SVMs을 이용한 중국어 최장 명사구 자동 식별

윤 창호[1] 이 금희[2] 정 유진[2] 김 동일[3] 이종혁[2]
포항공대 정보통신대학원 정보처리학과[1] 포항공대 컴퓨터공학과[2] 중국연변과학기술대학 언어공학연구소[3]
첨단정보기술 연구센터
{yustian[1], ljj[2], prizer[2], dongil[3], jhlee[2]}@postech.ac.kr

Identification of Chinese Maximal Noun Phrase on Different Context Size Settings Using SVMs

Changhao Yin[1] Jinji Li[2] You-jin Chung[2] Dong-ill Kim[3] Jong-hyeok Lee[2]
Dept. of Graduate School for Information Technology, POSTECH[1], Dept. of Computer Science &Engineering, POSTECH[2], Language Engineering Institute, YUST[2] China, and Advanced Information Technology Research Center(AITrc)

요약

중국어의 명사구는 기본 명사구, 최단 명사구, 최장 명사구 등으로 분류할 수 있다. 최장 명사구를 잘 식별해 낼 수 있다면 구문 분석의 복잡도를 크게 낮추고 구문분석의 성능을 향상시킬 수 있다. 각 단어는 시작 태그(O), 종결 태그(C), 한 단어로 이루어진 구 태그(S), 그 외의 태그(N) 등 4가지로 태깅된다. 본 논문은 서로 다른 윈도우 크기(window size)에 기반한 5가지 SVMs 학습 모델을 구축하고 시스템 합성 방법을 이용하여 중국어 최장 명사구 식별에서 85.17%의 정확률을 보여줬다.

1. 서론

영어에서 기본 명사구는 구 안에 다른 명사구를 내포하지 않으면서, 연속되어 있고, 문법적으로 연관되어 있는 비-중첩, 비-내포인 명사구를 뜻한다[1]. Abney가 처음으로 구 단위묶음(chunk)을 구문분석을 위한 전처리 단계로 구문분석을 두 단계로 나누어 해결하는 방법을 제시하였다[1]. 영어와는 달리 중국어는 아래와 같은 두 가지 언어적 특성을 갖고 있다. 첫째로는 단어의 형태적인 변형 혹은 전치사의 도움 없이 직접 다른 단어와 문법적인 관계를 갖는다. 예를 들면,

[1]我/r 的 书, 秀丽/a 的 山河, 移动/v 的 目标
wo/r de shu, xiuli/a de shanhe, yidong/v de mubiao

'我(wo)', '秀丽(xiuli)', '移动(yidong)' 는 각각 대명사, 형용사, 동사 이지만 조사 '的' 와 형태적인 변형을 하지 않고 직접 결합하여 명사구를 형성하고 있으며, 따라서 구의 중의성 문제를 발생시키고 있다. 둘째로는 영어에서 기본 명사구의 정의와는 달리 기본 명사구가 또 다른 명사구를 내포해야 할 필요가 있다. 예를 들면,

公司/[项目/经理], [皮/领子]大衣, [电脑/维修]/书
gongshi/[xiangmu/jingli],[pi/lingzi]/dayi, diannao/weixiu]/shu

는 명사구로서 문장에서 하나의 문법적인 단위로 볼 수 있다. 위의 예문 중에서 괄호 안에 묶여있는 명사구는 큰 단위 안에 내포되어 있음을 볼 수 있다. 이것은 단어들의 단순한 나열이 구를 형성할 수 있는 중국어의 언어적 특성에서 기인된 것이며 중국어 명사구의 정의에 내포를 허용해야 함을 의미한다. 이런 특성에 따라 중국어 명사구는 기본 명사구, 최단 명사구, 최장 명사구 등 여러 가지로 분류 될 수 있으며 문제의 난이도와 구문분석에 대한 기여도면에서 차이를 보인다.

[1] 중국어 품사 r: 대명사, a: 형용사, v: 동사

[표1.1] 3가지 유형의 중국어 명사구

최단 명사구	放/在/**桌子**/上/的/**我**/的/**书包**
기본 명사구	放/在/**桌子**/上/的/**我的**/**书包**
최장 명사구	**放在桌子上的我的书包**
	fang/zai/zhuozi/shang/de/wo/de/shubao

위 예에서 최단 명사구는 식별이 쉬운 반면에 구문분석에 대한 기여도가 작고, 기본 명사구는 구에서 어디까지를 기본으로 정하느냐에 따라 문제의 복잡도가 다르기 때문에 기본 명사구에 대한 일관된 정의가 없다. 최장 명사구는 문장에서 문법적으로 명사구의 작용을 발휘하는 가장 큰 단위를 명사구로 보기 때문에 구문분석에 대한 기여도가 가장 크다. 반면에 식별할 때 참조해야 하는 문맥의 길이가 다른 명사구에 비해 커야 하는 단점이 있다. 본 논문은 중국어 최장 명사구 식별에서 주변 문맥의 길이가 주는 영향을 살펴보며 서로 다른 문맥 길이에 기반 된 5가지 SVMs 학습 모델을 구축하고 시스템 합성 방법을 통하여 실험을 하였다.

2. 관련 연구

Erik[4]는 [2] IOB 등 구표기방법으로 각각의 classifier마다 5종류의 학습 모델을 만든 후 그 결과에 가중치를 부여하는 방법으로 내부시스템합성(internal-system compbination)하고, [3]MaxEnt 등 7종류의 classifier로 학습 모델을 만든 후 역시 같은 방법으로 외부 시스템합성(external-system combination)하여 영어 기본 명사구 식별에서 최고성능을 보여주었다. Taku[5]는 SVMs으로 서로 다른 구표기방법에

[2] IOB: Inside, Outside, Beginning of a Consecutive Chunk
IOB1: Initialized with B, others same with IOB
IOE: Inside, Outside, End of Chunk followed by another Chunk
IOE1: Initialized with E, others same with IOE1
O+C: Open and Close word of Chunk

[3] MaxEnt, ALLis, IGTree, SNoW, MBSL, C5.0

기반한 5종류의 학습모델을 만들고 내부 시스템합성하는 방법을 제안하였다. 시스템합성 테크닉이 단일 시스템보다 좋은 성능을 보여주는 이유를 살펴보면 몇 개의 시스템을 동일한 문제에 적용했을 때 각 시스템의 오류 결과는 서로 다르기 때문이다. 예하면 시스템 A의 결과는 오류이지만 만일 시스템 B, C, D의 결과가 정확하다면 majority voting 등 가중치 부여 방법으로 투표를 한다면 시스템 A의 오류를 보완할 수 있는 것이다.

3. 자질 선택 및 SVM 학습 모델 구축

3.1 명사구 표기방법

명사구 식별을 각 단어에 대한 구 클래스 태깅(tagging) 문제로 변환 시키는 방법은 Ramshaw 와 Marcus[7] 가 처음으로 시도하였다. 본 논문에서는 OCSN 표기 형식을 사용하여 중국어 최장 명사구 식별을 진행하였다. 최장 명사구는 구 내부에 다른 명사구가 내포되지 않기 때문에 위의 네 가지 태그로 쉽게 표기가 가능하고 기존의 영어에서 사용 되였던 기본 명사구 식별방법 들을 적용해볼 수 있는 장점을 갖고 있다.

[표3.1] 명사구 표기 방법

태그	설명
O	명사구의 시작
C	명사구의 종결
S	한 개의 단어로 이루어진 명사구
N	그 외의 모든 경우

OCSN 표기 방법의 특성은 명사구의 경계(boundary) 표기에 초점을 맞췄다는 것이다. 구의 시작과 종결이 아닌 즉 구안에 포함되어 있는 단어를 구가 아닌 단어와 구별하지 않고 모두 N으로 태깅하였다.

3.2 자질 선택

학습 자질로는 현재 태깅 하려는 단어의 의미코드[8] 및 품사 정보와 윈도우 크기 N개 단어의 품사 정보를 참조하였다. 의미코드는 《同义词词林》[7]의 분류체계 중 1428 종류의 소분류를 이용하였다. N 은 3, 5, 7, 9, 11 등 5가지로 실험을 진행하였다. 의미코드를 이용함에 있어서 학습 벡터의 차원(dimension)을 줄이기 위해 Chi-Squared[9] 방법을 사용하였다.

$$W = T(tp,(tp+fp)Ppos) + T(fn,(fn+tn)Ppos) + T(fp,(tp+fp)Pneg) + T(tn,(fn+tn)Pneg)$$

$$\textbf{Where } T(count,expect) = (count-expect)^2/expect$$

[그림3.2][4]chi-squared feature selection

본 논문에서는 임계 가중 치를 30으로 정하여 O, C, S, N 각 태그 별 상위 100개의 의미코드를 선택하여 학습 자질로 사용하였다.

태그	의미 코드
O	[5]Ed611,Di021,Kd011,뭳 ..Hj672,Bn042
C	Ab02c,Db091,Fa201,.....Bn111,Bm041
S	Aa021,Aa041,Aa031,뭳 ..Dn051,Bn031
N	Kd011,Aa021,Aa041,뭳 .Df011,Bn031

[표3.3] O, C, S, N 상위 100개의 의미 코드

위의 표에서 보면 S 태그와 N 태그가 선택한 의미코드 분포가 비슷함을 알 수 있다. 이는 S 태그와 N 태그 사이에 많은 중의성이 존재하며 S 태그의 정확률이 상대적으로 낮을 것임을 추측할 수 있다.

3.3 LIBSVMs

기본적으로 SVMs 는 이진 classifier이다. 때문에 O, C, S, N 같은 다중 클래스 문제를 해결하기 위해서는 SVMs의 보완이 필요하다. 첫 번째 (One Vs All Others) 방법으로는 K개의 클래스를 식별하기 위해서 K개의 classifier를 만들고 그 중 한 클래스를 나머지와 구분하는 것이다. 두 번째(One VS One)방법으로는 (K-1)*K/2개의 classifier를 만들고 모든 쌍에 대해 식별을 진행하고 가중치 부여를 통하여 최종 결정을 내리는 방법이다. 두 번째 방법이 비록 classifier를 많이 만들지만 SVMs 의 학습 시간인 $O(n^2) \sim O(n^3)$에 반해 상대적으로 적은 부하이고 One vs One가 One vs All Others 에 비해 좋은 성능을 보여주고 있기 때문에 본 논문에서는 LIBSVM[10]이 제공하는 one vs one 방법을 사용하였다. LIBSVM은 통합 SVM 소프트웨어로서 기본 알고리즘은 SVMLIGHT의 알고리즘을 사용하고 있다. SVMLIGHT가 이진 classifier인 반면에 멀티 클래스를 지지하며 데이터 스케이링(scaling)과 unbalanced 클래스에 대한 penalty 부가 기능을 갖고 있으며 기본 커널은 RBF 커널을 사용하고 있다.

3.4 5종류의 SVM 학습 모델 구축

관계 절이 포함되어 있는 비교적 긴 명사구를 식별하기 위해서는 참조해야 하는 윈도우 크기가 상대적으로 커야 하고 기본 명사구 같은 작은 명사구는 작은 윈도우 크기에서도 식별이 가능하다. 따라서 윈도우 크기에 기반한 서로 다른 SVM 학습 모델을 만든다면 이들 각각의 처리 능력이 다를 것이며 시스템 합성을 한다면 보다 좋은 성능을 보일 것이다. 이런 생각에 기반하여 윈도우 크기 3, 5, 7, 9, 11에서 5가지 다른 학습 모델을 구축하였다. 학습 자질로 사용되는 품사정보는 대 분류로 명사(n), 동사(v), 전치사(p) 등 20가지를 사용하였다. 매 단어는 자질선택 과정을 통하여 [6] 400+N*20 크기의 벡터로 표현되어 SVMs의 학습 데이터로 사용된다.

3.5 Weighted Voting 을 이용한 시스템 합성

시스템합성 방법에는 각 시스템에 균일한 가중치를 부여하는 균일가중치 부여 방법과 시스템의 정확률, 재현률 등 성능을 참조하여 가중치를 부여하는 방법이 있다. 본 논문에서는 윈도우 크기 3, 5, 7, 9, 11 등 5개의 시스템 성능을 평가하고 각 시스템에서 O, C, S, N 태그의 정확률을 가중치로 사용하였다.

---

[4] tp: true positives fn: false negatives
fp: false positives tn: ture negatives
pos: tp + fn neg: fp + tn
Ppos = pos/all Pneg = neg/all

[5] Ed611: 这个(이것) Ab02c: 男女(남녀) Aa021: 我(나) Kd011: 的(의)
Bn042: 建筑物(건축물) Bm041: 材料(재료) Bn031: 房间(방) Df011: 意识(의식)

[6] N: 윈도우 크기 3, 5, 7, 9, 11

4. 실험 및 결론

4.1 학습 말뭉치

학습 코퍼스는 Balanced Corpus of Modern Chinese TreeBank[11] 이고 크기는 11000 문장에 약 30만 단어이다. 테스트 문장은 북경대학 트리뱅크 중 임의로 500문장을 선택하였으며 약 5천 단어이다. 학습 코퍼스 중 약 70%의 단어가 의미코드가 있고 테스트 문장은 약 90%의 단어가 의미코드를 가지고 있다.

4.2 실험 결과(Open Test)

[표4.2.1] 전체 태그 정확률

윈도우 크기	정확 률
3	81.82%
5	**82.64%**
7	82.58%
9	82.54%
11	82.44%

[표4.2.2] 각 태그 정확률

윈도우 크기	O	C	S	N
3	**79.17%**	79.26%	62.35%	85.69%
5	77.23%	80.76%	**63.98%**	86.21%
7	77.88%	80.58%	64.12%	**85.99%**
9	77.56%	**80.92%**	63.82%	85.98%
11	77.38%	80.63%	63.82%	85.90%

[표4.2.3] 각 태그 재현률

윈도우 크기	O	C	S	N
3	65.33%	56.26%	75.59%	93.70%
5	69.14%	61.26%	78.86%	94.16%
7	67.99%	60.25%	78.86%	94.22%
9	66.84%	59.67%	77.45%	94.29%
11	66.14%	59.12%	77.45%	94.22%

[표4.2.4] 시스템 합성을 한 후 정확률

전체 태그	O	C	S	N
85.17%	**79.58%**	**84.24%**	**70.59%**	**88.09%**

4.3 결론

전체 태그의 정확률은 윈도우 크기 5에서 최고의 성능을 보이며 그 이상에서는 오히려 성능이 떨어 짐을 알 수 있다. 각 태그 별 정확률은 각 태그마다 수렴하는 문맥의 길이가 서로 틀리 다는 것을 알 수 있다. O 태그는 문맥 길이 3에서, C 태그는 9에서, S 태그는 5에서, N 태그는 7에서 최고의 성능을 보임을 알 수 있다. 이는 각각 태그 별 정확 률 을 가중치로 부여하여 시스템 합성을 했을 때 성능이 향상 될 수 있음을 보여주고 있다. 시스템 합성 후 전체 태그 정확률은 단일 시스템에서의 최고의 성능인 82.64%에서 85.17%으로 향상 되었고 각 태그 별 정확 률 도 따라서 향상 되었음을 볼 수 있다. 재현률을 살펴보면 O, C, S 태그가 상대적으로 낮고 N 태그의 재현률이 월등하게 높음을 볼 수 있다. N 태그의 재현 률 이 다른 태그에 비해 높다는 것은 명사 구의 경계는 대부분 정확하게 찾았지만 이들 경계 단어에 대한 O, C, S 태그의 정확률이 높지 않음을 의미한다. 그러나 문법적 규칙에 기반한 후처리 작업으로 이 문제를 해결 할 수 있을 것이다.

5. 향후 작업

(1) O, C, S 태그의 재현률의 향상과 O 태그와 C 태그의 불일치에 대한 후처리 작업이 필요하다.

(2) 자질로 좌우 단어의 품사 정보를 사용하였는데 이는 중국어 최장 명사구의 식별에서 낮은 재현률을 보이고 있다. 품사의 좀 더 세분화된 정보 혹은 의미 코드의 사용이 필요하다. 학습데이터의 차원을 줄이기 위해서는《同义词词林》[7]의미코드분류체계 중 94개의 중분류를 사용하는 것도 바람직하다.

(3) 태그 표기법에 있어서 본 논문에서는 명사구의 경계 태그 식별에 초점을 맞추었기 때문에 명사구안에 포함된 시작과 끝이 아닌 중간 태그와 명사구 밖의 태그를 구분하지 않고 모두 N으로 태깅 하였다. 이 두 경우에 대한 구별이 필요하다.

(4) Decision tree, Maximum Entropy Model, Memory Based Learning 등 다른 학습 방법을 사용하여 외부 시스템합성 방법으로 성능을 향상을 할 필요가 있다.

6. 참고문헌

[1]Steven P. Abney,"*Parsing by Chunks*", In Principle-Based Parsing, pages 257-278. Kluwer Academic Publishers, Dordrecht, 1991

[3]Zhou Qiang, Sun Maosong and Huang Changning "*Automatically Identify Chinese Maximal Noun Phrase*", 1998

[4]Erik F.Tjong Kim Sang &al "*Applying System Combination to Base Noun Phrase Identification*" In: Proceedings of COLING 2000, 857-863,2000

[5]Taku Kudo and Yuji Matsumoto,"*Chunking with Support Vector Machines*", In NAACL, 2001

[6]Lance A. Ramshaw and Mitchell P. Marcus, "*Text Chunking using Transformaion-Based Learning*". In Proceedings of the Third ACL Workshop on Very Large Corpora.Cambridge, MA, USA,1995

[7]梅家驹，竺一鸣 & al《同义词词林》，上海辞书出版社，上海，1983

[8]Georage Forman,"*An Extensive Empirical Study of Feature Selection Metrics for Text Classification*" In: **The Journal of Machine Learning Research Volume 3 , March 2003**

[9]Urich. H.-GKreBel.*"Pairwise Classification and Support Vector Machines*□In Advance in Kernel Mathods.MIT Press

[10]Chin-Chung Chang and Chih-Jen Lin. "*a Library for Support Vector Machines*" , http://www.csie.ntu.edu.tw/~cjlin/libsvm/ November 12, 2003

[11]Academia Sinica Balanced Corpus of Modern Chinese, http://www.sinica.edu.tw/