의견의 주체를 찾기 위한 후보어휘의 의견주체점수 부여 방법과 Self-training

정헌영 ○, 김준기, 이예하, 이종혁

포항공과대학교 전자컴퓨터공학부 컴퓨터공학과

{blesshy○, yangpa, sion, jhlee}@postech.ac.kr

Opinion Holders Identification by Candidate Lexical Score and Self-training

Hun-young Jung○ , Jungi Kim, Yeha Lee, Jong-Hyeok Lee

Department of Computer Science and Engineering
Division of Electrical and Computer Engineering
Pohang University of Science and Technology

요 약

의견의 주체를 찾는 일은 의견 분석을 하는데 있어 중요하게 여겨지는 부분이다. 본 논문은 시스템의 성능을 높이기 위해 각 명사구에 의견주체후보로서의 점수를 부여하는 방법을 제안하였고 점수가 학습 데이터에 제약되는 문제를 해결하기 위해 Self-training을 도입하였다. 본 논문의 방법 중 의견주체의 후보점수를 부여하는 방법은 Baseline과 비교하여 정확률을 11.8% 더 높였고 Self-training 방법으로는 정확률을 0.97% 높였다. 두 방법을 동시에 사용하였을 경우 정확률이 13.2% 증가하였다.

1. 서 론

의견분석은 문서에 대한 분석기술이 발달하면서 등장한 분야로 최근 많은 관심을 끌고 있다. 의견분석이란 문서, 혹은 문장에 의견이 있는지 판별하는 것뿐만 아니라 의견의 극성, 대상, 주체 등을 분석하는 것까지 포함된다. 본 논문에서는 이러한 분석 대상 중에서 의견의 주체를 판별하는 것을 목표로 하였다.

의견의 주체는 문서나 문장에서 의견을 표현한 주체이다. 예를 들어 “나는 그가 그녀를 좋아하는 것을 이해할 수 없다.”라는 문장에서 ‘좋아하는’의 주체는 ‘그’이고 ‘이해할 수 없다’의 주체는 ‘나’가 된다. 의견의 주체는 직접적인 언급이 아닌 경우에도 해당이 되는데 예를 들어 “그 보고서에 따르면 제품에 대한 사용자의 평가가 좋다.”라는 문장에 있어서 ‘좋다’라는 의견을 가진 주체는 ‘사용자’이지만 ‘보고서’도 해당 의견을 간접적으로 나타낸 것이 된다.

이렇게 의견의 주체를 알아낸 결과는 여러 분야에 응용된다. 예를 들면 의견의 주체와 대상이 같은 문장을 연결하는 방식으로 의견문서를 요약하는 시스템[1], “누가 어떤 것에 대해 어떻게 생각하는가?”라는 종류의 질문에 대한 질의응답시스템[2] 등이 있다.

의견의 주체를 판별하기 위한 한가지 방법은 문장의 표층적 형태에 기반한 패턴을 정하고 이를 이용하는 규칙을 사용하는 것이다. 의견의 주체는 대부분 ‘~에 따르면’, ‘~가 말하였다’, ‘~을 좋아한다’, ‘~라고 생각한다’ 등 몇몇 일정한 패턴으로 나타난다. 이러한 특성을 이용하여 미리 규정한 패턴에 따른 몇 가지 규칙을 정하여 의견 문장에서 주체를 찾아내는 연구가 있었다[3].

또 하나의 방법은 의미역 결정(Semantic Role Labeling)을 이용한 것이다. 의미역 결정 방법은 문장 내의 여러 개체에 대해서 문장에서 가지는 의미상의 역할을 판단하는 것이다. 이 방법으로 문장 내의 여러 구에 의미상의 역할을 판별하여 표시하게 되는데 이러한 시스템의 결과를 사용하여 의견의 주체를 알아내거나[4] 표시할 태그의 종류를 바꾸어 사용하는 연구가 있었다[2]. [2]에 따르면 Propbank와 FrameNet의 데이터를 살펴본 결과 의견을 가진 문장에서 ‘AGENT’로 표시된 구들이 거의 의견의 주체와 일치하였다고 보고했다.

의견의 주체를 찾는 문제를 각 단어에 대한 순차적인 태깅의 관점으로 바라보기도 한다. 각 단어가 의견의 주체가 표시된 구절의 시작단어인지 중간에 있는 단어인지, 주체를 나타낸 구절에 포함되지 않는 단어인가를 표시하여 시작하는 단어로부터 시작하여 연속적인 단어를 묶음으로서 의견의 주체를 알아내는 것이다. 대표적으로는 [5]가 있는데 이것은 Conditional Random Field를 사용하여 Multi-Perspective Question Answering (MPQA)

말뭉치[1]에 사용한 연구이다.

그리고 기계학습의 방법을 이용하여 문장에 존재하는 각 단어나 명사구에 점수를 부여하고 이를 통해 적합한 명사구를 선택하는 방법으로 접근한 연구가 있었다. [6]에서는 이러한 방법으로 접근하는데 있어서 단어적인 정보를 사용하지 않고 문법적 정보만을 사용하였다. [7]은 문법적, 단어적 측면의 자질 이외에 미리 정한 표층 형태에 기반한 패턴에 소속되는 가를 자질로 사용하여 의견의 주체를 찾는 연구 이였다.

본 논문에서는 의견의 주체를 판별하는 성능을 높이기 위하여 각 단어가 주체로서 사용될 가능성을 계산하고 이를 통해 명사구에 의견의 주체로서 사용될 가능성에 대한 점수를 부여하는 방법을 제안하였다. 그리고 이러한 점수가 학습 데이터에 제약되는 문제를 해결하기 위해 Self-training을 적용하여 보았다.

사용한 말뭉치는 온라인 상의 영어 신문 기사를 모은 것으로 NTCIR-7 workshop에서 학습데이터로 제공한 것과 MPQA데이터이다.

2. 접근 방법

주어진 의견 문장이 신문 기사일 경우, 의견의 주체는 문장에 포함 되는 경우가 많다. 이에 따라 의견의 주체를 찾을 때 의견 문장의 명사구중에서 의견의 주체로서 가장 적합한 명사구를 선택하게 된다. 본 논문에서는 기계학습방법으로 각 명사구가 의견의 주체로서 사용될 확률을 구하고 높은 확률을 지닌 명사구를 의견의 주체로서 선택하는 방법으로 연구를 진행하였다.

한 문장에 하나의 의견이 있다는 가정 하에 문장 중에서 가장 높은 확률을 지닌 명사구를 선택하도록 하였다. 의견이 문서의 저자로부터 비롯되는 경우에는 문장의 어떤 명사구도 의견의 주체가 아니다. 이 경우를 고려하여 임계값을 넘는 확률을 가진 명사구가 문장에 없을 경우 '저자'를 의견의 주체로 선택하였다. 인용문이나 대용어구 같이 의견의 주체가 이전문장에 있는 경우는 고려하지 않았다.

각 명사구가 의견의 주체로서 사용되는 확률을 계산하기 위해 Maximum Entropy (ME) 모델을 사용하였다. ME 모델은 데이터로부터 얻은 강제조건들 이외의 확률들에 대해서는 최대한 동일한 확률을 부여하는 모델[8]로서 자연언어 처리[8]나 의견 분석[9]에 사용된다. 본 논문에서는 모델을 구현하는데 기존 연구에서 사용되었던 자질을 사용하였다[2,4,5,7,10]. <표 1>의 F_1~F_{18}은 이러한 자질들이다. F_1~F_5는 명사구에 대한 정보에 해당하고 F_6~F_{18}은 의존문법트리에서 추출할 수 있는 주변정보이다.

개별 단어가 의견의 주체로서 사용될 수 있는 가능성을 평가하고 이것을 ME 모델에 반영하기 위해서 단어에 의견주체점수를 측정하고 이것으로 명사구의 점수를 구하여 자질로서 사용하는 것을 제안하였다(<표 1>의 F_{19}). 그리고 이 점수가 학습 데이터에 의해 제약되지 않도록 Self-training을 적용하였다.

표 1 의견의 주체를 판별하기 위해 사용된 자질

자질	의미
F_1	명사구의 표층 형태
F_2	명사구가 대문자를 포함하고 있는지 여부
F_3	문장에서 명사구의 위치
F_4	명사구의 지배소
F_5	F_4의 형태소 태그 정보
F_6	F_4의 지배소
F_7	F_6의 형태소 태그 정보
F_8	F_4, F_6사이의 의존관계
F_9	의존문법트리에서 명사구에 가장 가까운 동사
F_{10}	F_9의 형태소 태그 정보
F_{11}	의존문법트리에서 명사구에서 F_9까지 도달하는 과정
F_{12}	F_{11}에서 첫 번째 의존관계
F_{13}	F_{11}에서 마지막 의존관계
F_{14}	문장의 주동사
F_{15}	F_{14}의 형태소 태그 정보
F_{16}	의존문법트리에서 명사구에서 F_{14}까지 도달하는 과정
F_{17}	F_{16}에서 첫 번째 의존관계
F_{18}	F_{16}에서 마지막 의존관계
F_{19}	단어의 의견주체점수

2.1 후보어휘의 의견주체점수 부여

의견의 주체로서 사용되는 명사구를 살펴보면 사람, 단체 등의 고유명사 이외에도 직업, 집단, 서류를 의미하는 단어들이 사용되는 것을 알 수 있다. 이러한 단어들은 의미상으로 의견을 내세울 수 있는 주체로서의 가능성을 가지고 있다. 하지만 의견의 주체로서 쓰인 명사구의 모든 단어가 그런 가능성을 가진 것은 아니다. 예를 들면 '인터넷 사용 분석 연구원'의 경우 주체로서 사용되는 단어는 오직 '연구원'뿐이다. 이러한 단어는 '농업 연구원', '천문 연구원', '전자부품연구원'등 의견의 주체가 되는 여러 명사구에서 공통적으로 등장할 가능성이 높다. 이러한 추측을 바탕으로 각 단어가 의견의 주체로서 사용될 수 있는 가능성을 측정하고 단어의 점수를 통해 명사구의 점수를 계산하여 ME 모델의 자질로서

[1] http://www.cs.pitt.edu/mpqa/databaserelease/

사용하였다. 의견의 주체인 명사구에 등장하는 개별 명사에 대해서 다음과 같은 점수를 부여하여 사용하였다.

$$\text{Score}(\text{word}) = \frac{\text{Count}(\text{word} \in S_{\text{holder}})}{\text{Count}(\text{word})}$$

여기서 S_{holder} 는 의견의 주체로서 사용된 명사구의 집합이다. 그리고 $\text{Count}(\text{word})$ 는 전체 말뭉치에서 단어가 등장한 횟수이다.

단어의 점수를 바탕으로 각 명사구에는 다음과 같은 점수를 부여하였다.

$$\text{Score}(\text{NP}) = \frac{\sum_{\text{word} \in \text{NP}} \text{Score}(\text{word})}{\text{Length}(\text{NP})}$$

여기서 $\text{Length}(\text{NP})$ 는 명사구의 길이로, 명사구에 포함되는 단어의 수 이다.

2.2 Self-training

Self-training은 부트스트래핑 알고리즘 중 하나로 부분지도학습으로 사용되는 방법이다. 지도학습에 사용할 수 있는 데이터의 양이 적은 경우에 사용된다. 먼저 학습데이터를 학습하여 만든 기본 모델을 사용하여 평가한 데이터를 평가한다. 그 결과를 옳은 답으로 간주하고 기존의 학습 데이터에 합하여 사용함으로 기존의 학습 데이터 이외의 다른 데이터를 학습 데이터로 사용하는 것과 같은 효과를 보게 된다[11]. <그림 1>은 Self-training사용시에 각 데이터가 사용되는 순서와 흐름을 나타낸다.

의견의 주체를 판별하는 작업에 있어서 평가 데이터가 문서단위로 주어질 경우, 해당 문서 내에서는 여러 의견문장에 대해서 하나의 명사구가 동일하게 의견의 주체인 경우가 많다. 또한 같은 형태의 명사구라도 문서에 따라 의견의 주체로 사용되는 정도가 다르다. 이 때문에 단어의 의견주체점수를 계산하는데 있어서 학습 데이터만을 사용할 경우 점수가 학습 데이터에 제약되어서 다른 데이터에 사용할 경우 해당 평가 데이터의 정보와 일치하지 않는다. 이런 문제를 해결하기 위해 Self-training을 사용하였다.

Self-training 사용하기 위해 앞서 말하였던 방식으로 만든 ME 모델로 평가 데이터를 평가하여 각 명사구마다 의견의 주체인지를 표시하였다. 이를 기존의 학습 말뭉치와 합하여 ME 모델의 학습 말뭉치로 사용하였다.

3. 실험 방법

3.1 Baseline

성능 향상의 기준을 측정하기 위해 학습 말뭉치에 있는 명사구마다 <표 1>에서 F_1~F_{18}에 해당하는 자질에 대한 정보를 추출하여 ME 모델에 학습하였다. 문장에서 명사구를 추출하는 작업과 각 명사구에 대한 해당 자질을 추출하기 위해서 Stanford parser[2]를 사용하였다. Maximum Entropy 모델의 구현은 기존에 개발되어있는 "Maximum Entropy Modeling Toolkit for Python and C++"[3]을 사용하였다.

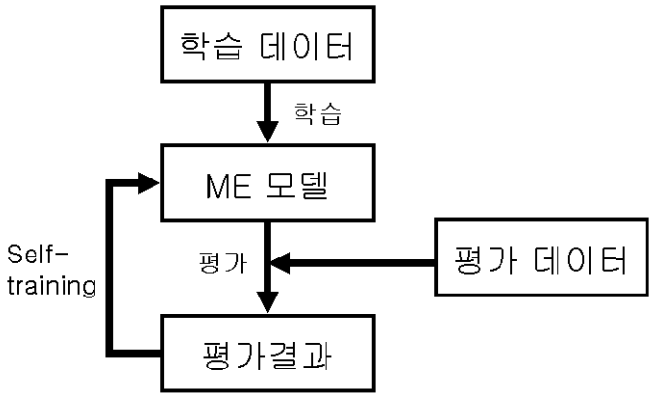

그림 1 Self-training 도식

3.2 실험 데이터

3.2.1 학습 데이터(training corpus)

본 논문에서는 ME 모델을 학습하기 위해 학습 데이터로 MPQA 말뭉치를 사용하였다. 이 말뭉치는 535개의 문서를 각 문장마다 수동으로 평가하여 의견을 나타낸 부분마다 의견의 주체가 표시되어 있다[12]. 본 논문에서는 이 말뭉치 전체를 학습 데이터로 사용하지 않았다. MPQA 말뭉치 중 의견의 주체를 문장 안에 포함하고 있는 499개의 문장을 선택하여 사용하였다.

3.2.2 환경변수 조정(development data) 및 평가(test data) 데이터

시스템의 성능을 평가하기 위하여 NTCIR-7 Workshop에서 학습 데이터로 제공한 말뭉치에서 241개 문서를 사용하였다. 평가 데이터 중에서 1386개의 의견문장만을 대상으로 하여 정확률을 측정하여 평가 결과로 하였다.

각 시스템의 임계값을 결정하기 위하여 별도의 환경변수 조정 데이터를 사용하지 않고 평가 데이터에 N-fold cross validation을 사용하였다. N-fold cross validation은 환경변수를 조정하기 위한 별도의 데이터가 없을 경우 평가 데이터를 무작위적으로 N개로 나눈 후에 각 1개의 데이터를 평가하는데 있어서 나머지 N-1개의 데이터를 환경변수 조정 데이터로 삼아 환경변수를 조정하는 방법이다. 이번 논문에서는 3-fold와 6-fold를 적용하였다.

---

[2] http://nlp.stanford.edu/software/lex-parser.shtml

[3] http://homepages.inf.ed.ac.uk/s0450736/maxent_toolkit.html

4. 실험 결과 및 분석

System1은 3.1에서 설명한 Baseline을 구현한 것이다. System2는 System1에 Self-training을 적용한 것이다. System3는 System1과 동일하게 ME 모델을 학습하는데 있어서 F_1~F_{18}의 자질뿐만 아니라 2.1에서 설명한 단어의 의견주체점수(F_{19})를 자질로서 추가하여 시스템을 만든 것이다. System4는 System3에 Self-training을 적용하여 만든 시스템이다.

표 2 N-fold를 사용한 평가 결과

시스템 종류	정확률	
	3-fold	6-fold
System1 (Baseline)	0.3124	0.3697
System2 (Baseline + self-training)	0.3853	0.3733
System3 (Baseline + F_{19})	**0.4114**	**0.4134**
System4 (System3 + self-training)	0.3958	0.4074

<표 2>를 보면 단어의 의견주체점수를 사용한 것이 의미 있는 성능향상을 가져오는 것을 System1-System3을 비교함으로 알 수 있다. Self-training의 사용 유무나 3-fold, 6-fold 데이터에 상관 없이 모든 실험한 조건에서 성능이 향상되었다. Self-training이 없이도 성능이 향상 되었는데 이는 '전문가', '학생', '사람들', '보고서'와 같은 일반명사나 '우리', '그들'같은 인칭대명사가 MPQA, NTCIR-7 말뭉치 양쪽에서 공통적으로 사용되었기 때문이다.

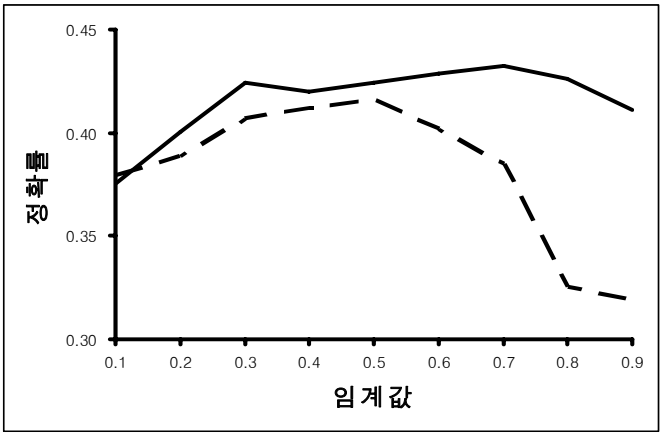

그림 2 임계값에 따른 정확률의 변화
(점선: 환경변수 조정 데이터, 실선: 평가데이터)

Self-training의 경우 <표 1>에 나온 자질만을 사용한 경우에는 성능의 향상이 있었으나(System1-System2) 새로운 자질을 사용한 경우에는 오히려 성능에 저하가 있었다(System3-System4). 이는 부적합한 임계값을 사용하였기 때문이라고 생각하였다. 추측을 확인하기 위해 NTCIR-7 3-fold를 적용하여 나누어진 데이터 중 하나를 선택하고 그 나머지에 해당하는 환경변수 조정 데이터를 사용하여 System3, System4에 대해서 각 임계값에 따른 성능을 평가해 보았다. <그림 2>는 그 결과이다. <그림 2>를 보면 환경변수 조정 데이터가 최고 성능을 보이는 임계값인 0.5에서 평가 데이터의 성능이 최고가 아님을 알 수 있다. 이 결과를 바탕으로 System3, System4를 3-fold, 6-fold에 대해서 환경변수 조정 데이터에서 최고성능이 아니라 두 번째 성능을 보이는 임계값으로 평가를 해 보았다.

표 3 변경된 임계값을 사용한 결과

시스템 종류	정확률	
	3-fold	6-fold
System3	0.4235	0.4057
System4	**0.4301**	**0.4186**

<표 3>에 나온 것과 같이 N-fold를 사용할 때 환경변수 조정 데이터에서 가장 좋은 성능을 보이는 임계값이 아니라 두 번째 높을 때의 임계값을 사용하였을 때에는 System4가 3-fold에서 0.4301, 6-fold에서 0.4186으로 더 높은 성능을 내었다. 임계값에 따라 성능에 차이가 나는 이유는 Self-training을 하는데 있어서 평가 데이터의 N-fold된 문서만을 사용하여 비교적 적은 양의 데이터가 Self-training에 사용되었기 때문으로 생각된다.

5. 요약 및 결론

본 논문에서는 의견의 주체를 찾기 위해서 각 단어에 의견의 주체로 사용될 가능성에 대한 점수를 부여하였다. 이 점수가 각 문서마다 다른 단어와 어구가 의견의 주체로 사용되어 적은 양의 학습데이터에 제약되는 문제를 극복하기 위해 Self-training을 적용하였다.

실험 결과 두 가지 방법 모두 전체 성능을 높이는 것을 알 수 있었으나 임계값에 따라 성능이 크게 변하는 단점이 있었다. 임계값의 영향이 적은 모델을 사용하는 방법에 대한 연구가 추가적으로 수행되어야 할 것이다. 또한 본 논문에서는 단어에 점수를 부여하는 방법이나 단어의 점수를 명사구의 점수로 변환하는 방법에 있어서 개별 단어나 명사구의 표층적인 정보만을 이용하였다. 이 부분에 있어서 명사구의 지배소 정보나 문법적인 정보, 의존관계 정보를 더 이용하여 성능을 높일 여지가 있을 것이다.

감사의 글

본 논문은 2009년도 두뇌한국21사업과 지식경제부 및 정보통신 진흥연구원의 정보통신선도기반기술개발사업의 지원으로 수행되었습니다.

참고문헌

[1] Veselin Stoyanov and Claire Cardie. Toward Opinion Summarization: Linking the Sources. In Proceedings of the Workshop on Sentiment and Subjectivity in Text. pp. 9-14. 2006.

[2] Steven Bethard, Hong Yu, Ashley Thornton, Vasieleios Hativassiloglou, and Daniel Jurafsky. Automatic extraction of option propositions and their holders. In Proceedings of AAAI Spring Symposium on Exploring Attitude and Affect in Text. pp. 22-24. 2004.

[3] Youngho Kim and Sung-Hyon Myaeng . Opinion Analysis based on Lexical Clues and their Expansion. In Proceedings of the NTCIR-6 Workshop. pp. 308-315. 2007.

[4] Soo-Min Kim, Eduard Hovy. Extracting Opinions, Opinion Holders, and Topics Expressed in Online News Media Text. In Proceedings of the Workshop on Sentiment and Subjectivity in Text. pp. 1-8. 2006

[5] Yejin Choi, Claire Cardie, Ellen Riloff and Siddharth Patwardhan. Identifying Sources of Opinions with Conditional Random Fields and Extraction Patterns. In Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language. pp. 355-362. 2005.

[6] Soo-Min Kim and Eduard Hovy. Identifying and Analyzing Judment Opinios. In Proceedings of the Human Language Technology Conference of the North American Chapter of the ACL. pp. 200-207. 2006.

[7] Youngho Kim, Yuchul Jung and Sung-Hyon Myaeng. Identifying Opinion Holders in Opinion Text from Online Newspapers. In Proceedings of IEEE International Conference on Granular Computing. pp. 699-702. 2007.

[8] Adam L. Berger, Stephen A. Della Pietra, Vincent J. Della Pietra. A Maximum Entropy Approach to Natural Language Processing. Computaitional Linguistics. 22(1). pp. 39-71. 1996.

[9] Bo Pang, Lillian Lee, Shivakumar Vaithyanathan. Thumbs up? Sentiment Classification using Machine Learning Techniques. Proceedings of EMNLP. pp. 79-86. 2002.

[10] Jungi Kim, Hun-Young Jung, Sang-Hyeob Nam, Yeha Lee, Jong-Hyeok Lee. English Opinion Analysis for NTCIR7 at POSTECH. In Proceedings of NTCIR-7 Workshop Meeting. pp. 241-246. 2008.

[11] 김병수, 이용훈, 이종혁. 비지도 학습을 기반으로 한 한국어 부사격의 의미역 결정. 정보과학회논문지 : 소프트웨어 및 응용. 34(2). pp. 112~122. 2007.

[12] Janyce Wiebe, Theresa Wilson, and Claire Cardie. Annotating expressions of opinions and emotions in language. Language Resources and Evaluation, volume 39, issue 2-3, pp. 165-210. 2005