여 (그림 6)의 3> 과정을 반복한다.

(그림 6)에서, (3.1)은 부정규칙 셋에도 긍정규칙과 똑같은 <A, B, C>가 존재하는 경우로서 방향 정보 C가 ANY로 바뀐 최종 방향규칙을 얻는다. 구문카테고리 A와 어휘 B에 대하여 긍정규칙과 부정규칙 양쪽에 똑같은 방향태그 C가 존재하기 때문에, 방향정보는 의미가 없다. 그러므로 방향정보 C 가 ANY로 바뀐다.

(3.2)의 경우에는 구축된 방향규칙이 긍정규칙의 방향정보 C를 그대로 사용하는 경우다. 이 경우는 부정규칙 셋이 'C의 역방향' 정보를 가지고 있다. 'C의 역방향'이라 함은, C가 RIGHT인 경우는 LEFT, C가 LEFT인 경우는 RIGHT 방향을 가지고 있다는 의미다. 부정규칙 셋에서의 방향은 긍정규칙 셋에서의 것과 반대이므로, 이 경우 긍정규칙 셋의 방향정보는 신뢰도가 높다. 따라서 긍정규칙과 똑같은 방향을 가진 방향규칙 <A, B, C>을 구축한다.

(3.3)의 경우에는 부정규칙 셋이 A, B와 연관된 어떤 규칙도 가지고 있지 않는 경우다. 이 때는 긍정규칙 정보에만 의존할 수밖에 없으므로, 긍정규칙 셋에서의 템플릿과 같은 방향규칙을 획득한다.

위의 (3.1)~(3.3) 과정을 더 이상의 방향규칙이 획득되지 않을 때까지 반복한다. 그리하여 최종적으로 획득된 방향규칙은 <표 2>와 같다. 긍정규칙 셋에서 나타나지 않은 구문카테고리에 대해서는 어떤 방향도 허용하는 것으로 간주하여, <표 2>에서 묘사된 바와 같이 구문카테고리 'otherwise'에 대해서는 방향 'ANY'를 허용하는 규칙을 추가한다. 학습데이터 사이즈가 작기 때문에, 'otherwise' 정보는 데이터부족 문제를 해결할 수 있다.

〈표 2〉 단계 3에 의해 구축된 방향 규칙의 예

어휘	구문관계	방향
act	comp_as:V-N	RIGHT
act	comp_to:V-V	ANY
act	subj:V-N	LEFT
act	comp_in:V-N	RIGHT
act	otherwise	ANY
activate	subj:V_PASS-N	RIGHT
activate	subj:V-N	LEFT
activate	obj:V-N	RIGHT
activate	otherwise	ANY
affect	subj:V-N	LEFT
affect	otherwise	ANY
bind	subj:V-N	LEFT
bind	comp_to:V-V	ANY
bind	otherwise	ANY
block	obj:V-N	RIGHT
...	...	...

3.4 제안된 방법을 테스트데이터에 적용

우리는 LLL05가 제공한 유전자 사전을 사용하여 테스트 셋으로부터 에이전트 후보를 인식한다. 에이전트 후보 노드로부터 시작하여, 우리는 모든 가능한 구문 사슬을 확장해 나간다.

3단계를 거쳐 획득된 유전자-전이 구문관계, 상호작용동사 리스트, 방향규칙은 아래의 단계를 따라서 테스트데이터에 적용된다. 각 구문사슬에 대해, 우리는 아래의 과정을 반복한다.

1) 만약 현재 노드가 유전자이고 구문사슬이 상호작용동사를 포함하면, 우리는 현재 노드를 타겟으로 결정하고, 구문사슬의 확장을 멈추고 종료한다.
2) 다음 노드 후보를 잇는 구문카테고리가 유전자-전이 구문관계에 속하면, 우리는 다음 노드 후보를 추가함으로써 구문사슬을 확장한다.
3) 만약 현재 노드 어휘가 상호작용동사이고 다음 노드 후보의 방향 및 다음 노드 후보를 잇는 구문카테고리가 방향규칙과 일치하면, 우리는 구문사슬을 확장한다.

최종적으로 얻어진 구문사슬에서, 우리는 첫 번째 노드를 에이전트, 마지막 노드를 타겟이라고 결정한다.

〈표 3〉 본 논문의 시스템과 기존 시스템과의 성능 비교

시스템들		Green-wood [11]	Haken-berg [4]	Goad-rich [12]	Rie-del [15]	Popeli-nsky [13]	Katren-ko [14]	본 논문의 시스템
테스트 데이터 성능 (%)	정확률	22.2	28.1	28.3	60.9	46.5	39.2	72.5
	재현율	11.1	31.4	79.6	46.2	50.0	26.5	68.5
	F-measure	14.8	29.6	41.7	52.6	48.2	31.6	70.4

4. 실 험

유전자 상호작용을 위한 실험 셋은 객관적인 비교를 위하여 LLL05에서 제공한 학습데이터와 테스트데이터를 그대로 사용한다. 학습에서의 실험성능은 LLL05에서 제공한 도구를 사용하여 측정하였고, 테스트데이터에 대한 성능은 LLL05 홈페이지에서 결과를 측정할 수 있으므로 객관적으로 성능 체크를 할 수 있다. 학습데이터로는 대용어와 생략이 없는 문장으로 구성된 첫 번째 학습데이터인 77문장을 사용했고, 테스트데이터는 144문장으로 구성되어 있다. 실제로 LLL05의 학습데이터에서는 대용어나 생략이 있는 문장에 대한 구문관계 정보는 제공해 주지 않으므로, 대용어가 포함된 데이터는 학습데이터로 고려하지 않았다.

우리는 다음 2가지에 초점을 맞춰서 실험을 했다.

1. 우리의 3단계 방법의 성능 vs. 다른 실험 방법들의 성능
2. 각 단계를 없앴을 경우의 성능 변화

그리하여 실험에서 다음 결과를 얻었다.

1) 우리의 3단계 방법은 70.4%의 F-measure 성능을 보였다. (표 3 참조)
2) 유전자 상호작용에 대한 우리의 3단계 인식 방법은 기존의 다른 방법들에 비해 17.8~55.6%의 성능 향상을 보였다. (표 3 참조)
3) 3단계 중 하나의 과정이 삭제될 때, 성능이 급격히 안 좋아졌다. (표 4 참조)

표 3에서 보듯이, 6개의 기존 성능들과 비교해 본 결과, 우리 방법이 정확률 72.5%, 재현율 68.5%, F-measure 70.4%로 가장 성능이 좋았다. 또한 각 단계의 공헌도를 파악하기 위해 단계 하나씩을 없앤 후 성능을 측정해 보았다. 우선, 단계 1을 생략한 경우 두 종류의 실험을 실시했다. 첫 번째 실험은 유전자-전이 구문관계로서 모든 구문관계를 포함한 경우로서, 많은 구문사슬이 만들어지게 되므로 재현율은 더 높아졌으나 정확률이 급격히 낮아져서 F-measure의 성능이 떨어졌다. 두 번째 실험은 어떤 구문관계도 유전자-전이 구문관계에 포함되지 않도록 한 경우다. 이 때는 구문사슬이 확장되는 경우가 줄어들어서 완성된 구문사슬의 개수가 급격히 줄어들게 되므로 <표 4>에서 제공하는 다섯 종류의 실험에서 가장 낮은 재현율을 보였다.

단계 2를 제외한 실험의 경우는 모든 동사를 상호작용동사로 가정하여 실험해 보았다. 이 때 또한 많은 잘못된 구문사슬들이 만들어지기 때문에 정확률이 가장 낮았다. 단계 3을 제외한 실험의 경우는, 방향규칙이 없다고 가정하고 실험해 보았다. 구문사슬을 확장할 때 일치여부를 따졌던 방향규칙정보가 사라지게 되므로 구문사슬의 생성이 많아진다. 따라서 재현율은 더 높아졌으나 정확률이 39%로서 전체적으로 F-measure의 성능이 급격히 떨어졌다. 종합해 보면, 단계 1인 유전자-전이 구문관계의 형성은 재현율 향상에 기여를 하고 단계 2인 상호작용동사 리스트와 단계 3인 방향규칙은 정확률 향상에 기여했음을 알 수 있다.

위의 실험들을 종합한 결과, 본 논문에서 제안한 3가지 단계가 유전자 상호정보를 인식하는 데 중요하다는 것을 알 수 있다.

〈표 4〉 각 단계를 제외했을 경우 성능 변화

	학습 데이터 성능(%)		테스트 데이터 성능(%)
모든 단계를 다 사용했을 경우	정확률	89	72.5
	재현율	87	68.5
	F-measure	88	70.4
단계 1을 제외했을 경우 (유전자-전이 구문관계로서 모든 구문관계를 포함시켰을 경우)	정확률	60	34.7
	재현율	90	75.9
	F-measure	72	47.6
단계 1을 제외했을 경우 (유전자-전이 구문관계로 어떤 구문관계도 포함하지 않을 경우)	정확률	97	64.2
	재현율	32	16.6
	F-measure	48	26.4
단계 2를 제외했을 경우 (상호작용동사로 모든 동사를 포함함)	정확률	31	25.1
	재현율	92	90.7
	F-measure	46	39.3
단계 3을 제외했을 경우 (방향규칙을 없앴을 경우)	정확률	38	39.0
	재현율	89	72.2
	F-measure	53	50.6

5. 결 론

이 논문은 문서로부터 유전자 상호작용을 인식하기 위해서 구문관계에 기반한 3단계 방법을 제안한다. 실제 상호작용하는 에이전트와 타겟이 상호작용동사와 직접적인 구문관계를 맺고 있지 않은 경우가 많으므로, 실제 직접적인 구문관계를 맺고 있는 있는 용어 노드들까지 에이전트와 타겟이 전이할 수 있도록 해주는 구문관계들을 인식하는 유전자-전이 구문관계 추출을 첫 단계에 하고, 두 번째 단계에는 상호작용동사들을 추출한다. 실제 상호작용하는 두 유전자를 인식한 후에는 두 유전자 중 어느 것이 에이전트이고 어느 것이 타겟인지를 판단해야 하므로 이를 위해 에이전트로부터 타겟까지 이르는 구문관계의 방향규칙을 세 번째 단계에서 획득한다.

실험 결과, 본 논문에서 제안한 방법이 기존의 방법보다 더 좋은 성능을 보였고, 학습데이터가 적음에도 불구하고

실제 70.4%의 F-measure 성능을 나타냈다. 또한 각 단계별로 중요도를 체크해 본 결과, 두 번째와 세 번째 단계가 정확률 향상에 기여를 하였고, 특히 세 번째 단계인 방향규칙이 정확률을 높이는 데 가장 큰 역할을 한다는 것을 알 수 있다. 또한 첫 번째 단계는 재현율을 높이는 데 기여를 하였다.

앞으로 큰 학습데이터를 대상으로 위의 방법의 적용 결과 어느 정도 성능이 향상될 수 있는지 테스트해 볼 계획이다. 또한 현재는 LLL05에서 제공하는 구문관계로부터 정보를 얻고 있는데, 추후 실제 구문분석기를 실시간으로 사용하여 LLL05와 포맷이 다른 구문분석기의 구문관계 결과를 가지고 얼마나 견고한 성능을 낼 수 있는지 실험해 볼 계획이다. 또한 구문분석기의 오류에도 불구하고 어느 정도 성능이 유지될 수 있는지 실험도 해 볼 필요가 있다.

참 고 문 헌

[1] D. Otasek. K. Brown, I. Jurisica, "Confirming protein-protein interactions by text mining", SIAM Conference on Text Mining, Bethesda, Maryland, April 2006

[2] J.C.Park, H.S.Kim, J.j.Kim, "Bidirectional Incremental Parsing for Automatic Pathway Identification with Combinatory Categorial Grammar", Pacific Symposium on Biocomputing (PSB), pp. 396-407, Hawaii, USA, 2001

[3] C.Blaschke, M.A.Andrade, C.Ouzounis, and A.Valencia, "Automatic extraction of biological information from scientific text: protein-protein interactions", Proceedings of the seventh international conference on Intelligent Systems for Molecular Biology (ISMB 99), pp. 60-67, 1999

[4] J. Hakenberg, C. Plake, U. Leser, H. Kirsch, and D. R-Schuhmann, "LLL05 Challenge: Genic Interaction Extraction - Identification of Language Patterns Based on Alignment and Finite State Automata", Proceedings of ICML05 workshop on Learning Language in Logic (LLL05), pp.38-45, 2005

[5] M. Huang, X. Zhu, Y. Hao, D. G. Payan, K. Qu, and M. Li, "Discovering patterns to extract protein-protein interactions from full texts", Bioinformatics, Vol.20, pp.3604-3612, 2004

[6] N. Daraselia, A. Yuryev, S. Egorov, S. Novichkova, A. Nikitin, and I. Mazo, "Extracting human protein interactions from medline using a full-sentence parser", Bioinformatics, Vol. 20, pp.604-611

[7] B. Stapley, L. Kelley, and M. Sternberg, "Predicting the sub-cellular location of proteins from text using support vector machines", Proceedings of the Pacific Symposium on Biocomputing, pp.374-385, 2002

[8] B. Rosario, and M. Hearst, "Classifying semantic relations in bioscience texts", Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics(ACL), pp. 430-437, 2004

[9] J. Xiao, J. Su, G. Zhou, and C. Tan, "Protein-Protein Interaction Extraction: A Supervised Learning Approach", Proceeding of the Symposium on Semantic Mining in Biomedicine, pp.51-59, 2005

[10] J. Saric, L. Jensen, R. Ouzounova, I. Rojas, and P. Bork, "Large-scale Extraction of Protein/Gene Relations for Model Organisms", Proceeding of the Symposium on Semantic Mining in Biomedicine, pp.50, 2005

[11] M. A. Greenwood, M. Stevenson, Y. Guo, H. Harkema, and A. Roberts, "Automatically Acquiring a Linguistically Motivated Genic Interaction Extraction System", Proceedings of ICML05 workshop on Learning Language in Logic (LLL05), 2005

[12] M. Goadrich, L. Oliphant, J. Shavlik, "Learning to Extract Genie Interactions Using Gleaner", Proceedings of ICML05 workshop on Learning Language in Logic (LLL05), 2005

[13] L. Popelinsky, J. Blatak, "Learning genic interactions without expert domain knowledge: Comparison of different ILP algorithms", Proceedings of ICML05 workshop on Learning Language in Logic (LLL05), 2005

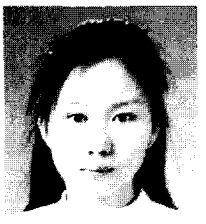
[14] S. Katrenko, M. S. Marshall, M. Roos, and P. Adriaans, "Learning Biological Interactions from Medline Abstracts", Proceedings of ICML05 workshop on Learning Language in Logic (LLL05), 2005

[15] S. Riedel, and E. Klein, "Genic Interaction Extraction with Semantic and Syntactic Chains", Proceedings of ICML05 workshop on Learning Language in Logic (LLL05), 2005

[16] D. Lin, "Dependency-based evaluation of MINIPAR", In Workshop on the Evaluation of Parsing Systems, "Dependency-based evaluation of MINIPAR", In Workshop on the Evaluation of Parsing Systems, 1998

[17] P. Uetz, R. L. Finley, Jr. "From protein networks to biological systems", FEBS Lett 579:1821-182, 2005

[18] Rinaldi F, Schneider G, Kaljurand K, Hess M, Andronis C, Konstanti O, Persidis A: "Mining of Functional Relations between Genes and Proteins over Biomedical Scientific Literature using a Deep-Linguistic Approach" *Artificial Intelligence in Medicine*, Vol. 39, Issue 2, pp. 127-136, 2007



김 미 영

e-mail : miykim@sungshin.ac.kr

1995년 3월~1999년 2월 포항공과대학교 컴퓨터공학과 학사

1999년 3월~2005년 8월 포항공과대학교 컴퓨터공학과 박사

2006년 3월~현재 성신여자대학교 컴퓨터정보학부 전임강사

관심분야 : 자연언어처리, 정보검색, 기계번역 등