기계 학습을 이용한 바이오 분야 학술 문헌에서의 관계 추출에 대한 실험적 연구*

An Experimental Study on the Relation Extraction from Biomedical Abstracts using Machine Learning

최 성 필 (Sung-Pil Choi)**

목 차

초 록

본 논문에서는 지지벡터기계(Support Vector Machines, SVM) 기반의 기계 학습 모듈을 활용하여 특정 문장 내에서의 두 개체 간의 관계를 자동으로 식별하고 분류하는 바이오 분야 관계 추출 시스템을 제안한다. 제안된 시스템의 특징은 개체를 포함하고 있는 문장 내에서 풍부한 언어 자질을 추출하여 학습에 활용함으로써 그 성능을 극대화할 수 있는 다양한 기능들을 포함하고 있다는 점이다. 제안된 시스템의 성능 측정을 위해서 전 세계적으로 많이 활용되고 있는 바이오 분야 관계 추출 표준 컬렉션 3가지를 활용하여 심층적인 실험을 수행한 결과 모든 컬렉션에서 높은 성능을 획득하여 그 우수성을 입증하였다. 결론적으로, 본 논문에서 수행한 바이오 분야 관계 추출에 대한 광범위하고 심층적인 실험 연구가 향후 기계학습 기반의 바이오 분야 텍스트 분석 연구에 많은 시사점을 제공할 것으로 보인다.

ABSTRACT

This paper introduces a relation extraction system that can be used in identifying and classifying semantic relations between biomedical entities in scientific texts using machine learning methods such as Support Vector Machines (SVM). The suggested system includes many useful functions capable of extracting various linguistic features from sentences having a pair of biomedical entities and applying them into training relation extraction models for maximizing their performance. Three globally representative collections in biomedical domains were used in the experiments which demonstrate its superiority in various biomedical domains. As a result, it is most likely that the intensive experimental study conducted in this paper will provide meaningful foundations for research on bio-text analysis based on machine learning.

키워드: 관계 추출, 지지벡터기계, 단백질 간 상호작용 추출, 텍스트 마이닝, 기계 학습
Relation Extraction, Support Vector Machines, Protein-Protein Interaction Extraction, Text Mining, Machine Learning

---

* 본 연구는 한국과학기술정보연구원 주요사업 "초고성능컴퓨팅 기반 건강한 고령사회 대응 빅데이터 기술개발" 과제의 연구비 지원으로 수행되었음(K-16-L03-C02-S02).

** 경기대학교 문헌정보학과 조교수(spchoi@kgu.ac.kr, sungpil@gmail.com)

논문접수일자: 2016년 5월 4일 최초심사일자: 2016년 5월 4일 게재확정일자: 2016년 5월 18일
한국문헌정보학회지, 50(2): 309-336, 2016. [http://dx.doi.org/10.4275/KSLIS.2016.50.2.309]

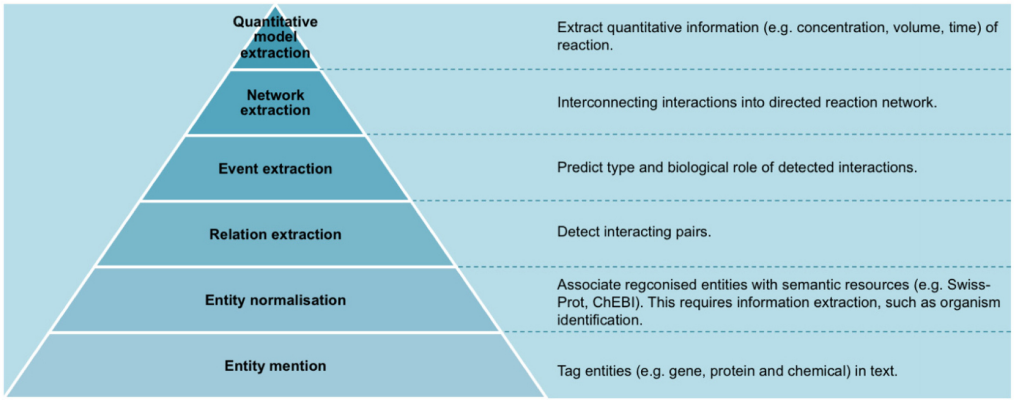
1. 서 론

고수준의 정보 기술은 연구개발 업무의 질적 수준과 효율성을 높임으로써 궁극적으로는 개별 과학 분야의 총체적인 발전에 실질적인 공헌을 할 수 있다. 특정 과학기술분야에서의 연구개발 효율성과 연구 종사자의 편의성을 확보하기 위해서 수행되고 있는, 기술 문헌의 심층 분석을 통한 기술 지식 창출은 이미 전 세계적으로 매우 중요한 이슈로 대두되고 있다. 특히 다양한 과학기술분야 중 기술적 진보가 빠르며 새로운 지식의 생산이 급격하게 진행되고 있는 의학, 생물학, 생화학 분야 내에서의 학술 정보의 지식화 연구는 핵심적인 글로벌 과제로 인식되고 있으며 전 세계적으로 많은 투자가 이루어지고 있다(Ananiadou, Kell and Tsujii 2006).

생의학 분야에서 전문가들이 생성할 수 있는 다양한 지식 중에서, 인간을 비롯한 모든 생명체 및 질병 등의 생체 활동 메커니즘을 규명하여, 직관적이고 가시적으로 표현한 대사 경로나 신호 전달 경로 그리고 분야별 온톨로지 등과 같은 생의학 지식베이스는 가장 수준이 높고 핵심적인 분야 지식이며, 미국 및 유럽의 선진국들은 이를 효율적으로 구축, 활용하기 위한 기술적, 제도적 노력을 지속적으로 경주하고 있다(Ananiadou et al. 2010).

일반적으로 생의학 지식베이스란 객관적인 연구 성과로서 도출된 학술 정보(논문, 특허, 기술 보고서 등)에 출현한 다양한 전문 용어와 그들 간의 의미적 상관관계를 네트워크 형식으로 표현하고 있으며, 생명공학 관점에서는 단백질, 유전자, 세포 등의 생체적 요소 간의 역학관계 혹은 상호작용 등을 세밀하게 기술한 생물학적 심층 지식으로 볼 수 있다. 이러한 심층 지식을 효율적으로 구축하기 위해서는 텍스트 문헌을 기계적으로 분석하여 세포명, 화합물명, 질병, 약품, 치료법 등과 같은 핵심 용어들을 자동으로 추출하고, 이들 간의 의미적 연관 관계를 문헌 내에 기술된 정보를 바탕으로 식별하는 바이오 텍스트 마이닝(Bio Text Mining) 및 정보 추출(Information Extraction) 기술이 필수적이다(Li, Liakata and Rebholz-Schuhmann 2014).

〈그림 1〉은 생의학 분야 학술 문헌에서 다양한 기술을 활용하여 최종적인 지식 네트워크를



〈그림 1〉 바이오 분야 네트워크 생성을 위한 6단계(Li et al. 2014)

추출하는 과정을 세부적으로 도식화하고 있다. 문헌에 존재하는 단백질, 유전자 및 화합물 개체를 식별하고, 개체 간의 관계를 추출한 다음, 이를 기반으로 생의학 이벤트(biomedical event)(Ananiadou et al. 2010)를 생성하여 이를 통합함으로써 네트워크가 생성됨을 보여주고 있다.

본 논문은 〈그림 1〉의 6단계 지식화 과정 중에서 관계 추출(Relation Extraction)에 대한 실험적 연구를 수행한다. 관계 추출은 문장 내에 존재하는 한 쌍의 개체 간에 의미적 연관 관계가 존재하는지 여부를 판단하는 관계 식별(Relation Identification)과, 관계가 존재할 경우 해당 관계의 종류를 판단하는 관계 분류(Relation Classification)로 구분할 수 있다(Choi et al. 2014). 〈그림 2〉는 대표적인 생의학 분야 자연어 처리 학술 대회인 BioNLP 2011에서 구축된 관계 추출 말뭉치 일부(BioNLP-ST-2011_REL)를 보여주고 있다. 그림에서 보듯이, 텍스트 내에는 다양한 개체 및 용어가 활용되고 있고, 이들 간의 문장 내의 표현들은 이들 간의 의미적 관계(Component, Protein-Component 등)를 설명하고 있다. 관계 추출은 문장 내에서 식별된 개체 간의 관계를 자동으로 찾아내는 기능을 수행한다. 본 논문에서는 지지벡터기계 기반의 기계 학습 모듈을 활용하여 특정 문장 내에서의 두 개체 간의 관계를 자동으로 식별하고 분류하는 바이오 분야 관계 추출 시스템을 제안한다. 또한 제안된 시스템의 성능을 객관적으로 측정하기 위해서 다양한 실험 말뭉치들을 활용하여 성능 측정을 수행한 결과를 도출하고 분석한다.

논문의 구성은 다음과 같다. 우선 2장에서는 범용 관계 추출 및 바이오 분야 관계 추출 관련한 대표적인 기존 연구들을 분석하고 한계점을

〈그림 2〉 BioNLP-ST-2011_REL 컬렉션 일부

지적한다. 이어서 3장에서는 본 연구에서 개발된 기계학습 기반의 바이오 분야 관계 추출 시스템을 소개하고 세부적인 기능들을 살펴본다. 또한 4장에서는 제안된 시스템의 성능 측정을 위한 실험 결과를 설명하고 분석하며, 마지막으로 5장에서는 결론 및 향후 연구 방향을 기술한다.

2. 관련 연구

생의학 분야에서의 관계 추출 기술은 단백질 간 상호작용 추출(Protein-Protein Interaction Extraction, PPIE)에 주로 적용이 되어 왔다. 단백질 간 상호작용 정보는 서로 인접한 두 단백질들 상호 간의 직접적인 연관 관계뿐만 아니라, 수십 나노미터 떨어진 수용성 단백질(hydrated protein)들 사이에서 수용체(aqueous solution), 전해질(electrolyte) 등에 의해 이루어지는 간접적인 상호작용까지도 포괄한다("Protein-protein interaction," 2016). 이러한 PPI 정보는 다양한 생물학적 기능을 설명하고 분석하는데 핵심적인 역할을 수행하며, 현재까지 생의학 분야 연구자들이 생화학적 기법, 생물 물리학적 기법 등을 기반으로 실험 혹은 이론적 분석에 의거하여 관련 연구를 수행해 왔다. 도출된 연구 성과는 텍스트 형태로 논문에 주로 기술되는데, 단백질 간 상호작용 자동 추출은 텍스트 내에 표현된 단서 어휘 및 구문 구조 자질들을 활용하여 출현한 다수의 단백질들 간의 상호작용에 관한 정보를 자동으로 추출하는 기술이며 PPI 식별(PPI Identification, PPII)과 PPI 분류(PPI Classification, PPIC)로 구성된다(Choi and Myaeng 2010). PPII는 다중의 단백질명을 가지는 특정 문장이 그 단백질들 간의 상호작용을 표현하고 있는지 여부를 판단하는 기술이다. 기계 학습 관점에서 볼 때, PPII 문제는 이진 분류(binary classification) 모델로 표현할 수 있으며, 현재까지 많은 연구가 진행되어 왔다. 더불어 PPI는 PPI를 포함한 혹은 포함했다고 판단된 문장을 대상으로 보다 심층적인 분석을 통해서 구체적인 상호 작용의 종류를 결정하는 작업이다. 상호작용의 종류가 3개 이상이므로 다중 분류 모델(Multi-class Classification)로 설명될 수 있으며 현재까지도 다양한 방법론들이 연구되고 있다. 또한 PPII와 PPIC는 하나로 결합되어 단일 다중 분류 모델로도 표현될 수 있다.

Zhou and He(2008)는 최근까지 연구된 PPI 추출 모델을 언어학적 방법, 규칙 기반 방법 그리고 기계 학습 및 통계적 기법의 세 가지 종류로 분류하여 설명한다. 우선 언어학적 기법에서는 PPI를 표현할 수 있는 대표적인 문장 구조를 분석하여 이를 문법으로 구성한다. 이러한 요소 문법들은 상호작용 추출을 위한 특화된 언어 분석 시스템(품사 태거, 기저구 인식기, 구문 분석기 등)의 기반 문법으로 활용된다. PPI 추출 기술의 특성에서 볼 때, 문장에 대한 심층 분석은 필수적이며, 그 분석 수준에 따라 부분 구문 분석을 활용한 기법과 완전 구문 분석 기반 기법으로 나눌 수 있다. 부분 구문 분석 기법은 문장을 요소 기저구들로 분리하고 이들 기저구 간의 지역적 의존 관계를 파악함으로써 PPI 포함 문장에 대한 식별을 가능하게 하였다(Sekimizu, Park and Tsujii 1998). 이에 반해서 완전 구문 분석 기법은 단백질, 유전자 혹은 세포 간의 상호작용을 식별할 수 있는 어휘 분석기와 확장된

문맥 자유 문법을 구성하여 이를 기반으로 특화된 구문 분석을 수행한다(Papanikolaou et al. 2014; Temkin and Gilder 2003). 이렇게 도출되는 구문 구조의 패턴을 파악함으로써 PPI가 포함된 문장을 식별하였다. 두 번째로 규칙 기반 기법은 상호작용 표현의 단서가 될 수 있는 어휘적 패턴 집합을 수작업으로 정의하고, 이를 기반으로 문장에서 이들 패턴과 일치하는 부분을 찾는 과정을 수반한다. 이 범주에 속하는 방법의 하나로서 Blaschke, Hirschman and Valencia(2002)는 상호작용 단서 어휘 집합을 수집하고 이를 기반으로 어휘적 규칙을 고안하여 PPI 추출에 적용하였다. 이에 따라, 문장 내에서 발견한 어휘적 규칙에 대한 신뢰도를 자동으로 계산하여 이를 추출된 PPI의 신뢰도로써 활용하였다. Ono et al.(2001)은 부정 표현 구조까지도 포괄하는 어휘 및 구문 자질 기반의 상호작용 추출 패턴을 정의하고, 효모의 일종인 "사카로미세스 세레비시아(Saccharomyces cerevisiae)"와 대장균속 세균인 "에스케리치아 콜리(Escherichia coli)"에 관한 문서를 대상으로 실험한 결과 높은 성능을 보여주었다. 더불어 Fundel, Küffner and Zimmer(2007)는 의존 구문 트리 기반의 관계 추출 모델을 제안함으로써 고수준 자연어 처리 시스템을 적용한 관계 추출의 중대한 발판을 마련하였다. LLL과 HPRD50을 이용한 실험에서 각각 82%, 78%(F-score)의 높은 성능을 나타내었다.

마지막으로 기계학습 및 통계적 기법은 가장 최근에 도래한 기법으로서, 지도학습(supervised learning) 혹은 반지도 학습(semi-supervised learning) 기반의 기계학습 모델을 적용하여, 미리 수작업으로 구성된 학습 집합을 기반으로 관계 및 상호작용을 표현하는 핵심 단서인 자질 집합을 자동으로 추출하고 이를 학습에 적용한다. 확장성 및 효율성 측면에서 가장 높은 성능을 나타내고 있으며, 지금까지도 연구가 활발하게 진행되고 있다(Andrade and Valencia 1998; Bunescu et al. 2005; Craven and Kumlien 1999). 기계 학습 기반 방법 중에서도 특히 커널 기반 관계 추출에 관한 연구가 활발하게 진행되고 있다. Airola et al.(2008)은 기존 의존 구문 트리 커널의 단점을 극복하기 위해서 후보 문장들에 대한 의존 구문 트리를 그래프로 변형하고 이에 그래프 커널을 이용하여 단백질 간 상호작용 추출 시도를 하였으나, 기존의 기법에 비해서 나은 성능을 나타내지는 못 하였다. 한편, Miwa et al.(2009)은 단어 자질 커널, 구문 트리 커널, 그래프 커널 등을 모두 적용한 혼합 커널을 구성하여 앞에서 소개한 총 5가지의 말뭉치를 대상으로 실험을 수행하였다. 그러나 적용한 기법의 다양성이나 광범위한 단서 자질의 적용에도 불구하고 성능은 일반적인 수준이었다. 특히 Fundel et al.(2007)과 비교해서는 오히려 성능이 낮게 나타났다(LLL: 80.1%, HPRD50: 70.9%).

본 논문의 접근 모델은 위에서 기술한 방법론들 중에서 지지벡터기계를 활용한 지도학습 기법을 활용한다. 현재까지 고안된 대부분의 자질 추출 방법론들을 조사하여 적용하고, 성능을 극대화할 수 있는 최적화 방법론을 채택하였다. 또한 주로 단일 학습 컬렉션을 이용한 실험 결과를 제시하고 있는 기존의 연구들에 반해, 본 연구에서는 다양한 학습 및 평가 컬렉션을 적용하여 성능 평가의 객관성을 높인다.

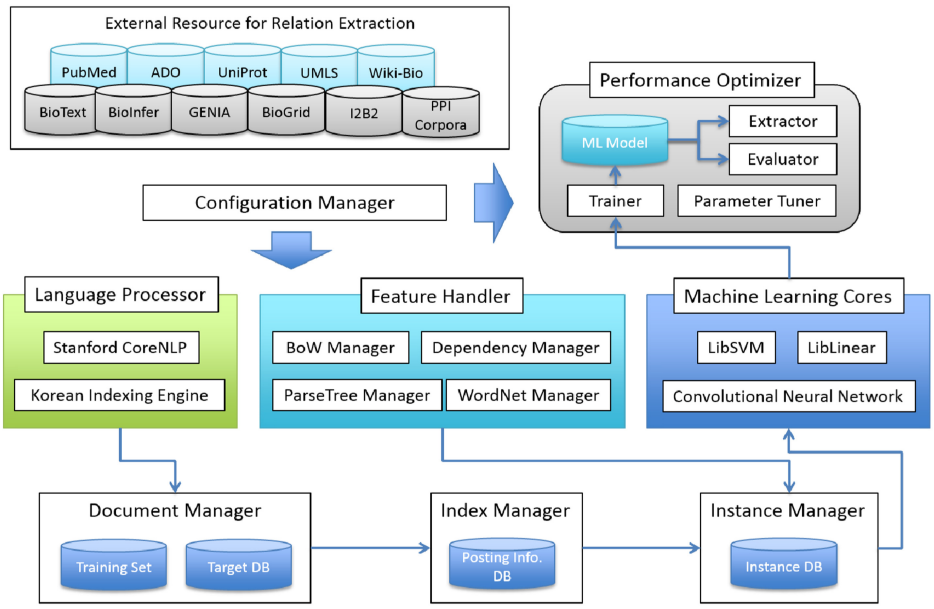
3. 기계 학습 기반 바이오 분야 관계 추출 시스템

본 논문에서 개발된 시스템의 최종 목적은 문헌 내에 다양하게 출현하는 개체명 간의 의미적 연관 관계를 기계 학습을 이용하여 효과적으로 추출하는데 있다. 따라서 학습 집합 및 추출 대상 문헌 집합을 체계적으로 관리, 활용하는 기능과 추출 성능을 최적화하는 다양한 기능들이 부가적으로 추가되어야 한다. 3장에서는 단지 기계학습 모듈뿐만 아니라 관계 추출 프로세스 실행의 효율성을 확보할 수 있는 다양한 부가 기능까지도 세부적으로 설명한다. 본 시스템의 가장 큰 특징은 특정 학습 집합이나 대상 자료에 국한되지 않고 다양한 분야에 범용적으로 적용할 수 있는 관계 추출 통합 테스트베드 역할을 수행한다는 점이다.

3.1 시스템의 구조 및 설명

본 연구에서는 다양한 학습 집합을 활용하여 위의 두 가지 기능을 모두 수행할 수 있는 기계 학습 기반 관계 추출 시스템 프로토타입을 개발하였다. 〈그림 3〉은 개발된 관계 추출 시스템의 전체 구성도를 보여주고 있다.

일반적으로 대부분의 이전 연구 결과나 논문에서는 관계 추출의 두 가지 기능 중에서 주로 관계 분류에 편중되어 연구가 진행되어 왔다. 그러나 본 연구에서는 개발된 시스템의 활용성을 높이기 위해서 약 20가지 이상의 관계 클래스를 다루는 관계 분류는 물론 한 쌍의 개체명을 포함하는 문장에서 그 개체 쌍이 문장 내에서 의미적으로 연관성을 가지는지를 엄밀하게 분석하는 관계식별 기능도 수행할 수 있는 통합 시스템을 개발하였다.



〈그림 3〉 바이오 분야 관계 추출 시스템 구성도

개발된 시스템은 총 8가지 세부 모듈로 구성되어 있다. 우선 문서 관리자(Document Manager)는 학습 집합 혹은 배치로 처리될 대상 문서를 관리하고 메모리상에서 검색할 수 있는 기능을 가지고 있다. 특히 문서 관리자에는 입력된 텍스트를 분석하고 그 결과를 다시 텍스트 파일로 저장할 수 있는 기능이 존재한다. 이는 입력 텍스트에 대해서 언어 처리 모듈(Language Processor)을 이용하여 분석(색인어 추출 및 구문 분석 등)하고 유효한 자질을 추출하여 이를 수치 벡터로 변환하는 과정을 단순화하여, 다양한 자질 추출 기법을 적용할 수 있는 방법론을 제공한다. 문서 관리자에서 관리되는 문서의 구조를 설명하는 필드는 〈표 1〉과 같다.

최초로 입력되는 문서에는 "#KWD=", "#DEP=", "#PT="의 값이 지정되어 있지 않으며 문서 관리자에서 언어 분석을 통해서 도출된 결과를 저장하게 된다. 이렇게 분석된 분석 결과의 실제 저장 예는 〈그림 4〉와 같다.

〈그림 4〉에서 보는 바와 같이 키워드 추출 결과, 의존 구문 분석 결과 그리고 구문 구조가 문서 레코드에 저장되어 있다. 문서 혹은 문장에 대한 언어 분석은 한 번만 수행하면 되므로 이 결과를 바탕으로 다양한 자질 추출 기법들을 적용하여 성능을 개선시킬 수 있다.

색인 관리자(Index Manager)는 앞에서 분석 완료된 결과를 바탕으로 주요 색인어를 추출하고 이들에 대한 가중치 부여를 수행하는 모듈이다. 비록 관계 추출에서는 그 활용 가치가 높지 않으나 관계 인스턴스의 단위가 크거나 문헌을 분류할 때 유용하게 활용할 수 있다.

인스턴스 관리자(Instance Manager)는 뒤

〈표 1〉 문서 관리자에서 관리되는 문서(문장) 레코드 구조

필드명	설명 (문서와 문장을 동일한 개념으로 고려한다)
@RELATION	하나의 문서(관계 인스턴스)를 구분하는 레코드 분리자
#ID	문서 식별자
#CON	문서의 내용 (실제 언어 분석이 되는 문장 혹은 문서)
#REP_CON	문서(문장) 내에 포함된 개체에 대해서 익명화시킨 문장 (기본적으로 이진 관계 추출을 처리하므로 한 쌍의 개체에 대해서 첫 번째 개체는 "ENTITYONE"으로 두 번째 개체는 "ENTITYTWO"로 대체함)
#ENT1	첫 번째 개체 문자열
#ENT2	두 번째 개체 문자열
#ENT1_TYPE	첫 번째 개체의 유형 (단백질, 유전자, 질병, 세포, 약품 등)
#ENT2_TYPE	두 번째 개체의 유형 (단백질, 유전자, 질병, 세포, 약품 등)
#ENT1_START	문장 혹은 문서에서 첫 번째 개체의 출현 위치
#ENT1_END	문장 혹은 문서에서 첫 번째 개체의 마지막 위치
#ENT2_START	문장 혹은 문서에서 두 번째 개체의 출현 위치
#ENT2_END	문장 혹은 문서에서 두 번째 개체의 마지막 위치
#CAT	분류 정보
#KWD	문장 혹은 문서에서 추출된 키워드 리스트 및 출현 빈도
#DEP	문장 혹은 문서에 대한 의존 구문 분석 결과 (단어의 원형 포함)
#PT	문장 혹은 문서에 대한 구문 트리 (Penn-Tree Bank 스타일로 저장)

```
@RELATION
#ID=0
#CON=alpha-catenin inhibits beta-catenin signaling by preventing formation of abeta-catenin*T-cell factor*DNA complex.
#REP_CON=ENTITYONE inhibits ENTITYTWO signaling by preventing formation of a beta-catenin*T-cellfactor*DNA complex.
#ENT1=alpha-catenin
#ENT2=beta-catenin
#ENT1_TYPE=Individual_protein
#ENT2_TYPE=Individual_protein
#ENT1_START=0
#ENT1_END=12
#ENT2_START=23
#ENT2_END=34
#CAT=INHIBIT
#KWD=prevent1 beta-catenin 1 t-cell 1 signaling 1 dna 1 complex 1 inhibit 1 formation 1 factor1 ENTITYTWO 1 entityone 1
#DEP=inhibit inhibits VBZ 2 nsubj entityone ENTITYONE NN 1 signaling signaling NN 4 nn ENTITYTWO ENTITYTWO NNP 3 inhibitinhibits VBZ 2 dobjsignaling signaling NN 4 inhibit inhibits VBZ 2 prepc_by prevent preventing VBG 6 preventpreventing VBG 6 dobjformation formation NN 7 beta-catenin beta-catenin NN 10 det a aDT 9 formation formation NN 7 prep_of beta-catenin beta-catenin NN 10 factorfactor NN 13 dep * *SYM 11 factor factor NN 13 nn t-cell T-cell NN 12 formation formationNN 7 depfactor factor NN 13 complex complex NN 16 dep * * SYM 14 complex complex NN 16 nndna DNANN 15 formation formation NN 7 dep complex complex NN 16
#PT=(ROOT(S (NP (NN ENTITYONE)) (VP (VBZ inhibits) (NP (NNP ENTITYTWO) (NN signaling))(PP (IN by) (S (VP (VBG preventing) (NP (NP (NP (NN formation)) (PP (IN of) (NP(DT a) (NN beta-catenin))) (X (X (SYM *)) (NP (NN T-cell) (NN factor)))) (X (X(SYM *)) (NP (NN DNA) (NN complex)))))))) (. .)))
```

〈그림 4〉 실제 문서 레코드 예제

에서 설명할 자질 추출 모듈을 이용하여 텍스트 문장을 하나의 수치 벡터로 저장하는 기능을 수행한다. 특정 자질에 대한 존재 유무를 0 또는 1로 지정하여 벡터를 구성함으로써 기계 학습 모듈이 학습을 수행할 때 복잡도를 낮추어 성능 및 속도 향상을 추구하였다. 인스턴스 관리자를 통해서 도출된 단일 문서에 대한 자질 벡터 예는 〈그림 5〉와 같다.

〈그림 5〉는 2건의 관계 인스턴스를 나타내고 있다. 첫 번째 필드는 관계의 종류를 식별자로 표시하고 있으며 이들 식별자는 카테고리 관리자에 의해서 관리된다. 두 번째 필드부터는 "자질번호:자질유무" 형태의 요소가 나열되어 있는 형태의 관계 인스턴스 자질이 표현되고 있다. 이렇게 구성된 자질 벡터는 기계 학습의 입력 벡터로 들어가게 된다.

```
6  3:1.000 4:1.000 172:1.000 236:1.000237:1.000 238:1.000 239:1.000 240:1.000 241:1.000
242:1.000 243:1.000 245:1.000 247:1.000 249:1.000 251:1.000 253:1.000 259:1.000
260:1.000 261:1.000 262:1.000 263:1.000 264:1.000 265:1.000 266:1.000 267:1.000
269:1.000 275:1.000 276:1.000 277:1.000 278:1.000 279:1.000 280:1.000 281:1.000
282:1.000 283:1.000 284:1.000 285:1.000 286:1.000 287:1.000 288:1.000 289:1.000
293:1.000 294:1.000 295:1.000 296:1.000 297:1.000

5  3:1.000 298:1.000 299:1.000 300:1.000301:1.000 302:1.000 303:1.000 304:1.000
305:1.000 306:1.000 307:1.000 308:1.000 309:1.000 310:1.000 311:1.000 312:1.000
313:1.000 314:1.000 315:1.000 316:1.000 317:1.000
```

〈그림 5〉 생성된 관계 추출용 자질 벡터 예시

언어 처리기(Language Processor)는 앞에서 설명한 문서 관리자(Document Manager)에서 텍스트를 분석하는 역할을 수행한다. 현재는 대상 데이터가 모두 영어이므로 영어 텍스트 분석을 위해서 Stanford CoreNLP(Manning et al. 2014)가 라이브러리 형태로 이식되어 있다.

자질 관리자(Feature Handler)는 앞에서 언급한 색인 정보, 텍스트 분석 결과 등을 바탕으로 관계 추출에 필요한 다양한 자질들을 추출하는 모듈이다. 총 5종류(어휘 자질, 의존 그래프 자질, 구문 자질, 의미 자질, 개체 자질 등)의 자질들을 추출할 수 있으며 보다 상세한 내용은 3. 관계 추출용 자질 종류 및 세부 사항에서 다룬다.

기계 학습 핵심 모듈(Machine Learning Core)은 다양한 형태의 기계 학습 모듈을 손쉽게 도입하여 이식할 수 있도록 구성된 Wrapper 클래스이며, 현재 LibSVM(Chang and Lin 2011)과 LibLinear(Fan et al. 2008)를 활용하고 있다. 비록 LibSVM이 다양한 커널(Kernel)들을 지원하며 많은 매개 변수들이 존재함으로 인해 성능 최적화에 많은 장점이 있을 것으로 예상했으나, 본 연구에서 수행된 관계 식별, 관계 분류 등에서는 그 효과가 미미했다. 결론적으로 본 연구에서는 학습 속도도 빠르고 성능도 큰 차이를 보이지 않는 선형 커널 기반의 LibLinear를 기반으로 현재까지 연구된 거의 모든 자질들을 추출하여 적용함으로써 관계 추출 성능을 극대화하는 방향으로 연구를 진행하였다.

성능 최적화 도구(Performance Optimizer)는 기계 학습 모델의 매개 변수, 자질 추출 방법, 자질 선택 방법 등을 지속적으로 변경하면서 가장 높은 수준의 성능을 나타내는 설정 정보를 찾아내는 기능을 수행한다. 기본적으로 N-겹 교차 검증(N-fold cross validation)을 기반으로 특정 환경 설정 값에 대한 성능 수치를 계산함으로써 주어진 학습 컬렉션을 기반으로 일반화 강도(generalization power)가 가장 높은 설정치를 선택한다.

환경 설정 관리기(Configuration Manager)는 관계 추출 엔진의 학습 및 검증을 위한 〈그림 6〉과 같은 기본 정보를 지정할 수 있는 기능을 제공한다.

〈그림 6〉의 설정 대상 중에서 관계 추출과

```
# 실행 모드 지정. (DOC_CAT, REL_EXT)
execution.mode=REL_EXT
# 분류 개수 지정 (따로 지정하지 않으면, 내부적으로 계산)
category.number=0
# 문서집합 내에서 레코드를 시작하는 헤더 지정.
document.record.header=@RELATION
# 원본 문서집합이 저장된 디렉토리 (다른 문서는 포함되어서는 안됨.)
document.directory=./relex/bio_infer
# 문서에 대한 색인 결과 저장 파일 지정.
document.info.file=./relex/bio_infer.di
# 색인 엔진 선택.
# 현재는 영어 분석 처리를 위한 STANFORD(English)만 지원.
index.engine.mode=STANFORD
# 색인 및 자질 추출이 완료된 역파일 저장 파일명 지정(가장 중요)
index.file=./relex/bio_infer.index
# 기계 학습 엔진 선택
# LIBSVM(다양한 커널 적용 가능), LIBLINEAR(선형 커널만 가능한 대신 속도가 빠름)
train.engine.mode=LIBLINEAR
# 기계 학습 엔진에 입력되는 매개변수 지정
train.engine.param=-q
# 학습 인스턴스 집합 저장 파일명 지정.
train.instance.file=./relex/bio_infer.train
# 기계 학습이 완료된 모델 파일명 지정.
train.model.file=./relex/bio_infer.model
# 학습이 완료된 모델을 기반으로 테스트를 수행할 실험 인스턴스 집합 파일명 지정.
test.instance.file=./relex/bio_infer.train`
# 문헌에서 추출된 자질 정보 저장 파일.
feature.file=./relex/bio_infer.feature
# 자질 가중치 방법 지정
# FC_TF, FC_LOG_TF, FC_ITF, FC_IDF, FC_TF_IDF, FC_LOG_TF_IDF, FC_TF_RF, FC_BIN
# 디폴트는 FC_TF_IDF
feature.weighting.method=FC_LOG_TF_IDF
# 자질 선택 기법 지정
# 현재 FS_SD_CDF, FS_SD_CTF, FS_DF, FS_MI_MAX, FS_MI_AVG, FS_CHI_MAX, FS_CHI_AVG,
FS_IG, FS_BNS_MAX, FS_BNS_AVG, FS_WLLR_MAX, FS_WLLR_AVG 기법 제공.
# 디폴트는 FS_DF
feature.selection.method=FS_DF
# 자질 선택 기법에 따른 선정 최소값 지정. (지정된 값보다 같거나 큰 자질만 선택)
feature.selection.thresh=0.01
# 자질 추출 방법 지정 (필수) -- lex, semlex, dep, parse, entity
feature.extraction.method=lex, semlex, dep, parse, entity
# 교차 검증을 위한 인스턴스 파일 지정.
cv.instance.file=./relex/bio_infer.train
# 교차 검증에서의 fold 개수 지정.
cv.fold.number=10
# 학습 모델, 자질 가중치 기법, 자질 선택 방법을 모두 적용한 최적화 작업의 결과 저장 파일.
optimizer.output.file=./relex/bio_infer.out
```

〈그림 6〉 관계 추출 설정 파일 예

직접적인 관련이 있는 항목은 입력 데이터(학습 집합 혹은 대상 문헌 집합) 내에서 단일 문서 혹은 문장을 구분 짓는 구분자를 지정하는 "document.record.header", 데이터가 존재하는 위치를 지정하는 "document.directory", 문서에 대한 분석 정보가 종합적으로 저장되는 파일명을 지정하는 "document.info.file", 색인 및 자질 추출이 완료된 역파일 저장 파일명을 지정하는 "index.file", 자질 추출 방법을 지정하고 있는 "feature.extraction.method" 등이 있다.

그 외에도 〈그림 3〉에서 보는 바와 같이, 고수준의 관계 추출에 필요한 다양한 외부 자원들을 수집, 관리, 활용하는 모듈 및 리포지터리가 존재한다. 그 중에서 BioText,[1] BioInfer,[2] Five PPI Corpora[3]는 바이오 분야 관계 식별 및 관계 추출을 위한 학습 컬렉션으로 활용된다.

3.2 관계 추출을 위한 기계 학습 모델

본 연구에서 관계 추출을 위해 활용된 기계 학습 모델은 Support Vector Machines(SVM)이다. 이 모델은 지도 학습 모델로서, p-차원에 존재하는 학습 벡터를 (p-1)-차원의 선형 초평면(linear hyperplane)으로 분류함에 있어서 두 분류 집합을 분리하는 초평면들의 집합 중에서 마진(margin)이 최대인 초평면을 선택하는 방법론을 제공한다. 여기서 마진이란 서로 다른 두 분류의 점들을 구분 짓는 초평면과 가장 근접한 점들 사이의 거리를 의미하므로 이론적으로 최적의 분류 초평면을 구할 수 있는 장점이 있다. 그러나 일반적으로 유한 차원 공간에서 데이터 집합이 선형 구분이 되지 않는 문제가 자주 발생하며 이러한 문제를 해결하기 위해서 초기의 유한 데이터 차원에서 더 높은 자질 차원로 모든 데이터 점들을 대응시킴으로써 이 고차원에서의 선형 분리가 가능하게 하는 방법이 제안되었으며 이 과정에서 다양한 커널 함수들이 활용된다.

이러한 SVM을 관계 추출에 적용하기 위해서 기존에 개발된 두 가지 시스템 즉, libSVM과 libLinear를 활용한다. libSVM은 SVM 기반 기계 학습을 위한 다양한 기능들을 제공하고 있으며 특히 C++, Java, Python, MATLAB, R C#, Perl 등의 프로그래밍 언어로 활용 가능한 많은 Wrapper들이 개발되었다. 현재 전 세계적으로 가장 많이 활용되는 SVM 기반 기계 학습 라이브러리라고 볼 수 있다. 또한 LIBLINEAR는 SVM에서 활용되는 커널 함수 중에서 대표적인 4가지 함수들 즉, 선형 커널(Linear Kernel), 다항 커널(Polynomial Kernel), 가우시안 방사 기저 함수 커널(Gaussian Radial Basis Function Kernel), 쌍곡 탄젠트 커널(Hyperbolic Tangent Kernel) 중에서 선형 커널만을 이용하면서 대용량의 학습 데이터를 효과적으로 처리할 수 있는 특징을 가지고 있다. 학습 속도가 매우 빠르면서도 기존의 SVM 기반 모델과 비교하여 성능이 우수한 것으로 평가된 이 모듈을 바탕으로 본 연구에서는 다중 자질 기반 관계 추출(multi-feature based relation extraction) 모델을 개발하였다.

1) http://biotext.berkeley.edu/
2) http://mars.cs.utu.fi/BioInfer/
3) http://mars.cs.utu.fi/PPICorpora/

3.3 관계 추출용 자질 종류 및 세부 사항

이 절에서는 개발된 기계 학습 기반 관계 추출 엔진에 활용되는 다양한 자질들을 유형별로 정리하여 설명한다. 관계 추출에서 활용되는 자질 추출 대상 인스턴스는 전체 문서가 아니라 그 범위가 제한된 특정 문장이므로 세밀한 언어 처리(구문 분석 등)를 통해서 가급적 다양한 형태의 자질들을 추출하고 이들에 대한 효과성을 확인해야 한다. 이 부분은 다음 절의 성능 평가에서 세부적으로 설명한다.

본 연구에서 개발된 관계 추출 시스템은 전체적으로 5가지 종류의 자질 집합을 활용한다. 우선 어휘 자질은 관계 추출 대상 개체 주변이나 개체 사이에 존재하는 단어들을 수집하여 지정한다. 주로 관계 설정에 직접적으로 관여할 가능성이 높은 문맥 단어들을 추출하여 자질로 활용한다. 부가적으로 어휘 바이그램(lexical bi-gram)도 부가적으로 추출함으로써 자질의 식별력(discriminant power)을 높이고 있다. 두 번째로 의존 경로 자질(dependency path feature)은 두 개체 간의 의존 구문 분석 결과를 바탕으로 어휘, 품사, 의존 유형(dependency type) 기반의 최단 경로를 추출한 결과이다. 또한 구문 자질로서 두 관계를 포함한 문장에 대한 구문 트리를 바탕으로 말단 노드들을 제외한 경로 포함 트리(Path-Enclosed Tree)를 사용하였다. 의미 자질을 추출하기 위해서 WordNet(Miller 1995)을 이용하여 위에서 기술한 어휘 자질 각각에 대한 부모 신셋(parent synset)을 검색하여 해당 번호를 추출하였다. WordNet에서 부모 신셋에 대한 검색을 위해서는 MIT에서 개발한 JWI 자바 라이브러리[4]를 활용하였다. 마지막으로 개체 자질은 대상 개체 문자열, 유형 등을 활용하였다. 〈표 2〉에 자질의 이름, 종류 및 그에 대한 설명을 기술하였다.

〈표 2〉에서 보는 바와 같이 총 5종류로 구분된 자질들은 설정 파일에서 지정한 방법대로 추출된다. 예를 들어, 설정 파일의 "feature.extraction.method" 항목에 "lex, dep, parse"가 지정되어 있다면, 자질 추출 모듈은 어휘 자질, 의존 경로 자질, 구문 자질 만을 추출하여 학습 인스턴스 혹은 실행 인스턴스를 생성한다. 일반적으로 관계 추출의 성능에 직접적인 영향을 주는 자질로서 개체 자질을 많이 활용한다. 그러나 개체 자질은 특정 학습 집합에 과적합된(overfitted) 학습 모델을 생성하기가 쉬우며, 본 연구에서의 관계 식별 및 관계 분류 실험에서는 개체 자질을 활용하지 않고, 단지 주변 문맥 정보만을 활용하여 성능을 측정하였다. 〈그림 7〉은 실제 추출된 자질의 예시를 보여주고 있다.

〈그림 7〉에서 보는 바와 같이 각 라인의 첫 번째 토큰은 추출된 자질을 나타내고 있다. 특정 자질은 "WORDS-BEFORE-FIRST__exogeneous"에서와 같이 "자질이름__자질값"으로 구성된다. 그 뒤에는 현재 자질에 대한 식별자가 나온다. 위의 예시에서 "DEPS-LEMMA-PATH"는 첫 번째 개체("entityone")와 두 번째 개체("entitytwo") 사이의 어휘 경로를 지정하고 있다 ("entityone 〈 bind 〈 associate 〉 half 〉 entitytwo"). "PATH-ENCLOSED-TREE" 자질은 PennTree Bank 스타일의 중첩 괄호 형태

4) http://projects.csail.mit.edu/jwi/

〈표 2〉 관계 추출을 위한 자질의 종류 및 설명

자질 구분	자질 명칭	세부 설명 및 예시
어휘 자질	WORDS-BETWEEN-NONE	두 개체 사이에 문맥 단어가 존재하지 않는 경우에 1.0으로 지정
	WORDS-BETWEEN-ONE	두 개체 사이에 문맥 단어가 하나만 존재하는 경우에 그 단어를 지정
	WORDS-BETWEEN-FIRST	두 개체 사이에 문맥 단어가 적어도 2개 이상 존재할 경우에 그 첫 번째 단어를 지정
	WORDS-BETWEEN-LAST	두 개체 사이에 문맥 단어가 적어도 2개 이상 존재할 경우에 그 마지막 단어를 지정
	WORDS-BEFORE-FIRST	첫 번째 개체 직전에 나타나는 단어
	WORDS-BEFORE-SECOND	첫 번째 개체 이전 두 번째에 나타나는 단어
	WORDS-AFTER-FIRST	두 번째 개체 직후에 나타나는 단어
	WORDS-AFTER-SECOND	두 번째 개체 이후 두 번째에 나타나는 단어
	WORDS-BETWEEN-BOWS	두 개체 사이에 존재하는 모든 단어들에 대한 Bag-of-Words (BoWs)
	WORDS-BETWEEN-BIGRAM	두 개체 사이에 존재하는 모든 단어들에 대한 바이그램 집합
의존경로 자질	DEPS-TAG-PATH	두 개체 사이의 최단 의존 경로(shortest dependency path)에 존재하는 품사열
	DEPS-LEMMA-PATH	두 개체 사이의 최단 의존 경로(shortest dependency path)에 존재하는 단어열
	DEPS-TYPE-PATH	두 개체 사이의 최단 의존 경로(shortest dependency path)에 존재하는 의존 유형열(dependency type path)
구문 자질	PATH-ENCLOSED-TREE	두 개체 사이의 구문 트리(parse tree) (말단 노드 제외)
의미 자질	HYPER-BETWEEN-ONE	두 개체 사이에 문맥 단어가 하나만 존재하는 경우에 그 단어의 부모 신셋을 지정
	HYPER-BETWEEN-FIRST	두 개체 사이에 문맥 단어가 적어도 2개 이상 존재할 경우에 그 첫 번째 단어의 부모 신셋을 지정
	HYPER-BETWEEN-LAST	두 개체 사이에 문맥 단어가 적어도 2개 이상 존재할 경우에 그 마지막 단어의 부모 신셋을 지정
	HYPER-BEFORE-FIRST	첫 번째 개체 직전에 나타나는 단어의 신셋
	HYPER-BEFORE-SECOND	첫 번째 개체 이전 두 번째에 나타나는 단어의 신셋
	HYPER-AFTER-FIRST	두 번째 개체 직후에 나타나는 단어의 신셋
	HYPER-AFTER-SECOND	두 번째 개체 이후 두 번째에 나타나는 단어의 신셋
	HYPER-BETWEEN-BOWS	두 개체 사이에 존재하는 모든 단어들에 대한 신셋 Bag-of-Words (Synset BoWs)
	HYPER-BETWEEN-BIGRAM	두 개체 사이에 존재하는 모든 단어들에 대한 신셋 바이그램 (synset bigram) 집합
개체 자질	ENTITY-WORD1	첫 번째 개체 문자열
	ENTITY-WORD2	두 번째 개체 문자열
	ENTITY-TYPE1	첫 번째 개체의 타입
	ENTITY-TYPE2	두 번째 개체의 타입
	ENTITY-WORD-PAIR	개체 문자열 바이그램
	ENTITY-TYPE-PAIR	개체 타입 바이그램

```
WORDS-BETWEEN-BIGRAM__1.9-10 8696
ENTITY-WORD1__rhp51 19861
HYPER-BETWEEN-BIGRAM__to-than 13214
DEPS-LEMMA-PATH__entityone < villin > entitytwo 15901
WORDS-BETWEEN-BIGRAM__FLICE-tnf-induced 17718
ENTITY-WORD2__keratin 10413
HYPER-BETWEEN-BIGRAM__SID-03605915-N-SID-02677097-V 5303
WORDS-BETWEEN-BIGRAM__subfragment-1 8715
DEPS-TYPE-PATH__ nn prep_of prep_with nsubj prep_of 7277
WORDS-BETWEEN-BIGRAM__the-unaltered 16648
HYPER-BEFORE-FIRST__SID-03080309-N 3582
DEPS-LEMMA-PATH__entityone < bind < associate > half > entitytwo 13758
PATH-ENCLOSED-TREE__(ROOT (S (NP) (VP (PP (S (ADVP) (VP (NP))))))) 8649
DEPS-LEMMA-PATH__entityone < protein > entitytwo-associated 11341
WORDS-BEFORE-FIRST__exogeneous 14940
HYPER-BETWEEN-BIGRAM__be-together 7513
```

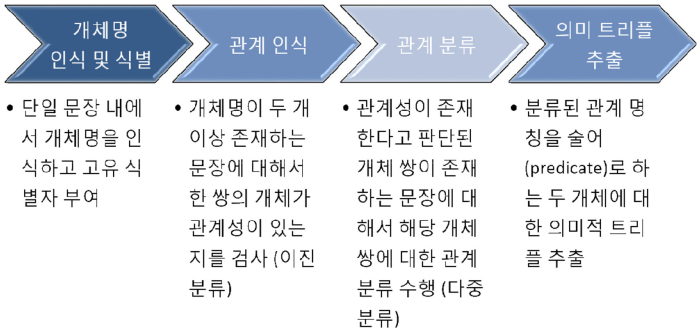
〈그림 7〉 추출된 기계학습 자질 예시

로 변환되어 지정된다. 의미 자질은 단어 대신에 WordNet에서의 그 단어의 부모 신셋 번호(“SID-14944888-N”)가 지정되고, 만일 해당 단어에 대한 부모 신셋이 존재하지 않으면 그 단어가 그대로 지정된다.

4. 실험 및 성능 평가

앞에서도 언급하였듯이 일반적으로 정보 추출(information extraction)의 과정에서 문장 기반 이진 관계 추출(sentence-based binary relation extraction)을 위해서는 〈그림 8〉과 같은 절차가 진행되어야 한다.

특정 문장에 대해서 개체명 인식 및 식별이 완료된 후, 우선적으로 두 개 이상의 개체명이 인식된 문장만을 대상으로 모든 인식된 개체명에 대해 관계 인식(relation identification) 작업이 수행된다. 이 과정에서는 〈그림 8〉에서 보는 바와 같이 한 쌍의 개체명에 대한 관계 존재 여부를 판단하게 된다. 이는 자동 분류(classification)의 관점에서는 이진 분류(binary classification) 문제로 생각할 수 있다. 그 이후에 관계성이 존재한다고 판단된 한 쌍의 개체명에 대해서 세부적으로 관계 분류 작업을 수행한다. 여기서는 실질적으로 어떠한 관계가 존재하는지를 판단하게 되는데 이는 다중 분류 문제(multi-class classification)로 해결한다. 이렇게 관계 추출이 완료된 한 쌍의 개체명은 “개체-관계-개체” 형태의 의미적 트리플로 추출되어 다양한 분야에 활용이 될 수 있다.



〈그림 8〉 개체명 인식에서 관계 추출까지의 세부 프로세스

4.1 관계 식별 성능 평가

4.1.1 학습/평가 컬렉션 및 실험 방법

특정 문장에서 두 개체 간에 관계가 있는지의 유무를 판단하기 위한 관계 식별 시스템을 개발하기 위해 본 연구에서 사용한 학습 집합은 Five PPI Corpora(Pyysalo et al. 2008)이다. 이 학습 집합은 AIMed, BioInfer, HPRD50, IEPA, LLL 등 총 5가지의 단백질 간 연관 관계 식별(Protein-Protein Interaction Extraction) 컬렉션을 결합한 형태로 배포되고 있다. 각 컬렉션은 임의로 선택된 생의학 분야 논문 초록 문장에서 단백질을 식별하고 이들 간의 관계가 명시되어 있는지를 수동으로 판별하여 구축한 데이터이다. 개별 컬렉션에 대한 통계 정보는 〈표 3〉과 같다.

〈표 3〉에서 보듯이 특정 문장에 포함된 단백질 쌍이 서로 연관 관계가 있는 인스턴스의 개수가 4,196건인 반면, 그렇지 않은 문장은 12,884건에 달한다. 따라서 전체적으로는 상당히 불균형성이 높은 컬렉션이라고 볼 수 있다. 이 컬렉션의 모든 단백질 명칭은 익명화되어 있다. 다시 말해서, 기계 학습 시에 개체 자질을 활용할 수 없으며 단지 주변 문맥 자질만을 활용할 수 있다는 것이다. 원래 BioInfer와 같은 컬렉션은 단백질뿐만 아니라 다른 개체 유형도 태깅되어 있으며 이들 간의 관계가 세부적으로 명시되어 있다. 이를 관계 식별 말뭉치로 활용하기 위해서 이진화(binalization)를 수행한 버전이 포함되어 있다.

4.1.2 실험 결과 및 분석

성능 평가 실험은 전체 데이터를 대상으로 10-겹 교차 검증을 통해서 수행되었다. 개체 자질

〈표 3〉 Five PPI Corpora 통계 정보

	AIMed	BioInfer	HPRD50	IEPA	LLL	Total
#Sentence	1,955	1,100	145	486	77	3,763
#Positive Instance	1,000	2,534	163	335	164	4,196
#Negative Instance	4,834	7,132	270	482	166	12,884

을 제외하고 앞에서 설명한 모든 자질들을 활용하였으며, 그 결과를 Macro-averaged Recall, Precision, F-measure로 계산하였다. LIBLINEAR의 매개변수 C(Penalty Score)에 따른 성능 수치 변화 과정을 〈표 4〉에 나타내었다.

〈표 4〉에서 보는 바와 같이 LIBLINEAR의 매개변수에 따른 성능 변화가 거의 없으며, 정확률(Macro-Averaged Precision) 기준으로 최대값은 0.8170(C=0.0313), 평균치는 0.7988을 보이고 있다. 또한 F-스코어 기준으로는 최대치가 0.7464를 나타내었다. 정확도와 재현율이 유사한 분포를 보이는 것은 시스템이 이 기준으로 안정적인 성능을 나타내는 것을 의미한다. 더 세부적으로 F-스코어의 최대치인 0.7464를 나타내는 매개변수(C=0.0625)에서의 혼동 행렬(Confusion Matrix)은 〈표 5〉와 같다.

〈표 4〉 관계 식별 실험 결과(Five PPI Corpora 전체, 10겹 교차 검증)

C-value	Accuracy	Precision	Recall	F-measure
0.0313	**0.8170**	**0.7694**	0.7267	0.7428
0.0625	0.8119	0.7568	**0.7382**	**0.7464**
0.125	0.8096	0.7550	0.7275	0.7388
0.25	0.8036	0.7446	0.7411	0.7428
0.5	0.8006	0.7407	0.7315	0.7358
1	0.7925	0.7304	0.7312	0.7308
2	0.7982	0.7375	0.7352	0.7363
4	0.8006	0.7406	0.7362	0.7383
8	0.7934	0.7311	0.7252	0.7280
16	0.7921	0.7299	0.7329	0.7313
32	0.7966	0.7353	0.7247	0.7296
64	0.7911	0.7279	0.7207	0.7241
128	0.7999	0.7398	0.7299	0.7345
256	0.7909	0.7278	0.7230	0.7253
512	0.8030	0.7440	0.7313	0.7371
1024	0.7901	0.7276	0.7319	0.7297
2048	0.8025	0.7436	0.7292	0.7357
4096	0.7935	0.7315	0.7320	0.7317
8192	0.8010	0.7413	0.7343	0.7376
16384	0.7979	0.7370	0.7246	0.7302
32768	0.7887	0.7255	0.7273	0.7264
평균	**0.7988**	**0.7389**	**0.7302**	**0.7340**

〈표 5〉 관계 식별 실험 결과(혼동 행렬: Macro-Averaged F1=0.7464) (전체)

예측/정답	FALSE	TRUE
FALSE	10177	1662
TRUE	1239	2342
F-SCORES	**0.8753**	**0.6175**

〈표 5〉 혼동 행렬에서 알 수 있듯이 본 연구에서 개발된 관계 식별 엔진은 관계가 존재하지 않는 인스턴스를 찾아내는 능력이 더 우수함을 보여준다. 다시 말해서 "FALSE"가 정답인 인스턴스를 "FALSE"로 분류하는 능력이 뛰어나다. 이는 향후 관계 분류에서의 오동작을 최소화하고 잘못된 결과가 도출되는 상황을 어느 정도 방지하는데 도움을 줄 수 있다.

부가적으로 AIMed 컬렉션에 대한 성능 측정 결과를 〈표 6〉에 나타내었다. 5개의 컬렉션들 중에서 특히 AIMed 컬렉션에 대해서 별도로 성능을 측정한 이유는 기존의 많은 시스템들이 이 컬렉션을 이용하여 성능 수치를 발표했기 때문에 상호 비교가 가능하다.

일반적으로 전체 데이터를 대상으로 실험한 결과보다는 다소 떨어지는 성능 수치를 나타내고 있다. 그러나 정확률 기준으로는 오히려 전체 컬렉션을 활용한 것보다 더 나은 0.8592를 보이고 있고 최대 F-스코어 기준으로 보아도 0.7367 정도로 전체를 대상으로 했을 때의 0.7464와 비교해서 근소한 차이를 보이고 있다. 그러나 평균적으로는 0.7112를 나타냄으로써 전체보다는 다소 수치가 낮은 것을 알 수 있다(〈표 7〉 참조).

〈표 7〉에서 보듯이 본 연구에서 개발된 관계 식별 기능은 현재까지 개발된 시스템에 비해서

〈표 6〉 관계 식별 실험 결과(AIMed, 10겹 교차 검증)

C-Value	Accuracy	Precision	Recall	F-measure
0.0313	**0.8592**	**0.7767**	0.7033	0.7298
0.0625	0.8560	0.7647	0.7144	0.7345
0.125	0.8446	0.7395	**0.7279**	0.7334
0.25	0.8297	0.7111	0.6835	0.6953
0.5	0.8317	0.7188	0.7221	0.7204
1	0.8365	0.7241	0.6864	0.7017
2	0.8400	0.7315	0.7225	0.7268
4	0.8508	0.7542	0.7028	0.7230
8	0.8030	0.6861	0.7212	0.6997
16	0.8345	0.7210	0.6995	0.7091
32	0.8329	0.7175	0.6883	0.7007
64	0.8333	0.7184	0.6916	0.7031
128	0.8235	0.7001	0.6783	0.6878
256	0.8371	0.7258	0.6790	0.6970
512	0.8261	0.7055	0.6867	0.6951
1024	0.8155	0.6867	0.6721	0.6787
2048	0.8428	0.7364	0.7076	0.7201
4096	0.8349	0.7240	0.7267	0.7253
8192	0.8369	0.7255	0.7040	0.7136
16384	0.8337	0.7191	0.6918	0.7036
32768	0.8578	0.7690	0.7155	0.7367
평균	**0.8362**	**0.7265**	**0.7012**	**0.7112**

〈표 7〉 관계 식별 실험 결과(AIMed 기준으로 타 시스템과의 성능 비교)

	POS	NEG	Precision	Recall	F-measure
Our System	1,000	4,834	**76.90**	71.55	**73.67**
(Choi and Myaeng 2010)	1,000	4,834	72.80	62.10	67.00
(Miwa et al. 2009b)	1,000	4,834	60.00	**71.90**	65.20
(Miwa et al. 2009a)	1,000	4,834	58.70	66.10	61.90
(Miwa et al. 2008)	1,005	4,643	60.40	69.30	61.50
(Miyao et al. 2008)	1,059	4,589	54.90	65.50	59.50
(Giuliano et al. 2006)	-	-	60.90	57.20	59.00
(Airola et al. 2008)	1,000	4,834	52.90	61.80	56.40
(Sætre et al. 2007)	1,068	4,563	64.30	44.10	52.00
(Erkan et al. 2007)	951	4,020	59.60	60.70	60.00
(Bunescu and Mooney 2005)	-	-	65.00	46.40	54.20

월등히 높은 성능 수치를 보이고 있다. 이는 관계 식별 문제가 단순한 LIBLINEAR의 선형 커널을 이용해도 충분히 높은 성능 수준에서 해결이 가능함을 의미하며, 다양한 자질들의 결합 즉, 어휘 자질, 의존 구문 자질, 구문 자질, 의미 자질 등의 결합이 기존의 합성곱 구문 트리 커널(convolutional parse tree kernel)이나 복합 커널(composite kernel) 등과 같은 복잡하고 비효율적인 방법을 사용하는 것보다 더 효과적이라는 것을 보여준다.

4.2 BioText 컬렉션을 이용한 관계 분류 성능 평가

앞에서 언급하였듯이 관계 분류는 관계가 있다고 판단된 한 쌍의 개체명을 가진 문장에 대해서 그 문맥을 살펴보고 어떤 관계를 표현하고 있는지를 분류하는 다중 분류 문제로 귀결될 수 있다. 본 연구에서는 개발된 시스템의 관계 분류 성능을 측정하기 위해서 두 가지 컬렉션 즉 BioText와 BioInfer를 활용한다. 본 절에서는 우선 BioText 컬렉션을 이용한 성능 실험 결과를 보여주고 세부적으로 분석해 본다.

4.2.1 학습/평가 컬렉션 및 실험 방법

BioText 컬렉션은 HIV-1 Human Protein Interaction Database가 제공하는 HIV-1 단백질, 숙주 세포 단백질(host cell protein) 그리고 HIV 및 AIDS 질병과 관련이 있는 질병 유기체의 단백질 간의 상호작용 정보(interaction type information)를 이용하여 이들이 포함된 PubMed 초록 및 본문 내의 텍스트를 추출한 결과이다. 추출된 텍스트는 후처리 수작업 정제 과정을 통해서 초록 단위, 문장 단위로 데이터를 구분하여 최종적인 학습 집합이 구축되었다(Rosario and Hearst 2004). 예를 들어, 위 데이터베이스에 존재하는 "AIP1 binds HIV-1-p6 14519844"라는 정보는 PubMed 식별자 "14519844"를 가지는 논문에 "AIP1"과 "HIV-1-p6" 단백질이 "binds" 상호작용(interaction type)을 가진다는 설명이 존재한다는 뜻이다. 이 데이터베이스에는 총 65종류의 상호작용 정보가 존재하

며, 809개의 단백질에 대해서 총 2,224개의 단백질 쌍이 구축되어 있다. 문제가 되는 부분은 이 데이터베이스가 동일 단백질 쌍에 대해서 여러 개의 상호작용 정보를 부착하고 있다는 점이다. 위에서 예제로 설명한 "AIP1"과 "HIV-1-p6" 단백질은 데이터베이스 내에서 "binds"로 명시되어 있기도 하고, "incorporate"로 지정되어 있기도 하다. 컬렉션을 구축할 때, Rosario and Hearst(2004)는 이들을 모두 컬렉션 내에 포함시켰다. 결론적으로 관계 분류 시스템이 분류를 수행할 때 많은 혼돈이 발생할 수 있다. 위 논문에서는 반자동으로 구축된 전체 컬렉션에서 일부 상호작용(10개)에 대해서만 실험을 수행하였으나, 본 연구에서는 개발된 시스템의 성능을 보다 심층적으로 분석하기 위해서 총 22종류의 상호작용 유형을 기반으로 실험을 수행하였다. 이와 관련하여 본 연구에서 사용한 BioText 컬렉션 내의 각 상호작용 별 학습 인스턴스의 규모와 실험에 적용 여부는 〈표 8〉과 같다.

본 연구에서의 성능 실험에서 배제된 상호작

〈표 8〉 BioText 컬렉션의 통계 정보 및 실험 적용 여부

상호작용 유형	인스턴스 개수	적용 여부 (Rosario and Hearst 2004)	적용 여부 (Our System)
binds	422	○	○
requires	393	○	○
upregulates	217	○	○
synergizes with	187	○	○
stimulates	167	○	○
inhibits	162	○	○
interacts with	162	○	○
inactivates	160	○	○
suppresses	150	○	○
downregulates	124	X	○
regulates	124	X	○
activates	123	X	○
degrades	123	○	○
competes with	80	X	○
complexes with	75	X	○
induces	72	X	○
incorporates	68	X	○
modulates	63	X	○
phosphorylates	61	X	○
enhances	54	X	○
co-localizes with	40	X	○
recruits	37	X	○
stabilizes	15	X	X
ubiquitinated by	14	X	X
myristoylated by	2	X	X
합계	3,095	2,143	3,064

용은 3가지(“stabilizes”, “ubiquitinated by”, “myristoylated by”)이다. 배제의 기준으로는 일단 구축된 인스턴스의 수가 너무 적으며, 단백질 간 상호작용이라는 관점에서 그 중요도가 떨어진다고 판단했기 때문이다. 총 3,064개의 인스턴스에 대해서 10-겹 교차 검증을 수행했으며, 그 결과를 다음 절에 상세히 기술하였다.

한 가지 주지할 점은 BioText 컬렉션은 반자동으로 구축된 컬렉션이므로 상호작용으로 맺어지는 개체명(단백질명)의 종류가 한정되어 있다. 따라서 실험에 개체 자질을 활용하면 이 자질 정보만으로도 높은 성능을 보이는 왜곡된 실험 결과가 도출될 수 있다. 따라서 본 실험에서는 개체 자질을 활용하지 않고 단지 문맥 자질 만을 활용하여 실험을 수행하였다. 더불어, 비록 본 컬렉션이 개체 유형 중에서 단백질에만 해당하는 상호작용을 지정하였고 대상 분야도 HIV-1으로 한정시켰기 때문에, 다른 개체 유형이나 분야에 적용하기가 어렵다고 여겨질 수 있으나, 아래에서 수행한 BioInfer에서 보듯이 다른 개체 유형이나 분야에서 사용하는 관계들도 서로 비슷한 형태를 나타내고 있으므로 일반화 강도 및 확장성 측면에서 본 시스템의 성능을 파악하는데 나름대로의 의미가 있다고 볼 수 있다.

4.2.2 실험 결과 및 분석

성능 평가 실험은 위에서 선택된 총 3,064개의 인스턴스 데이터 전체를 대상으로 10-겹 교차 검증을 통해서 수행되었다. 개체 자질을 제외하고 앞에서 설명한 모든 자질들을 활용하였으며, 그 결과를 Macro-averaged Recall, Precision, F-measure로 계산하였다. LIBLINEAR의 매개변수 C(Penalty Score)에 따른 성능 수치 변화 과정을 〈표 9〉에 나타내었다.

〈표 9〉에서 보듯이 C값이 1024일 경우 F-스코어가 0.5712로 기존의 관계 식별 성능보다 매우 낮은 것으로 나타났다. 일단 그 이유가 앞에서 지적한 바와 같이 동일한 관계 인스턴스에 대해서 두 가지 이상의 상호작용 분류가 지정되어 있는 경우가 많은 관계로 학습 모델이 일관된 분석을 하지 못하는 것으로 예상된다. Accuracy는 0.6074로 나타났으며 평균 F-스코어는 0.5572가 도출되었다.

세부적으로 살펴보면, 성능 측정 결과 각 상호 작용 종류별 성능의 낙폭이 매우 크게 나타났다. “COMPLETE”에 대한 성능이 0.370이고 “COMPLEX”를 분류하는 성능은 그보다 낮은 0.295 정도밖에 되지 않는다. 그 반면에 “INACTIVATE”S는 0.807을, “PHOSPHORYLATE”는 0.835로 가장 높은 성능을 보이고 있다. “COMPLEX” 상호작용에 대한 성능이 낮은 이유는 이 분류에 포함된 인스턴스에 대해서 분류기가 “BIND” 상호작용과 혼동하는 경우가 많기 때문으로 나타났다. 또한 인스턴스의 개수가 많은 “BIND” 상호작용에 대한 분류도 “INTERACT”, “REGULATE”, “STIMULATE” 등과 같은 상호작용과의 혼동 현상이 많이 발생하고 있다. 앞에서 지적한 바와 같이 반자동으로 구축된 컬렉션이라는 이유로 설정된 상호작용의 일관성이 낮은 부분이 위와 같은 결과를 초래한다고 생각할 수 있다. 일반적으로 기계 학습 모델은 특정 분야에서 일관되게 태깅된 인스턴스의 규모가 크면 클수록 그 분야에 해당하는 성능이 높게 나타나지만, 위 실험에서 보듯이 각 상호작용별 성능은 그 상호작용에 해당하는 인스턴스의 수와는 크게 관련이 없는 것으로 나타났다.

〈표 9〉 관계 분류 실험 결과(BioText, 22개 관계분류, 10-겹 교차 검증)

C-Value	Accuracy	Precision	Recall	F-measure
0.0313	0.5943	0.6250	0.5178	0.5530
0.0625	0.6034	**0.6311**	0.5308	0.5639
0.125	0.6000	0.6198	0.5349	0.5624
0.25	0.5987	0.6076	0.5288	0.5540
0.5	0.5916	0.6047	0.5305	0.5562
1	0.5970	0.5950	0.5285	0.5533
2	0.6013	0.6039	0.5333	0.5579
4	0.5923	0.5958	0.5210	0.5460
8	0.5940	0.5984	0.5287	0.5530
16	0.5973	0.5969	0.5290	0.5533
32	0.5973	0.6056	0.5277	0.5546
64	0.6034	0.6047	0.5365	0.5605
128	0.6023	0.6011	0.5307	0.5536
256	0.5956	0.6032	0.5268	0.5541
512	0.5963	0.5904	0.5294	0.5512
1024	**0.6074**	0.6191	0.5438	**0.5712**
2048	0.6023	0.6032	0.5402	0.5631
4096	0.6074	0.6165	0.5386	0.5645
8192	0.6007	0.6074	0.5284	0.5564
16384	0.6034	0.5885	**0.5415**	0.5588
32768	0.5993	0.5978	0.5379	0.5595
평균	**0.5993**	**0.6055**	**0.5317**	**0.5572**

〈표 10〉은 Rosario and Hearst(2004)에서 수행했던 실험 대상과 동일한 10종의 상호작용을 대상으로 본 연구에서 개발된 시스템에 대한 성능 평가를 수행한 결과이다.

〈표 10〉에서 보는 바와 같이, 이전 실험과는 달리 성능이 다소 높아졌음을 알 수 있다. 평균 F-스코어는 0.6750이며 정확률(Macro-averaged Precision)은 0.6923이 나오고 있다. 최고 F-스코어 0.6820에 대한 혼동 행렬은 〈표 11〉과 같다.

가장 낮은 성능을 보이는 상호작용은 "INTERACT"로서 0.466을 보이고 있다. 주지할 만한 점은 "INTERACT"는 주로 "BIND", "REQUIRE" 등과 혼동이 많이 생긴다는 점이다. 이는 앞에서 언급한 동일한 단백질 쌍에 대해서 경우에 따라 서로 다른 상호작용이 지정된 이 컬렉션의 특성과도 부합한다. 특히 "INTERACT"는 다른 상호작용 유형과는 달리 다소 일반적인 유형이므로 문장에서의 상호작용에 대한 표현 과정에 있어서 모호한 경우가 많이 발생한다. Rosario and Hearst(2004)에서 실험한 성능 수치와 본 연구에서 수행한 성능 수치를 비교한 내용은 〈표 12〉와 같다.

위 논문에서 활용한 3가지 시스템에 비해서 본 연구에서 개발된 시스템이 월등히 높은 성능을 보여주고 있다. 특히 구축된 컬렉션을 최대한 효과적으로 처리하기 위해서 Rosario and Hearst

〈표 10〉 관계 분류 실험 결과(BioText, 10개 관계분류, 10-겹 교차 검증)

C-Value	Accuracy	Precision	Recall	F-measure
0.0313	0.6914	0.6974	0.6582	0.6738
0.0625	0.6957	0.6931	0.6621	0.6748
0.125	0.6928	0.6897	0.6643	0.6746
0.25	0.6928	0.6919	0.6629	0.6747
0.5	0.6852	0.6798	0.6584	0.6669
1	0.6919	0.6916	0.6671	0.6773
2	0.6981	0.6969	0.6712	**0.6820**
4	0.6842	0.6781	0.6537	0.6637
8	0.6995	0.6999	0.6658	0.6796
16	0.6928	0.6916	0.6653	0.6757
32	0.6876	0.6831	0.6582	0.6686
64	0.6943	0.6924	0.6667	0.6775
128	0.6904	0.6905	0.6655	0.6761
256	0.6813	0.6692	0.6552	0.6605
512	0.6952	0.6898	0.6709	0.6793
1024	0.6914	0.6918	0.6642	0.6758
2048	**0.6990**	**0.6979**	0.6692	0.6808
4096	0.6919	0.6967	0.6656	0.6787
8192	0.6947	0.6914	0.6706	0.6797
16384	0.6890	0.6880	0.6643	0.6737
32768	0.6986	0.6921	**0.6739**	0.6814
평균	**0.6923**	**0.6901**	**0.6644**	**0.6750**

〈표 11〉 관계 분류 혼동 행렬(BioText, 10개 관계, F-score: 0.6820)

P/A	BND.	DEG.	INA.	INH.	INT.	REQ.	STI.	SUP.	SYN.	UPR.
BIND	296	12	5	29	40	18	13	21	12	15
DEGRADE	6	80	1	0	2	1	0	3	3	4
INACTIVATE	0	0	122	7	5	1	0	1	1	4
INHIBIT	14	1	5	81	6	3	5	7	6	5
INTERACT	23	1	5	6	66	6	7	5	8	2
REQUIRE	22	4	2	14	18	320	29	5	3	5
STIMULATE	16	4	3	1	7	20	97	2	9	1
SUPPRESS	7	9	4	7	2	7	0	100	2	2
SYNERGIZE	12	1	3	7	4	7	13	2	124	4
UPREGULATE	14	8	3	7	4	4	2	3	9	173
F-SCORES	**0.680**	**0.727**	**0.830**	**0.555**	**0.466**	**0.791**	**0.595**	**0.692**	**0.701**	**0.783**

〈표 12〉 타 시스템과의 성능 비교(BioText, 10개 상호작용 대상)

비교 시스템		Accuracy
(Rosario and Hearst 2004)	Dynamic Model	60.5
	Naive Bayes	59.7
	Neural Network	51.6
개발된 시스템	Linear Kernel SVM	**69.9**

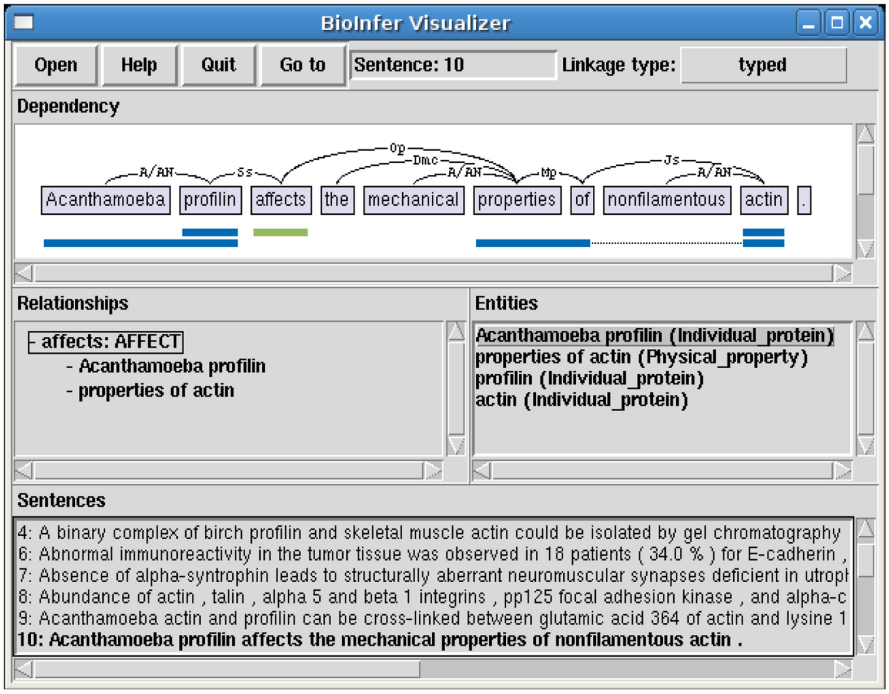
(2004)에서 새롭게 제안된 모델인 "Dynamic Model"보다 9.4%의 성능 향상을 이루었다. 비록 컬렉션 자체의 문제점으로 인해서 이 집합을 활용한 사후 연구가 많이 진행되지는 않았으나 본 시스템의 성능적 우월성을 보여주기에는 충분하다고 사료된다.

4.3 BioInfer 컬렉션을 이용한 관계 분류 성능 평가

4.3.1 학습/평가 컬렉션 및 실험 방법

이 절에서는 BioInfer(Pyysalo et al. 2007) 컬렉션을 활용한 시스템 성능 평가를 수행한다. BioInfer는 다양한 유형의 개체명이 다수 포함되어 있는 1,100 문장을 대상으로 수동으로 개체명 간의 관계를 부착한 관계 추출 컬렉션이다. 특히 이 컬렉션은 앞에서 설명한 관계 식별에서 사용한 이진화 된 BioInfer(Binalized BioInfer)의 원본 버전이다. 특히 단백질명은 물론 질병, 단백질 콤플렉스, 유전자 등의 다양한 개체명이 식별되어 있으며, 이들 간의 관계까지도 명시되어 있기 때문에 실용적인 관계 분류 성능 실험에 유용하다. 〈그림 9〉는 BioInfer에서 제공하고 있는 가시화 도구를 보여준다.

〈그림 9〉에서 문장 내에는 4개의 개체가 존재하고, 이들 중 "Acanthamoeba profilin"과 "properties of actin" 사이에는 "AFFECT" 관계가 있음을 보여준다. 본 연구에서 실험을 위해서 전처리 과정을 통해서 정제되고 변환된 컬렉션에는 "Individual Protein", "Gene", "DNA_



〈그림 9〉 BioInfer 가시화 도구 및 내용

family_or_group", "Protein_complex", "Protein_family_or_group"과 같은 총 5가지의 개체명 유형이 존재한다. 〈표 13〉은 실험에 사용된 관계 종류 및 인스턴스 수를 나타내고 있다. 〈그림 9〉에서 보는 바와 같이 특정 문장에는 많은 종류의 개체가 존재하고 그들 간의 관계 또한 다양하게 출현하기 때문에, 동일한 문장에서 많은 종류의 인스턴스가 도출될 수 있다. 이를 바탕으로 심층적인 관계 분류 실험을 진행하고 그 결과를 세밀하게 분석할 수 있다.

〈표 13〉 BioInfer 통계 정보(총 19개 관계)

관계명	인스턴스 개수
BIND	616
ASSEMBLY	46
ACTIVATE	50
MODIFY	38
RELATE	86
PHOSPHORYLATE	40
MEMBER	280
OTHERS	364
INTERACT	159
ATTACH	33
ASSEMBLE	33
CONTAIN	42
CAUSAL	55
LOCALIZE	72
CHANGE	73
SIMILAR	59
COLOCALIZE	112
INHIBIT	32
ENCODE	32
합계	2222

앞에서 실험한 BioText와 마찬가지로 개별 관계 분류 간의 인스턴스 개수의 편차가 매우 심한 편이므로 성능을 높이기 쉬운 컬렉션은 아니다. 한 예로 "BIND"에 해당하는 인스턴스의 수는 616개인 반면 "INHIBIT"에 대한 인스턴스의 수는 고작 32개 밖에 되지 않는다. 또한 인스턴스의 수가 매우 적은 나머지 분류들은 "OTHERS"로 통합함으로써 세부적인 관계는 아니더라도 임의의 관계가 있음을 찾아낼 수도 있는 분류 시스템을 구성하기 위해 노력하였다. 실제로 BioText에서 구축된 상호작용 집합들과 많은 부분이 겹치고 있다("LOCALIZE", "INHIBIT", "INTERACT", "BIND" 등). 따라서 단백질 간 상호작용(Protein-Protein Interaction)이나 타 개체 간의 관계들도 텍스트에서의 그 표현에 있어서는 서로 공유하는 부분이 많다고 볼 수 있다.

4.3.2 실험 결과 및 분석

성능 평가 실험은 위에서 선택된 총 2,222개의 인스턴스 데이터 전체를 대상으로 10-겹 교차 검증을 통해서 수행되었다. 개체 자질을 포함한 모든 자질들을 활용하였으며, 그 결과를 Accuracy, Micro-averaged Recall, Precision, F-measure로 계산하였다. LIBLINEAR의 매개변수 C(Penalty Score)에 따른 성능 수치 변화 과정을 〈표 14〉에 나타내었다.

19개의 관계를 대상으로 관계 분류를 수행한 결과, Precision의 최대값은 0.8158로 매우 높은 편이며, F-스코어의 최대값은 0.7242이다. 평균적으로 0.7018의 스코어를 나타내고 있으며 Macro-Averaged Precision(Accuracy)의 평균값은 0.7344로서 비교적 우수한 편이다. 선형 커널 기반의 SVM의 매개변수인 penalty score의 변동에 따른 성능 변화가 비교적 적은 편이므로, 활용된 다양한 자질의 효과적인 측면이 부각된다고 볼 수 있다.

〈표 14〉 관계 분류 실험 결과(BioInfer, 19개 관계 분류, 10-겹 교차 검증)

C-Value	Accuracy	Precision	Recall	F-measure
0.0313	0.7305	**0.8158**	0.6279	0.6895
0.0625	0.7341	0.7926	0.6363	0.6896
0.125	0.7377	0.7959	0.6555	0.7067
0.25	0.7355	0.7738	0.6634	0.7038
0.5	**0.7532**	0.7992	**0.6789**	**0.7242**
1	0.7459	0.7848	0.6711	0.7149
2	0.7345	0.7720	0.6559	0.6991
4	0.7318	0.7472	0.6636	0.6958
8	0.7314	0.7564	0.6610	0.6973
16	0.7245	0.7872	0.6336	0.6908
32	0.7414	0.7629	0.6741	0.7090
64	0.7355	0.7922	0.6616	0.7108
128	0.7355	0.7750	0.6640	0.7053
256	0.7227	0.7977	0.6431	0.6988
512	0.7291	0.7537	0.6582	0.6939
1024	0.7323	0.7752	0.6540	0.6991
2048	0.7291	0.7524	0.6577	0.6935
4096	0.7459	0.7860	0.6734	0.7158
8192	0.7332	0.7819	0.6509	0.7000
16384	0.7282	0.7750	0.6515	0.6962
32768	0.7309	0.7797	0.6573	0.7043
평균	**0.7344**	**0.7789**	**0.6568**	**0.7018**

관계 분류별 성능을 세부적으로 살펴보면 역시 마찬가지로 각 분류별 성능의 차이가 극명히 드러난다. 특히 가장 높은 성능을 보이는 "MEMBER"와 가장 낮은 성능을 보이는 "ASSEMBLE" 간의 점수 차이는 거의 0.45 차이가 난다. "OTHERS" 관계는 "BIND" 관계와 가장 혼동되고 있고, "INTERACT" 관계 역시 마찬가지이다. 주지할 만한 사실은 "OTHERS" 관계는 다양한 관계와 혼동 현상이 나타난다는 점이다. 이 분류는 앞에서 지적하였듯이 소수의 인스턴스를 가지는 다양한 관계를 하나로 결합해 놓은 가공의 관계라는 점에서 당연한 현상이라 판단된다. 전체적으로는 관계 의미가 유사한 인스턴스에 대한 분류 성능이 아직까지 높지 않음을 보여주며, 이를 위해서 학습 집합의 추가 구축은 물론 부가적인 자질의 적용 및 기계 학습 모델의 확장 등이 필요할 것이다. 또한 외부 응용에 필요한 관계의 명확한 설정이 필요하며 그에 따른 관계 분류 학습 집합의 지속적인 구축이 필요하다.

비록 BioInfer 컬렉션이 바이오 분야에서의 관계 분류에 유용한 학습 집합이긴 하지만, 현재까지는 대부분 관계 식별(Relation Identification)에만 집중되어 연구되었기 때문에 비교 대상 시스템이 존재하지 않는다. 본 연구에서 최초로 BioInfer를 이용한 관계 분류 시스템을 구축하

였으며 이에 대한 세부적인 성능 측정 결과를 도출시켰다. 또한 본 연구에서 적용한 총 5가지의 자질 종류에 대한 다양한 결합적 성능 측정 결과에서는 모든 자질을 다 적용하였을 경우 가장 좋은 성능을 보였다.

5. 결론 및 향후 연구 방향

본 연구에서는 바이오 분야 학술 정보에서 다양한 개체명 간 관계 추출을 위한 심층적 실험 연구를 수행하였다. 이를 위해서 바이오 분야에 특화된 관계 추출 시스템을 제안하고 3가지 컬렉션을 활용한 성능 측정 실험을 수행하였다. 그 결과 전체적으로 높은 성능을 나타내고 있으며 일부 컬렉션(AIMed)에서는 세계적인 수준보다 높은 성능을 보였다. 그러나 본 연구에서 도출된 성과를 지속적으로 발전시키기 위해서는 다음과 같은 다양한 후속 연구가 반드시 필요하다.

비록 현재까지 개발된 SVM 모델 중에서 본 연구에 가장 적합한 선형 커널 기반 SVM을 활용하여 성능을 극대화시켰으나 보다 다양한 기계 학습 모델 적용이 필요하다. 물론 개발된 시스템이 새로운 기계 학습 모델을 추가적으로 채택하고 활용하기에 편리하게 구성되어 있으므로 이러한 부분은 향후 연구로서 실행 가능성이 매우 높다. 특히 현재 Sequence Labeling에 독보적인 성능을 보이고 있는 딥 러닝(Deep Learning) 모델을 적용한 새로운 관계 추출 모델을 개발하고 성능 검증을 할 필요가 있다. 개체명 인식이나 의미역 부착(Semantic Role Labeling) 등과 같은 텍스트 시퀀스에 레이블을 자동으로 부착하는 문제들은 RNN(Recurrent Neural Network)이나 CNN(Convolutional Neural Network) 모델이 현재 가장 좋은 성능을 나타내고 있는 것은 주지할 만한 사항이다. 그러나 문장의 구문적, 의미적 구조를 이해하고 이를 통해서 두 개체명 간의 관계를 식별하고 분류하는 문제에 대한 연구는 아직까지 미흡하다.

추가적으로 학습 집합의 구축이다. 본 연구에서 개발된 체제를 적극적으로 활용한다면, 컬렉션 구축의 가장 어려운 문제인 초기 데이터 수집 문제를 쉽게 처리할 수 있다. 이를 통해서 컬렉션 구축자(분야 전문가)들이 보다 수월하게 데이터를 분석하고 태깅할 수 있는 기반을 만들 수 있다. 문제는 국내에는 이러한 데이터를 지속적으로 구축할 만한 바이오 분야 전문가가 매우 부족하다는 사실인데, 이 부분에 대해서도 정책적이고 기술적인 접근을 지속적으로 시도해야 한다.

참 고 문 헌

[1] Airola, A. et al. 2008. "All-Paths Graph Kernel for Protein-Protein Interaction Extraction with Evaluation of Cross-Corpus Learning." *BMC Bioinformatics*, 9(11): 1-12.

[2] Ananiadou, S., Kell, D. B. and Tsujii, J. 2006. "Text Mining and Its Potential Applications in Systems Biology." *Trends in Biotechnology*, 24(12): 571-579.

[3] Ananiadou, S. et al. 2010. "Event Extraction for Systems Biology by Text Mining the Literature." *Trends in Biotechnology*, 28(7): 381-390.

[4] Andrade, M. A. and Valencia, A. 1998. "Automatic Extraction of Keywords from Scientific Text: Application to the Knowledge Domain of Protein Families." *Bioinformatics*, 14(7): 600-607.

[5] Blaschke, C., Hirschman, L. and Valencia, A. 2002. "Information Extraction in Molecular Biology." *Briefings in Bioinformatics*, 3(2): 154-165.

[6] Bunescu, R. et al. 2005. "Comparative Experiments on Learning Information Extractors for Proteins and Their Interactions." *Artificial Intelligence in Medicine*, 33(2): 139-155.

[7] Chang, C. C. and Lin, C. J. 2011. "LIBSVM: A Library for Support Vector Machines." *ACM Transactions on Intelligent Systems and Technology*, 2(3): 1-39.

[8] Choi, S. P. et al. 2014. "An Intensive Case Study on Kernel-based Relation Extraction." *Multimedia Tools and Applications*, 71(2): 741-767.

[9] Choi, S. P. and Myaeng, S. H. 2010. "Simplicity Is Better: Revisiting Single Kernel PPI Extraction." In *Proceedings of the 23rd International Conference on Computational Linguistics*, August 23rd-27th, 2010, Beijing: Beijing International Convention Center: 206-214.

[10] Craven, M. and Kumlien, J. 1999. "Constructing Biological Knowledge Bases by Extracting Information from Text Sources." In *Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology*, August 6th-10th, 1999, Heidelbrg: Kongresshaus Stadthalle: 77-86.

[11] Fan, R. E. et al. 2008. "LIBLINEAR: A Library for Large Linear Classification." *Journal of Machine Learning Research*, 9: 1871-1874.

[12] Fundel, K., Küffner, R. and Zimmer, R. 2007. "RelEx — Relation Extraction Using Dependency Parse Trees." *Bioinformatics*, 23(3): 365-371.

[13] Li, C., Liakata, M. and Rebholz-Schuhmann, D. 2014. "Biological Network Extraction from Scientific Literature: State of the Art and Challenges." *Briefings in Bioinformatics*, 15(5):

856-877.

[14] Manning, C. D. et al. 2014. "The Stanford CoreNLP Natural Language Processing Toolkit." In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, June 22nd-27th, 2014, Baltimore, MD: 55-60.

[15] Miller, G. A. 1995. "WordNet: A Lexical Database for English." *Communications of the ACM*, 38(11): 39-41.

[16] Miwa, M. et al. 2009. "Protein-Protein Interaction Extraction by Leveraging Multiple Kernels and Parsers." *International Journal of Medical Informatics*, 78(12): e39-e46.

[17] Ono, T. et al. 2001. "Automated Extraction of Information on Protein-Protein Interactions from the Biological Literature." *Bioinformatics*, 17(2): 155-161.

[18] Papanikolaou, N. et al. 2014. "Protein-Protein Interaction Predictions Using Text Mining Methods." *Methods*, 74: 47-53.

[19] *Wikipedia*. 2016. San Francisco, CA: Wikimedia Foundation.. s.v. "Protein-Protein Interaction." [online] 〈https://en.wikipedia.org/w/index.php?title=Protein%E2%80%93protein_interaction&oldid=713402377〉

[20] Pyysalo, S. et al. 2008. "Comparative Analysis of Five Protein-Protein Interaction Corpora." *BMC Bioinformatics*, 9(3): 1-11.

[21] Pyysalo, S. et al. 2007. "BioInfer: A Corpus for Information Extraction in the Biomedical Domain." *BMC Bioinformatics*, 8(1): 50-73.

[22] Rosario, B. and Hearst, M. A. 2004. "Classifying Semantic Relations in Bioscience Texts." In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, July 21st-26th, 2004, Barcelona: Forum Convention Centre: 430-437.

[23] Sekimizu, T., Park, H. S. and Tsujii, J. 1998. "Identifying the Interaction between Genes and Gene Products Based on Frequently Seen Verbs in Medline Abstracts." *Genome Informatics*, 9: 62-71.

[24] Temkin, J. M. and Gilder, M. R. 2003. "Extraction of Protein Interaction Information from Unstructured Text using a Context-Free Grammar." *Bioinformatics*, 19(16): 2046-2053.

[25] Zhou, D. and He, Y. 2008. "Extracting Interactions Between Proteins from the Literature." *Journal of Biomedical Informatics*, 41(2): 393-407.