한국어 질의응답시스템을 위한 지지벡터기계 기반의 질의유형분류기

(A Question Type Classifier based on a Support Vector Machine for a Korean Question-Answering System)

김 학 수 † 안 영 훈 † 서 정 연 ††
(Harksoo Kim) (An, Young Hun) (Jungyun Seo)

요 약 고성능의 질의응답 시스템을 구현하기 위해서는 사용자의 질의 의도를 파악할 수 있는 질의 유형 분류기가 필요하다. 본 논문에서는 지지 벡터 기계(support vector machine, SVM)를 이용한 질의 유형 분류기를 제안한다. 본 논문에서 제안하는 질의 유형 분류기의 분류 과정은 다음과 같다. 우선, 사용자 질의에 포함된 어휘, 품사, 의미표지와 같은 다양한 정보를 이용하여 사용자 질의로부터 자질들을 추출한다. 다량의 자질들 중에서 유용한 것들만을 선택하기 위해서 카이 제곱 통계량을 이용한다. 추출된 자질들은 벡터 공간 모델로 표현되고, 문서 범주화 기법 중 하나인 지지 벡터 기계는 이 정보들을 이용하여 질의 유형을 분류한다. 본 논문에서 제안하는 시스템은 질의 유형 분류 문제에 자동 문서 범주화 기법을 도입하여 86.4%의 높은 분류 정확도를 보였다. 또한 질의 유형 분류기를 통계적 방법으로 구축함으로써 lexico-syntactic 패턴과 같은 규칙을 기술하는 수작업을 배제할 수 있으며, 응용 영역의 변화에 대해서도 안정적인 처리와 빠른 이식성을 보장한다.

키워드 : 질의 유형 분류기, 자동 문서 범주화, 지지 벡터 기계

Abstract To build an efficient Question-Answering (QA) system, a question type classifier is needed. It can classify user's queries into predefined categories regardless of the surface form of a question. In this paper, we propose a question type classifier using a Support Vector Machine (SVM). The question type classifier first extracts features like lexical forms, part of speech and semantic markers from a user's question. The system uses χ^2 statistic to select important features. Selected features are represented as a vector. Finally, a SVM categorizes questions into predefined categories according to the extracted features. In the experiment, the proposed system accomplished 86.4% accuracy. The system precisely classifies question type without using any rules like lexico-syntactic patterns. Therefore, the system is robust and easily portable to other domains.

Key words : Question type classifier, Document categorization, Support Vector Machine

1. 서 론

기존의 정보검색(information retrieval, IR)은 사용자의 질문에 대한 응답으로 대량의 문서를 검색하고 순위화하는데 초점을 맞추어 왔다. 그러나, 많은 사용자들은 명확한 의도를 가지고 질문을 하며, 정보 검색 시스템이 대량의 문서를 찾아주기 보다는 정답들을 곧바로 찾아 제시해 주기를 바란다[1]. 이러한 요구를 만족시키기 위하여 질의응답(question answering, QA)이라는 개념이 출현했으며, 많은 연구들이 AAAI[2]와 TREC[3]을 중심으로 수행되어 왔다.

질의응답 시스템이 정보 검색 시스템과 다른 점 중 하나는 질의 처리 과정(question processing)에 있다. 질의 처리 과정은 질의에서 사용자의 질의 의도를 파악할 수 있는 질의 유형(question type)이나 키워드

· 본 연구는 정보통신부 선도기술 개발사업과 서강대학교 산업기술연구소의 지원으로 이루어진 것임.
† 비 회 원 : 서강대학교 컴퓨터학과 자연어처리 연구실
hskim@diquest.com
cyllian@diquest.com
†† 종신회원 : 서강대학교 컴퓨터학과 교수
seojy@ccs.sogang.ac.kr
논문접수 : 2002년 5월 2일
심사완료 : 2003년 2월 7일

(keyword) 등의 정보를 질의로부터 추출하는 것이다. 특히 질의 유형은 질의응답 시스템이 문서에서 정답이 될 수 있는 정답 후보(answer candidate)들을 추출하는 데 중요한 정보를 제공한다. 최근 TREC(Text REtrieval Contest)에서 소개된 질의응답 시스템들은 대부분 질의 유형 분류(question type classification)를 위한 모듈을 포함하고 있다[4,5,6,7].

질의응답 시스템이 인터넷과 같은 실용적 환경에서 사용될 경우, 실제 사용자의 질의는 다양한 유형으로 나타나게 된다. 따라서 실용적인 시스템에서 사용되는 질의 유형 분류기(question type classifier)는 문장의 형태나 단어의 쓰임에 관계없이 같은 의도를 가진 질의를 같은 유형으로 분류해 낼 수 있어야 한다. 예를 들어 "올해의 프로야구 우승팀은?"과 "어디가 올해 프로야구에서 승리했죠?"는 형태는 다르지만 같은 의도를 가진 질의다. 또한, 시스템이 한 응용 영역(application domain)에서 다른 영역으로 옮겨질 때 응용 영역 간의 이식이 쉽고, 응용 영역의 변화에 따른 성능 차이가 없어야 한다. 이러한 사항들을 만족시키기 위하여 본 논문에서는 지지 벡터 기계(support vector machine, SVM) 을 이용한 질의 유형 분류기를 제안한다. 제안한 시스템은 질의 유형 분류 과정을 통계적 기법을 사용해서 자동화함으로써 다양한 형태의 질의를 강건하게 처리할 수 있고, 응용 영역의 변화에도 유연하게 대처할 수 있다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 질의 유형 분류 분야에 수행되었던 관련 연구들과 지지 벡터 기계에 대해서 살펴보고, 3장에서는 본 논문에서 제안하는 질의 유형 분류기에 대하여 설명한다. 그리고 4장에서 실험 및 평가를 하며, 5장에서는 결론 및 향후 과제를 기술한다.

2. 관련 연구

질의 유형은 사용자의 질의 의도를 특정한 범주(category)에 할당하는 것으로 질의응답 시스템 연구의 한 분야로 진행되어 왔다[4,5,6,7,8,9,10,11].

기존의 질의 유형 분류 기법은 규칙에 기반한 방법(rule-based method)[4,5,8] [12]과 통계에 기반한 방법(statistical method)[13,14,15]으로 나뉘어 진다. 규칙에 기반한 질의 유형 분류를 채택하고 있는 시스템들은 일반적으로 lexico-syntactic 패턴을 구축하고, 이러한 패턴을 유한 상태 오토마타와 매치(match)하여 질의 유형을 분류한다. MURAX[8]는 규칙에 기반한 질의 유형 분류를 사용하는 대표적인 질의응답 시스템으로, 대부분의 규칙에 기반한 시스템은 MURAX와 유사한 질의 처리 방법을 사용한다. LASSO[4]는 MURAX에 비해 다소 복잡한 방법으로 질의를 처리한다. LASSO는 질의 처리 과정에서 질의 유형과 함께 초점 단어(focus word)와 키워드를 추출한다. 일반적으로 규칙에 기반한 접근 방법을 채택한 질의응답 시스템들은 다음과 같은 장점을 가지고 있다.

- 질의 유형 분류 과정이 유한 상태 오토마타로 구현되므로 사용자의 질의에 대해서 즉각적으로 질의 유형을 분류해 낼 수 있다.
- 수동으로 기술된 규칙에 따라 질의 유형을 분류하므로 질의 유형을 잘못 분류해서 전혀 관계없는 엉뚱한 대답을 하는 경우를 방지할 수 있다.
- 응용 영역이 정해져 있을 경우 간단한 튜닝(tuning)으로 성능을 향상시킬 수 있다.

그러나 규칙에 기반한 질의 유형 분류 방법은 규칙을 수정하기 위해서 전문적인 지식을 가진 사람들의 노력이 필요하고, 규칙과 일치되지 않는 질의가 들어 왔을 때는 질의 유형을 분류할 수 없는 문제점을 가지고 있다. 그리고 규칙이 많아질수록 좋은 성능을 내기 위한 튜닝이 점점 더 어려워지게 된다. 또한 시스템이 다른 응용 영역에서 사용될 경우에는 기존의 규칙들을 모두 수정하거나 재작성 해야 하는 문제점이 있다.

통계적 방법에 기반한 질의 유형 분류는 수동으로 분류된 대량의 학습 데이타로부터 추출한 통계 정보를 이용한다. Ittycheriah는 [14]에서 질의 유형 분류를 위해서 최대 엔트로피 모델(maximum entropy model)을 이용하였다. 최대 엔트로피 모델을 위한 자질로는 단어의 n-그램(n-gram), 질의에서 나타난 단어들의 워드넷(WordNet)[16] 상에서의 상위어(hypernym), 질의에서의 단어의 위치 등을 이용한다. Hermjakob는 [12]에서 질의 유형 분류를 위해서 결정적 LR 파서(deterministic shift-reduce parser)인 Contex[17]를 확장하였다. 확장된 Contex는 Qtargets(질의 유형)로 분류되어 있는 Penn Treebank를 이용해 학습되며, 25만 여 개의 문장에 대해 성공적으로 구문 분석을 수행하였다. Mann은 [15]에서 가공되지 않은 데이타(unannotated data)를 이용해서 질의 유형을 분류하는 방법을 제안하였다. 이 방법은 상호 정보 척도(mutual information)에 기반하여 질의 데이타를 학습한다. 통계에 기반한 질의 유형 분류는 다음과 같은 장점이 있다.

- 대량의 학습 데이타를 이용한 통계 모델을 사용하기 때문에 안정적으로 질의의 유형을 분류할 수 있으며, 응용 영역의 변화에 대해 크게 영향을 받지 않는다.
- 자동화된 통계적 방법을 사용함으로써 시스템 구축을

쉽게 할 수 있다.

그러나 통계적 접근 방식은 가끔 사용자가 질의에서 의도하지 않은 결과를 정답으로 출력하는 경우가 있다. 예를 들어 "작년 프로야구는 누가 우승했나요?"는 우승팀을, "최근 국제 마라톤 대회에서 누가 우승했나요?"는 우승한 사람을 정답으로 요구하는 질의다. 하지만 두 질의는 구조적으로 매우 유사하므로 질의 유형을 제대로 분리하기 어렵다. 규칙에 기반한 시스템은 이러한 문제를 보완할 수 있는 규칙을 쉽게 수정하거나 추가할 수 있지만, 통계적 방법의 경우에는 보완이 쉽지 않다.

지지 벡터 기계는 두 개의 범주를 구분하는 문제를 해결하기 위해 1995년에 Vapnik에 의해 소개된 비교적 최근의 학습 기법으로 두 개의 클래스의 구성 데이타를 가장 잘 분리해 낼 수 있는 결정면(decision surface)을 찾는 모델이다[18]. 이러한 지지 벡터 기계는 선형 분리 문제(linear separable problem)에 사용되는 알고리즘이지만, 다차원의 부드러운 곡선을 이용하여 경계면을 설정하거나, 실제 데이타 벡터를 새로운 자질을 포함한 새로운 벡터 공간에 매핑(mapping)하는 방법을 통하여 비선형 분리 문제도 해결할 수 있다.

3. 질의 유형 분류기

본 논문에서 제안한 질의 유형 분류기는 크게 질의 학습 과정과 질의 분류 과정으로 나뉘어 진다. [그림 1]은 질의 유형 분류기의 전체 구성도이다.

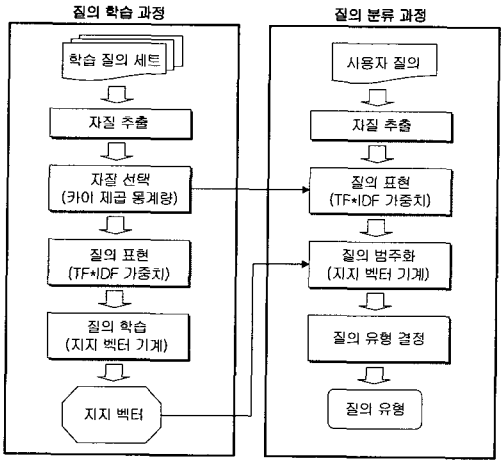

그림 1 질의 유형 분류기의 전체 구성도

질의 학습 과정에서는 먼저 학습 질의들로부터 자질들을 추출한다. 이 때 자질 추출을 위해서 형태소 분석기, 개체명 인식기, 슬라이딩 윈도(sliding window)를 사용한다. 자질 선택 과정은 추출된 자질들을 카이 제곱 통계량을 이용하여 정보량에 따라 순위화하는데, 정보량의 순위는 질의 표현 과정에서 자질의 수를 제한하는데 사용된다. 선택된 자질들은 TF·IDF 가중치를 통해서 질의 학습을 위한 벡터로 표현된다. 마지막으로 질의 학습 과정은 지지 벡터 기계를 이용해서 질의 유형에 대한 지지 벡터들을 만든다.

질의 분류 과정은 우선 질의 학습 과정과 같은 방법으로 입력된 질의에서 자질들을 추출한다. 추출된 자질들은 TF·IDF 가중치를 이용해서 벡터로 표현되는데, 이 때 자질 선택 과정에서 선택되지 않은 자질들은 중요한 정보를 가지지 않는다고 판단하여 이용하지 않는다. 질의 범주화 과정은 학습 과정에서 생성된 지지 벡터들을 이용해서 질의가 지지 벡터 상의 어느 범주에 속하는지 판별한다.

본 장에서는 우선 실험에 사용된 질의 유형 분류 체계와 개체명 사전(named entity dictionary)에 대해서 설명하고, 질의 유형 분류기의 각 부분에 대해서 자세히 설명한다.

3.1 질의 유형과 개체명 사전

본 논문에서는 사용자의 질의 유형을 105개의 의미 범주(semantic category)로 구분하고, 그것에 따라 질의 유형을 분류한다. [표 1]에서 보듯이 105개의 의미 범주는 2개의 계층으로 이루어진다.

표 1 의미 범주의 일부

The first layer	The second layer		
animal	bird	fish	mammal
	person	reptile	
location	address	building	city
	continent	country	state
	town		
date	day	month	season
	weekday	year	
time	hour	minute	second
organization	company	department	family
	group	laboratory	school
	team		

첫 번째 계층에 속한 의미 범주들은 두 번째 계층에 속한 것들보다 넓은 의미를 지닌다. 본 논문에서는 105개의 의미 범주를 결정하기 위하여 TREC에 참가한 질의응답 시스템들의 의미 범주를 참고하였고, 상업용 정보 검색 시스템[19]에서 수집한 질의 로그(log)를 분석

하였다.

질의 유형 분류기는 질의에서 유용한 자질들을 추출하기 위해서 개체명 인식기를 사용한다. 개체명 인식기는 PLO 사전이라 불리는 개체명 사전을 사용해서 질의에 나타난 개체명들을 인식한다. PLO 사전은 표제어와 그에 해당하는 의미 표지로 구성되며 다음과 같은 네 종류의 엔트리를 가진다.

- 고유 명사: 인명, 국가명, 도시명, 기관명 등
- 일반 명사: 직책, 직위, 취미 등
- 단위 명사: km, m, cm, kg, g, mg 등
- 기타: 질의응답 시스템에 필요한 특수한 단어

고유 명사 엔트리를 구축하기 위해서는 웹으로부터 얻어진 온라인 전화번호부, 상호 등을 정제하여 수동으로 구축하였다. 일반 명사 엔트리를 구축하기 위해서는 각 의미 범주에 속하는 소수의 단어들을 임의로 선정하고 그 단어들의 유의어들을 찾아 확장하는 방법을 이용하였다. 최종적으로 구축된 PLO 사전은 477,596개의 엔트리로 구성된다.

3.2 자질 추출

자질 추출 과정은 입력된 질의에서 범주화를 위해 필요한 자질들을 추출한다.

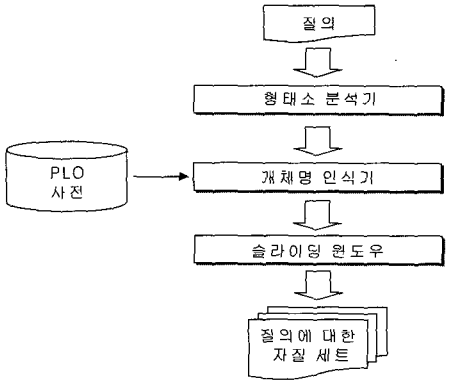

그림 2 자질 추출 과정

[그림 2]는 자질 추출 과정을 도식화한 것이다. 자질 추출 과정은 우선 형태소 분석기를 이용해서 질의에서 형태소를 추출하고 품사를 결정한다. 개체명 인식기는 분석된 형태소 중 PLO 사전에 존재하는 엔트리에 대해서 의미 표지를 할당한다. 의미 표지는 %와 @으로 구분되는데, %는 해당 형태소가 해당 의미 표지와 비슷한 의미 관계라는 뜻이고, @는 해당 형태소가 해당 의미 표지에 포함되는 관계라는 뜻이다. [표 2]는 "서강대학교의 총장실 전화번호는?" 에서 분석된 형태소, 품사, 의미 표지를 보여 준다.

표 2 형태소 분석기와 개체명 인식기의 분석 결과 예제

형태소	품사	의미 표지
서강대학교	nq_loc	(@location|@organization)
의	j	none
총장실	ncn	@location
전화번호	ncn	%tel_num
는	j	none
?	sf	none

[표 2]에서 none은 해당 형태소에 대한 개체명 인식기의 결과(의미 표지)가 없음을 의미한다. '서강대학교'에 해당하는 의미 표지는 @location과 @organization으로, 이 단어가 위치나 조직명으로 쓰일 수 있음을 나타낸다. 이와 같이 개체명 인식기가 하나의 형태소에 대해서 두 개 이상의 의미 표지를 부착할 경우, 자질 추출 과정은 의미 표지의 가능한 모든 조합으로 자질 세트(feature set)를 생성한다. 예를 들어, 전체 문장을 대상으로 의미 표지만을 이용하여 [표 2]에 대한 자질 세트를 구성하면 다음과 같이 두 개의 자질 패턴으로 이루어진다.

- [@location, none, @location, %tel_num, none, none]
- [@organization, none, @location, %tel_num, none, none]

자질 추출 방법은 형태에 따라 크게 두 가지로 구분된다. 한 단어가 하나의 자질을 구성하는 single-term 방법과 여러 단어로 이루어진 하나의 구가 자질이 되는 term-phrase 방법이 그것이다. Single-term 방법은 단일 단어로 구성되어 적은 문서에서도 많은 자질을 추출할 수 있다는 장점이 있는 반면에 문맥 정보를 포함할 수 없다는 단점이 있다. 이에 반해 term-phrase 방법은 공기 정보(co-occurrence information)를 이용해서 단어의 쌍을 하나의 자질로 보는 것으로 문맥 정보를 어느 정도 반영할 수 있다는 장점이 있다[20]. 본 논문에서는 슬라이딩 윈도 기법[20]을 이용하여 문맥 정보를 자질에 반영한다. 이렇게 자질 추출 과정에서 슬라이딩 윈도를 이용할 수 있는 이유는 일반적인 문서 자동 분류에 비하여 질의 분류가 가지는 다음과 같은 특징 때문이다.

- 일반 문서에 비하여 분류할 대상 데이타의 크기가 작다. 문서 자동 분류는 문서 전체를 대상으로 하지만 질의 분류는 사용자가 입력한 한 두 문장을 대상으로 한다.

• 분류할 대상 데이타의 크기가 작기 때문에 자질 수에 따른 학습 부담이 상대적으로 적다. 즉, 하나의 문장에서 추출된 자질의 수는 형태소 외적인 특성까지 반영하여도 문서 전체에서 추출한 자질 수보다 상대적으로 적다.

개체명 인식기를 거쳐서 추출된 자질 세트의 자질 패턴들은 슬라이딩 윈도 방법을 이용하여 조합되고 여러 개로 분할되어 새로운 패턴들이 된다. [그림 3]은 슬라이딩 윈도의 처리 방법을 나타낸 것이다. [그림 3]에서 Nn이 의미하는 것은 자질 세트에 포함된 특정 자질 패턴의 n번째 원소이다.

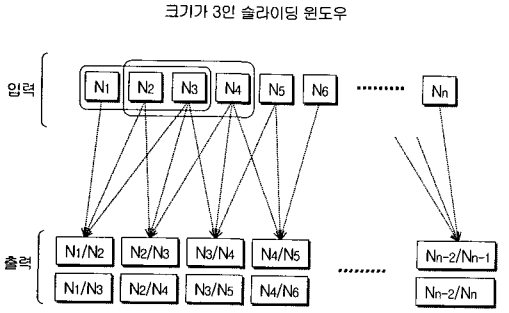

그림 3 슬라이딩 윈도 방법

슬라이딩 윈도는 형태소 단위로 처리되며, 형태소에 의미 표지가 부착되어 있다면 의미 표지를 이용하고 의미 표지가 없다면 형태소 분석 결과를 슬라이딩 윈도의 입력으로 이용한다. 이 때 형태소의 품사에 따라서 다음과 같이 다르게 처리한다.

• 체언, 용언, 외국어, 미등록어 : 형태소와 품사를 결합해서 이용
• 기호, 수식언을 제외한 기타 품사 : 품사 정보만을 이용
• 기호, 수식언 : 사용하지 않음

이렇게 조합, 분할된 한 세트의 자질 패턴들이 질의 유형 분류를 위한 학습 및 테스트 데이타로 이용된다.

3.3 자질 선택

Yang은 [21]에서 여러 가지의 자질 선택 방법을 사용하여 실험을 한 결과 카이 제곱 통계량과 정보 획득량을 사용하는 것이 범주 할당 문제에 가장 효과적임을 보였다. 본 논문에서는 이를 바탕으로 비교적 구현이 쉽고 고빈도 단어에 친화적인 카이 제곱 통계량을 사용하여 자질을 선택한다. 카이 제곱 통계량을 구하기 위한 수식은 (식1)과 같다.

$$\chi^2(t,c) = \frac{N\times(AD-CB)^2}{(A+C)\times(B+D)\times(A+B)\times(C+D)} \quad (1)$$

(식1)에서 A는 범주 c에 속해 있는 질의 중 용어 t를 포함하고 있는 질의의 수, B는 범주 c에 속하지 않은 질의 중 용어 t를 포함하고 있는 질의의 수, C는 범주 c에 속해 있는 질의 중 용어 t를 포함하지 않은 질의의 수, 그리고 D는 범주 c에 속하지 않은 질의 중 용어 t를 포함하지 않은 질의의 수이다.

(식1)의 계산이 끝나면 (식2)에 따라 각 범주별로 얻어진 카이 제곱 통계량 중에 가장 큰 값을 해당 용어의 자질 값으로 할당한다. 그리고, 이 값을 순위화하여 내림차순으로 정렬한다.

$$\chi^2_{\max}(t) = \max_{i=1}^{m} \chi^2(t,c_i) \quad (2)$$

카이 제곱 통계량에 의해 얻어진 순위를 바탕으로 정해진 순위 안에 들지 못한 자질들은 질의 유형 분류를 위한 의미 있는 정보를 제공해 주지 못하는 것으로 판단하여 자질에서 제외시킨다.

3.4 질의 표현

본 논문에서 제안하는 질의 유형 분류기는 질의 표현을 위해서 가장 일반적으로 사용되는 문서 표현 방법인 벡터 공간 모델을 이용한다. 이것은 문서 전체에 나타난 각 자질의 빈도를 이용하여 문서를 하나의 벡터로 표현하는 것으로, 보통 자질의 빈도(TF)와 역문헌빈도(IDF) 혹은 역범주빈도(ICF)를 이용하여 가중치를 줌으로써 문서를 표현한다.

본 논문에서는 질의에서 추출된 자질들을 문서로 취급하여, 일반적인 문서 표현을 위한 이진 가중치, 단어 빈도 가중치, TF·IDF 가중치, tfc-가중치를 질의 표현을 위해 사용한다.

이진 가중치 방법은 i번째 질의에서 자질 k의 가중치를 (식3)으로 구한다. 여기서 f_{ik}는 i번째 질의에서 추출한 자질 k의 출현 빈도이다.

$$a_{ik} = \begin{cases} 1 & \text{if } f_{ik}>0 \\ 0 & otherwise \end{cases} \quad (3)$$

단어 빈도 가중치 방법은 i번째 질의에서 자질 k의 가중치를 (식4)로 구한다.

$$a_{ik} = f_{ik} \quad (4)$$

TF·IDF 가중치 방법은 i번째 질의에서 자질 k의 가중치를 (식5)로 구한다.

$$a_{ik} = f_{ik}\times\log\left(\frac{N}{n_k}\right) \quad (5)$$

(식5)에서 N은 전체 질의의 수이며, n_k는 자질 k가

출현한 질의의 수이다.

tfc-가중치(tfc-weighting) 방법은 i번째 질의에서 자질 k의 가중치를 (식6)으로 구한다. 여기서 K는 질의를 표현하는 데 사용되는 전체 자질의 수이다.

$$a_{ik}=\frac{f_{ik}\times \log(\frac{N}{n_k})}{\sqrt{\sum_{j=1}^{K}[f_{ik}\times \log(\frac{N}{n_j})]^2}} \quad (6)$$

3.5 질의 범주화

본 논문에서 제안하는 질의 유형 분류기는 지지 벡터 기계를 질의 범주화를 위해 사용한다. 질의 학습 과정에서는 학습 질의를 바탕으로 지지 벡터 기계를 학습한다. 학습의 결과로 질의 유형 분류 체계에 대한 지지 벡터들이 생성되며, 생성된 지지 벡터들을 이용해서 질의 분류 과정에서는 입력된 질의의 범주를 구분한다. 지지 벡터 기계를 구현한 프로그램은 Joachims에 의해 구현된 SVM^{light}[22]를 사용한다.

지지 벡터 기계는 두 개의 범주를 구분하기 위해 사용되는 알고리즘이다. 따라서 질의 유형 분류기는 범주마다 지지 벡터 기계를 학습하고, 입력 질의를 학습된 각 범주의 결정면들과 비교하여 하나의 범주로 할당한다[23].

3.6 질의 유형 결정

자질 추출 과정에서 질의에 대한 자질 세트가 두 개 이상 생성된 경우, 각 자질 세트에 대해서 범주화 결과가 생성된다. 질의 유형 분류기는 [알고리즘 1]과 같은 방법으로 생성된 범주화 결과 중 가장 적합한 것을 해당 질의의 질의 유형으로 결정한다.

1. 범주화 결과를 의미 범주별로 그룹화(grouping)한다.
2. 각 그룹에서 가장 높은 범주화 확률을 가지는 값을 해당 그룹의 범주화 확률로 선택한다.
3. 범주화 확률이 가장 높은 그룹을 선택한다.
4. 선택한 그룹이 의미 범주의 첫 번째나 두 번째 계층만으로 구성되어 있다면 범주화 확률이 가장 높은 값을 질의 유형으로 결정한다.
5. 선택한 의미 범주가 두 계층 모두 값을 가지고 있다면 두 번째 계층에 속한 결과 중 범주화 확률이 가장 높은 값을 질의 유형으로 결정한다.

알고리즘 1 질의 유형 결정 과정

예를 들어 "월드컵의 개최 요일은?" 질의에 대해서 3개의 자질 세트가 생성되어 *date, day, weekday*로 범주화 되었다면, 두 번째 계층에 속한 *day*나 *weekday* 중 범주화 확률이 높은 쪽을 질의의 질의 유형으로 결정하게 된다. [그림 4]는 범주화 결과를 통해서 질의 유형을 결정하는 예를 그림으로 나타낸 것이다. [그림 4]와 같이 사용자의 질의가 4개의 자질 세트로 구성된다면 질의 범주화 과정에서는 각각의 자질 세트에 대해서 질의 유형과 범주화 확률을 계산한다. 범주화 결과의 범주 코드를 바탕으로 결과 중 3개는 *identification* 의미 범주로 그룹화(grouping)되고, 1개는 *animal*로 묶여서 [알고리즘 1]에서 설명한 방법으로 질의 유형을 최종 결정한다.

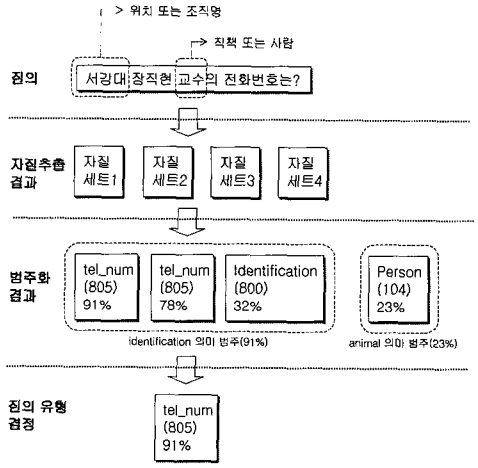

그림 7 질의 유형 분류 예

4. 실험 및 평가

4.1 실험 데이타

제안된 질의 유형 분류기의 성능을 실험하기 위해서 사용된 데이타는 서강대학교(www.sogang.ac.kr)와 코리아인터넷닷컴(korea.internet.com)과 같은 실제 웹사이트에서 수집한 사용자 질의 로그이다. 수집된 질의는 수작업으로 미리 정의된 질의 유형에 따라서 분류되었다. 수집된 데이타는 총 78개의 질의 유형으로 구분된 7,726개의 질의로 구성된다. 각 질의는 하나의 질의 유형만을 가지며 중복 할당을 허용하지 않았다. 질의 유형 분류기에서 사용하는 의미 범주는 105개이지만, 27개의 의미 범주에 대해서는 실제 수집된 질의에서 나타나지 않았다. 그리고 33개의 의미 범주에 대해서는 수집된 질의에서의 출현 빈도가 10회 미만이었다.

실험 데이타를 학습 데이타와 테스트 데이타로 분리하기 위해서 질의 유형별 분포에 맞춰서 임의로 10%의 데이타를 추출해서 테스트 데이타로 사용하고, 나머지 90%는 학습 데이타로 사용하였다.

4.2 실험

4.2.1 자질 추출 과정에 대한 실험

자질 추출 과정에서 어떤 자질이 질의 유형 분류기의 성능에 영향을 미치는지 확인하기 위해서 자질 추출 과정의 각 자연어처리 과정들을 하나씩 적용하면서 실험을 수행하였다. 이 실험에서 사용한 자질 추출 방법은 다음과 같다.

- 방법1 : 형태소 분석 결과만을 이용
- 방법2 : 형태소 분석 결과와 개체명 인식 결과를 이용
- 방법3 : 형태소 분석 결과, 개체명 인식 결과, 슬라이딩 윈도 결과를 이용 (슬라이딩 윈도 크기 6)

이 실험에서 지지 벡터 기계는 선형결정면으로 학습하였고, 문서 표현 방법으로는 tfc-가중치를 선택하였다. 자질 선택과정은 수행하지 않았으며, 두 개 이상의 질의에 출현한 모든 자질들을 사용하였다. [표 3]에서 보듯이'방법3'이 가장 좋은 성능을 보였으며, 형태소 분석 결과만을 사용한 방법보다 전체적으로 2.4% 정도 성능 향상이 있었다. 방법1에 슬라이딩 윈도를 사용한 경우는 방법2와 비슷한 성능을 보였다. 그리고, '방법3'은 분포도 상위 9개의 질의 유형 중에 6개의 질의 유형에 대해서 좋은 성능을 보였다.

표 3 자질 추출 과정에 대한 성능 비교

질의 유형	방법1	방법2	방법3
person	84.3	85.3	**90.8**
description	78.8	76.6	**80.0**
method	89.3	**89.8**	89.5
url	95.4	**96.4**	92.0
tel_num	**100.0**	**100.0**	**100.0**
location	76.5	78.0	**79.5**
department	76.9	77.9	**78.9**
document	83.0	84.6	**92.0**
date	**87.5**	85.1	83.3
total	83.8	84.3	**86.2**

다음으로 자질 추출 과정에서 슬라이딩 윈도의 크기가 성능에 미치는 영향을 알아보기 위해서 '방법3'에서 슬라이딩 윈도의 크기를 2에서 10까지 늘려가면서 실험하였다. [그림 5]는 슬라이딩 윈도의 크기에 따른 분류 정확성을 비교한 그래프로, 슬라이딩 윈도 크기 6 이후에는 더 이상의 성능 향상이 없거나 성능이 떨어지는 경우가 있음을 확인할 수 있다.

4.2.2 자질 선택 과정에 대한 실험

질의 학습 과정에서 얻어진 총 13,931개의 자질들을 자질 선택 과정에서 제한하여 자질 수에 대한 질의 유형 분류기의 성능을 비교하였다.

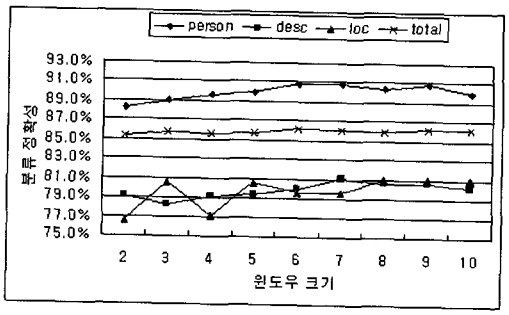

그림 5 슬라이딩 윈도의 크기에 따른 비교 그래프

이 실험에서는 지지 벡터 기계는 선형결정면으로 학습하였고, 문서 표현 방법으로는 tfc-가중치를 사용하였다. [그림 6]은 자질 선택 과정에서 자질 수를 1,000개 단위로 늘리면서 실험한 그래프다. 실험을 통해서 자질 수가 5,000개가 될 때까지는 자질 수의 증가에 따라 성능 향상을 보이다가 그 이상의 자질 수에서는 성능의 향상이 크게 없음을 알 수 있었다. 특히 *location*의 경우에는 자질 수가 증가하면서 성능이 떨어짐을 볼 수 있다. *description*은 자질 수가 적을 때 급격한 성능의 저하를 보였으며, 다른 질의 유형들의 경우에는 자질의 수에 관계없이 75% 이상의 분류 정확성을 보여 주었다. 결론적으로 자질 선택에 대한 실험 결과는 자질 수가 5,000개 일 때 학습 속도와 성능 면에서 가장 좋았다. 학습 속도는 2배 이상 단축되었으며, 정확률은 85.9%였다.

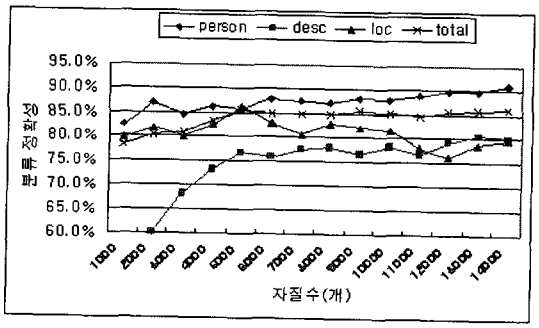

그림 6 자질 수에 따른 성능 비교 그래프

4.2.3 질의 표현 과정에 대한 실험

어떤 문서 표현 방법이 질의 유형 분류기에서 좋은 성능을 보이는지 알아보기 위해서 이진 가중치, 단어 빈도 가중치, TF · IDF 가중치, tfc-가중치로 질의 표현 방법

을 바꿔서 실험을 수행하였다. 이 실험에서 지지 벡터 기계는 선형 결정면으로 학습하였고, 자질 선택 과정은 사용하지 않았다. [표 4]에서 보듯이 tfc-가중치 방법이 가장 좋은 결과를 보였으며, 9개의 분포도 상위 유형 중 6개 유형에서 가장 좋은 결과를 나타내었다. TF·IDF 가중치 방법은 두 번째로 좋은 결과를 보였으며, 이진 가중치와 단어 빈도 가중치는 낮은 성능을 보였다. tfc-가중치를 사용한 경우에 이진 가중치나 단어 빈도 가중치 방법에 비해 1.2% 정도의 성능 향상이 있었다.

표 4 문서 표현에 따른 실험 결과

질의 유형	이진	단어 빈도	TF-IDF	tfc
Person	88.2	88.6	89.7	**90.8**
description	79.1	78.0	78.0	**80.0**
Method	89.3	89.3	**90.5**	89.5
url	91.7	92.6	**92.7**	92.0
tel_num	**100.0**	**100.0**	**100.0**	**100.0**
location	80.0	**81.5**	80.5	79.5
department	76.3	74.7	76.3	**78.9**
document	87.5	87.5	89.8	**92.0**
Date	80.9	80.9	80.9	**83.3**
Total	85.0	84.9	85.6	**86.2**

4.2.4 질의 범주화 과정에 대한 실험

질의 범주화 과정에서 지지 벡터 기계의 학습 방법이 성능에 어떠한 영향을 미치는지 확인하기 위해서 결정면에 대한 학습 방법을 바꾸면서 시스템의 성능을 살펴보는 실험을 수행하였다. 결정면 학습 방법으로는 선형(linear), 비선형(polynomial), rbf(radical basis function)를 사용하였다. 비선형은 2차, 3차 함수에 대해서 실험하였으며, rbf는 실험 상수를 0.8, 1.0로 조절하며 실험을 수행하였다. 나머지 지지 벡터 기계와 관련된 실험 상수는 전부 SVM^{light}의 기본값을 사용하였다. 이 실험에서 자질 선택 과정은 사용하지 않았고, 문서 표현 방법으로는 tfc-가중치 방법을 사용하였다. [표 5]에서 2차 함수를 이용한 비선형 학습 방법이 질의 유형 분류기에서 가장 좋은 성능을 보임을 알 수 있다. 전체적으로는 선형 학습 방법과 2, 3차 함수를 이용하는 비선형 결정면 학습 방법이 좋은 결과를 보여 주었다.

4.2.5 성능 평가

앞의 실험에서 자질 추출 과정에서 슬라이딩 윈도의 크기를 6으로 설정하고, 질의 표현 방법으로는 tfc-가중치 방법을 사용하며, 지지 벡터 기계를 2차 함수를 이용한 비선형 결정면으로 학습했을 때 질의 유형 분류기의 성능은 86.4%로 가장 좋은 성능을 보였다.

표 5 지지 벡터 기계의 학습 방법에 따른 실험 결과

질의 유형	선형	비선형		rbf	
		2	3	0.8	1.0
person	**90.8**	90.8	90.0	88.8	88.4
description	**80.0**	79.4	79.4	79.8	78.9
method	**89.5**	89.0	88.5	88.0	88.0
url	**92.0**	**92.0**	90.4	89.5	89.5
tel_num	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
location	79.5	**80.0**	78.2	79.1	78.2
department	78.9	77.9	77.3	**79.5**	79.5
document	92.0	**94.1**	94.1	**94.1**	94.1
date	**83.3**	83.3	80.9	80.9	78.3
total	86.2	**86.4**	85.4	85.3	84.9

본 논문에서 제안한 질의 유형 분류기의 성능을 평가하기 위해서 기본 시스템과 규칙 기반 시스템, C4.5를 이용한 통계 기반 시스템을 이용하였다. 기본 시스템은 다음과 같이 질의에 나타난 의문사와 초점 단어의 의미 표지에 따라서 휴리스틱(heuristic)으로 질의 유형을 결정한다.

- 질의에 '누구', '언제', '어디서' 같은 의문사가 포함되어 있다면 의문사에 의거해서 질의 유형을 결정한다. 예를 들어서 "삼성전자의 사장은 누구죠?"라는 질의에 대해서 *person*을 질의 유형으로 결정한다.
- 질의에 의문사가 포함되어 있지 않다면, 질의의 마지막 초점 단어의 의미 표지에 의거해서 질의 유형을 결정한다. 예를 들어서 "삼성전자의 사장은?"이라는 질의에 대해서, '사장'의 의미 지표가 *@person*이므로 *person*으로 질의 유형을 결정한다. 이 때 초점 단어가 여러 개의 의미 표지를 가진다면 제일 처음의 것을 선택한다.

규칙 기반 시스템으로는 한국어 질의응답 시스템인 [24]에서 채택한 방법을 사용하였고, C4.5를 이용한 통계 기반 시스템은 [25]에서 채택한 방법을 사용하였다. [표 6]은 각 시스템의 성능을 나타낸 것이다.

표 6 질의 유형 분류기의 성능 비교

시스템 종류	분류 정확도(%)
기본 시스템	61.8
규칙기반 시스템	84.0
통계기반 시스템(C4.5)	75.0
제안한 시스템	86.4

본 논문에서 제안한 질의 유형 분류기는 기본 시스템에 비해서 24.6% 더 높은 성능을 보였다. 또한 규칙 기반 시스템보다 2.4% 정도 더 좋은 성능을 보여줘서 복잡한 규칙을 기술하지 않고도 충분히 좋은 성능을 낼 수 있음을 보여 주었다. 그리고 기존의 통계 기반 시스템에 비해서도 11.4% 더 좋은 성능을 보여줘서 제안한 방법이 질의 유형 분류 문제에 더 적합함을 알 수 있었다.

5. 결 론

본 논문에는 질의 유형 분류 문제에 문서 범주화 기법을 도입한 자동 질의 유형 분류 기를 제안하였다. 질의 유형 분류기는 질의에서 자질들을 추출하기 위해서 형태소 분석기, 개체명 인식기, 슬라이딩 윈도 기법을 적용하였고, 문서 범주화 기법에서 좋은 성능을 내는 지지 벡터 기계 모델을 이용하여 질의 유형을 분류한다.

실험은 실제 웹 사이트에서 수집한 7,726개의 질의를 이용하여 수행되었다. 실험을 통해서 질의 유형 분류의 각 과정에서 좋은 성능을 보이는 방법들을 알아보았으며, 최대 86.4%의 높은 질의 분류 정확도를 보였다.

본 논문은 짧은 질의에서 가능한 많은 정보를 추출하기 위해서 자질 추출 과정에서 다양한 자연어처리 기법을 이용하였다. 하지만 실험에서 살펴 본 것과 같이 몇몇 질의 유형의 경우에는 오히려 이러한 추가적인 정보가 시스템의 성능을 떨어지게 하는 요인이 되었다. 향후에 본 논문에서 제안한 시스템의 성능을 향상시키기 위해서는 추출된 자질의 가중치를 차등 적용하는 방법 등을 통해 자질 추출 과정을 개선해야 할 것이다. 그리고 의미 범주에 따라 자질 선택을 별도로 하여 자질 자체가 범주에 따라 불공평하게 뽑혀지는 것을 개선할 것이다. 또한 지지 벡터 기계 외의 다른 문서 범주화 기법들을 본 논문과 같은 방식으로 적용해서 어떤 범주화 기법이 질의 유형 분류 문제에 적합한지 연구할 수 있을 것이다.

참 고 문 헌

[1] Voorhees E. and Tice D. M., "Building a Question Answering Test Collection", In *Proceedings of SIGIR 2000*, pp. 200-207, 2000.

[2] AAAI Fall Symposium on Question Answering, *http://www.aaai.org/Press/Reports/Symposia/Fall/fs-99-02.html*

[3] TREC (Text REtrieval Conference) Overview, *http://trec.nist.gov/overview.html*

[4] Moldovan D., Harabagiu S., Paşca M., Mihalcea R., Goodrum R., Gi^rju R. and Rus V., "LASSO: A Tool for Surfing the Answer Net", In *Proceedings of The Eighth Text REtrieval Conference (TREC-8)*, from http://trec.nist.gov/pubs/trec8/t8_proceedings.html, 1999.

[5] Prager J., Radev D., Brown E. and Coden A., "The Use of Predictive Annotation for Question Answering in TREC8", In *Proceedings of The Eighth Text REtrieval Conference (TREC-8)*, from http://trec.nist.gov/pubs/trec8/t8_proceedings.html, 1999.

[6] O. Ferret, B. Grau, G. Illouz, and C. Jacquemin, "QALC the Question-Answering program of the Language and Cognition group at LIMSI-CNRS", In *Proceedings of the Eighth Text REtrieval Conference (TREC-8), http://trec.nist.gov/pubs/trec8/t8_proceedings.html*, Gaithersburg, Maryland, 1999.

[7] Harabagiu S., Moldovan D., Pasca M., Mihalcea R., Surdeanu M., Bunescu R., Gîrju R., Rus V. and Morarescu P., "FALCON: Boosting Knowledge for Answer Engines", In *Proceedings of the Ninth Text REtrieval Conference*, from http://trec.nist.gov/pubs/trec9/t9_proceedings.html, 2000.

[8] Kupiec J., "Murax: A Robust Linguistic Approach for Question Answering Using an On-line Encyclopedia", *In Proceedings of SIGIR'93*, 1993.

[9] Berri J., Molla D., and Hess M., "Extraction automatique de réponses: implémentations du systéme ExtrAns", In *Proceedings of the fifth conference TALN 1998*, pp. 10-12, 1998.

[10] Vicedo J. L. and Ferrández A., "Importance of Pronominal Anaphora resolution in Question Answering systems", In *Proceeding of ACL 2000*, pp. 555-562, 2000.

[11] Prager J., Brown E. and Coden A., "Question-Answering by Predictive Annotation", *In Proceedings of SIGIR 2000*, pp. 184-191, 2000.

[12] Hermjakob U., "Parsing and Question Classification for Question Answering", *In Proceedings of the ACL Workshop Open-Domain Question Answering*, pp. 17-22, 2001.

[13] Ittycheriah A., Franz M., Zhu W. and Ratnaparkhi A., "IBM's Statistical Question Answering System", In *Proceedings of the Ninth Text REtrieval Conference, http://trec.nist.gov/pubs/trec9/t9_proceedings.html*, Maryland, 2000.

[14] Ittycheriah A., Franz M., Zhu W. and Ratnaparkhi

A., "Question Answering Using Maximum Entropy Components", In *Proceedings of NAACL*, 2001.
[15] Mann G. S., "A Statistical Method for Short Answer Extraction", In *Proceedings of the ACL Workshop Open-Domain Question Answering*, pp. 13-30, 2001.
[16] Miller G., *WordNet: An on-line lexical database*, International Journal of Lexicography, Vol. 3(4), 1990.
[17] U. Hermjakob and R. J. Mooney, "Learning Parse and Translation Decisions From Examples With Rich Context", In *Proceedings of the 35th ACL*, pp. 482-489, 1997.
[18] Vapnik V., *The Natural of Statistical Learning Theory*, Springer, New York, 1995.
[19] diquest, http://www.diquest.com
[20] Maarek Y., Berry D. and Kaiser G., *An Information Retrieval Approach For Automatically Construction Software Libraries*, IEEE Transaction On Software Engineering, Vol. 17, No. 8, pp.800-813, August 1991.
[21] Y. Yang and J. O. Pederson, "A comparative study on feature selection in text categorization", In *Proceedings of the 14th International Conference on Machine Learning*, 1997.
[22] SVM^{light}, http://ais.gmd.de/~thorsten/svm_light
[23] J. Weston and C. Watkins, "Support vector machines for multiclass pattern recognition", In *Proceedings of the 7th European Symposium On Artificial Neural Networks, April* 1999.
[24] Kim H., Kim K., Lee G. G. and Seo J., "MAYA: A Fast Question-answering System Based On A Predictive Answer Indexer", In *Proceedings of the ACL Workshop Open-Domain Question Answering*, pp. 9-16, 2001.
[25] 김학수, 안영훈, 서정연, *하이브리드 방법에 기반한 사용자 질의 의도 분류*, 한국정보과학회 제30권 1,2호., pp. 51-57, 2003.

김 학 수

1996년 건국대학교 전자계산학과 학사. 1998년 서강대학교 컴퓨터학과 석사. 2003년 서강대학교 컴퓨터학과 박사. 2001년~현재 다이퀘스트 연구소 책임연구원. 관심 분야는 자연어 처리, 대화 시스템, 생략 및 대용어 처리, 화행 분석, 정보 검색, 질의 응답 시스템

안 영 훈

2000년 서강대학교 컴퓨터학과 학사. 2002년 서강대학교 컴퓨터학과 석사. 2000년~현재 다이퀘스트 연구소 선임연구원. 관심 분야는 자연어 처리, 질의 의도 분석, 정보 검색

서 정 연

1981년 서강대학교 수학과 학사. 1985년 미국 Univ. of Texas, Austin 전산학과 석사. 1990년 미국 Univ. of Texas, Austin 전산학과 박사. 1990년~1991년 미국 Texas Austin, UniSQL Inc. Senior Researcher. 1991년 한국과학기술원 인공지능 연구센터 선임연구원. 1991년~1995년 한국과학기술원 전산학과 조교수. 1995년~1996년 서강대학교 전산학과 조교수. 1996년~현재 서강대학교 컴퓨터학과 부교수. 관심분야는 한국어정보처리, 자연언어처리, 기계번역, 대화처리