어휘정보와 통사정보를 모두 이용한 문서분류

박성배 O 장병탁

서 울 대 학 교 컴 퓨 터 공 학 부

{sbpark,btzhang}@scai.snu.ac.kr

Text Categorization Using Both Lexical Information and Syntactic Information

Seong-Bae Park O Byoung-Tak Zhang

School of Computer Science and Engineering, Seoul National University

요 약

현재 이용가능한 대부분의 자동문서분류 시스템의 가장 큰 문제는 문서에 포함된 단어 사이의 통사 정보는 무시한 채, 각 단어의 분포만 고려한다는 점이다. 하지만, 통사 정보도 문서 분류를 위해 매우 중요한 정보 중의 하나이다. 본 논문에서는 문서에 나타난 어휘 정보와 함께 통사 정보를 함께 고려하는 자동문서분류 방법을 제시한다. Reuters-21578 말뭉치에 대한 문서분류 실험결과, 제시된 방법은 어휘정보만 사용하는 방법과 통사정보만 사용하는 방법 모두보다 높은 성능을 보인다. 이 말뭉치에 대해서, 어휘정보만으로 학습된 Support Vector Machine 으로 약 97%의 매우 높은 정확도를 얻을 수 있음에도 약 0.63%의 추가적인 성능 향상이 있었다.

1. 서론

문서분류(text categorization) 문제는 레이블이 있는 학습 문서 집합으로부터 추출된 정보에 기초하여 향후에 주어지는 레이블이 없는 문서를 미리 정해진 범주로 분류하는 것이다. 자동문서분류는 점점 더 많은 양의 문서가 전자화됨으로써 실용적인 측면에서 그 중요성이 더욱 부각되고 있다.

하지만, 자동문서분류에 적용된 대부분의 기계학습(machine learning) 알고리즘들은 *TF•IDF* 와 같이 문서에 포함된 단어의 분포를 기본정보로 사용하는 반면에, 통사 정보(syntactic information)와 같은 또 다른 중요한 정보는 무시한다. 통사정보로부터 얻을 수 있는 문체 특징이 문서분류를 위한 중요한 요소임[6]에도 불구하고, 이런 정보가 너무 복잡하며 형식 정의(formal definition)가 없어서 잘 사용되지 않아왔다. 또한, 불행하게도 현재의 자연언어처리 기술도 통사 분석에 있어서 만족할 만큼 정확한 결과를 제공해 주지 못하는 실정이다. 따라서, 통사정보를 이용하기 위해 완전 구문 분석(full parsing)을 하기 보다는 필요한 통사 분석에 필요한 충분한 정보를 줄 수 있는 문서 단위화(text chunking)[2]를 사용하는 것이 더 현실적이다.

본 논문에서는 기존의 어휘정보에 통사정보를 추가로 활용함으로써 보다 높은 성능을 보이는 자동문서분류 방법을 제시한다. 주어진 문서에 대해, 어휘정보를 학습하는 분류기(classifier)와 통사정보를 학습하는 분류기를 각각 따로 학습한다. 본 논문에서는 분류기로 Support Vector Machines 을 사용한다. 레이블이 없는 문서의 레이블을 결정할 때, 두 분류기의 예측이 서로 다르면 보다 높은 신뢰도로 레이블을 추정한 분류기의 예측 결과를 따른다. Reuters-21578 말뭉치에 이 방법을 적용해 본 결과, 어휘정보만 사용한 분류기보다 평균적으로 약 0.63% 정도의 정확도 향상이 있었음을 발견하였다. Reuters-21578 말뭉치에 대해, SVM 이 약 96.89%의 매우 높은 정확도를 보이므로, 0.63%의 성능 향상은 매우 의미있는 결과이다.

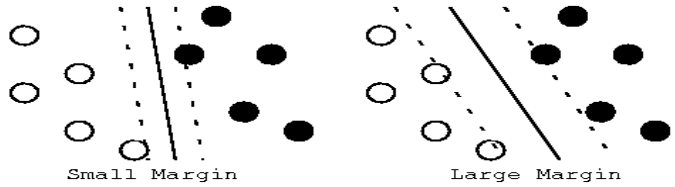

그림 1. SVMs 의 두 초 평면.

2. SVM 을 이용한 문서분류

Support Vector Machines 은 이진분류 문제를 풀기 위한 알고리즘이다. SVMs 는 일반적으로 높은 일반화 성능을 보임으로써 여러 가지 패턴 인식 문제, 특히 자동문서분류 문제[5]에 성공적으로 적용되어 왔다.

다음과 같이 주어진 학습 데이터에 대해서,

$(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N) \quad \mathbf{x}_i \in R^n, \quad y_i \in \{+1, -1\}$

SVMs 은 초평면 $(\mathbf{w} \cdot \mathbf{x}) + b = 0$ 을 학습한다. SVMs 은 학습 데이터를 두 클래스로 정확하게 분류하는 최적의 초평면을 찾는다. 최적을 정의하기 위해서 margin 이라는 개념을 도입한다. Margin 은 그림 1 에서 두 점선의 거리를 나타내며, SVMs 은 margin 을 최대화하도록 초평면을 학습한다. 두 점선과 margin(d)은 다음과 같이 씌여질 수 있다.

$$(\mathbf{w} \cdot \mathbf{x}) + b = \pm 1, \quad d = 2 / \| \mathbf{w} \|$$

따라서, SVM 학습은 최적화 문제로 생각될 수 있다. 즉, $y_i[(\mathbf{w} \cdot \mathbf{x}) + b] \geq 1$ 이라는 제약 하에서 $\|\mathbf{w}\|$를 최소화하도록 $\mathbf{w}$ 와 b 를 찾으면 된다.

3. 어휘정보와 통사정보를 이용한 문서분류

3.1 어휘정보를 이용한 문서분류

어휘정보를 이용한 문서분류에서는 일반적으로 문서를 *TF•IDF* 로 표현한다. *TF•IDF* 에서는 주어진 문서 d 의 단어 w_i 의 가중치 $d(i)$는 다음과 같이 정의된다.

$$d(i) = \mathrm{TF}(w_i, d) \cdot \mathrm{IDF}(w_i)$$

여기서, TF 는 단어 w_i 가 문서 d 에 나타난 횟수이고, IDF 는 다음과 같이 정의된다.

$$IDF(w_i) = \log\left(\frac{|D|}{DF(w_i)}\right)$$

document frequency DF 는 단어 w_i 가 나타난 문서의 개수를 뜻한다.

본 논문에서는 bow 툴킷[7]을 사용하여 문서들을 *TF•IDF* 로 표현하였다. 이 툴킷은 Porter's stemming algorithm 으로 문서를 벡터로 표현하고 stoplist 를 사용해 불용어들을 제거한다.

3.2 문서분류를 위한 통사정보

통사정보가 자동문서분류를 위한 좋은 자질이 될 수 있음에도 불구하고, 불행하게도 현재의 자연언어처리 기술 수준으로는 완벽한 구문분석이 불가능하다. 따라서, 전체 파싱을 해서 통사정보를 얻기 보다는 보다 정확한 결과를 얻을 수 있는 단위화(chunking)을 통해 통사정보를 얻는 것이 현실적이다.

90 년대 이후 많은 연구가 진행되어 온 단위화는 비교적 간단한 분석을 통해 주어진 문장을 중첩이 없는 구절들로 나누는 작업이다. 본 논문에서는 주어진 문서의 각 문장을 단위화하기 위해서, 별도로 단위화를 위한 SVMs 을 학습시켰다. CoNLL-2000 shared task[1]의 데이터를 이용하여, 단위화를 위한 SVMs 를 학습시켰다. CoNLL-2000 데이터 집합이 23 개의 분류 표지를 가지지만, 본 논문에서는 분포 비율이 가장 높은 NP, VP, PP, O 만을 고려 대상으로 삼았다. O 를 제외한 단위 표지 X 는 다시 각각 B-X 와 I-X 등으로 나누어진다. B-X 는 단위 X 의 *시작*을 나타내고, I-X 는 단위 X 의 *계속*을 나타낸다. SVMs 가 이진분류 문제에만 적용될 수 있으므로, 각 표지에 대해서 각각의 분류기를 학습시켜야 한다. 하지만, SVM 의 경우에는 이런 복수개의 분류기를 학습시키는 것보다 쌍 분류(*pairwise classification*) 방식으로 학습시키는 것이 성능이 훨씬 좋으므로 본 논문에서는 후자의 방식을 취했다.

표 1 은 문서분류를 위해 본 논문에서 사용된 통사정보 자질이다. 이 자질들은 단위화된 문서로부터 기계적으로 계산할 수 있는 수치값을 갖는다.

Stamatatos 등은 문서의 문체 표지(style markers)들이 서로 다른 종류의 문서를 분류하거나 주어진 문서의 저자(author)를 자동으로 분류하는데 효과적으로 사용될 수 있음을 보였다[1]. 이는 각 종류의 문서마다 독립적인 통사적 문체가 있음을 뜻한다. 표 1 은 문서분류를 위해 사용된 통사 자질이다. 상위 4 개의 자질은 각 문법 구절(grammatical phrase)이 얼마나 자주 쓰이는지를 나타내고, 하위 4 개의 자질은 각 구절이 얼마나 길게 쓰였는지를 나타낸다. 따라서, 어떤 저자가 문서를 쓰는 특징이 잘 반영될 수 있으며, 특히 문체가 다르게 나타날 수 있는 문서들 사이에서는 분류를 위한 매우 좋은 자질이 될 수 있다.

Feature	Description
SF1	Detected *NP*s / total detected chunks
SF2	Detected *VP*s / total detected chunks
SF3	Detected *PP*s / total detected chunks
SF4	Detected *O*s / total detected chunks
SF5	Words included in *NP*s / detected *NP*s
SF6	Words included in *VP*s / detected *VP*s
SF7	Words included in *PP*s / detected *PP*s
SF8	Words included in *O*s / detected *O*s

표 1. 문서분류를 위해 사용된 통사정보 자질 (feature).

3.3 두 문서분류를 위한 SVM 의 결합

하나의 문서에 대해 서로 다른 관점으로 학습된 두 SVMs 가 있으므로 이 두 분류기를 결합하여 문서를 분류한다. 두 분류기가 주어진 문서에 대해서 서로 다르게 레이블을 추정했을 때, 두 분류기 중 더 큰 margin 을 가지고 결정한 분류기 쪽의 결정을 따른다. SVMs 에서 margin 이 크다는 뜻은 초평면에서 더 멀리 떨어져 있다는 뜻이므로, 더 큰 신뢰도를 가진다는 뜻이다. 따라서, 우리는 margin 이 큰 쪽의 분류기의 결정을 더 신뢰할 수 있다.

4. 실험

4.1 실험 데이터

본 논문에서는 실험 데이터 집합으로 Carnegie 그룹이 1987 년 로이터 뉴스에서 수집한 Reuters-21578 말뭉치를 사용하였다. 이 말뭉치는 135 개의 토픽을 가지는데, 우리는 이중 주요한 10 개의 토픽만 실험대상으로 삼았다 (표 2). 이 말뭉치를 학습집합과 테스트집합으로 나누는 데에는, 세 종류의 방법이 있는데, "ModLewis", "ModApte", "ModHayes"가 그들이다. 이 중에서, 우리는 가장 널리 쓰이는 "ModApte"를 사용하였다. 따라서, 9,603 개의 학습 문서와 3,299 개의 테스트 문서를 사용하였다.

어휘정보를 사용하는 SVMs 를 위해, bow 툴킷을 사용하여 각 문서를 *TF•IDF* 로 표현하였다. 통사정보를 사용하는 SVMs 를 위한 데이터를 만들기 위해서, CoNLL-2000 shared task 데이터집합과 Brill's Tagger[3]를 사용하였다. Reuters-21578 말뭉치가 가공하지 않은 말뭉치이므로, 이 말뭉치에 있는 모든 문장을 단위화된 형태로 만들어야 한다.

우선, 문서에 있는 각 단어에 대해, Brill's Tagger 를 사용하여 품사를 결정하였다. 이 품사 태거는 확률 품사 태거에 의해 생길 수 있는 오류를 제거할 수 있는 변환 규칙 집합을 학습한다. 따라서, 다른 확률 품사 태거에 비해 언어학적 관점에서 복잡한 지식을 표현할 수 있는 장점을 가진다.

각 단어에 대한 품사를 결정한 후, CoNLL-2000 데이터 집합으로 학습된 단위화를 위한 SVMs 를 사용하여 단위화 레이블을 결정하였다. 이 데이터로부터 각 문서를 표 1 과 같은 8 차원의 벡터로 표현하였다.

[1] http://lcg-www.uia.ac.be/conll2000/chunking

4.2 실험 결과

표 2 는 *TF•IDF* 자질에 추가로 8 개의 통사 자질을 더하여 하나의 Support Vector Machine 을 학습시킨 후, 테스트 한 결과이다. 'Lexical' 열은 *TF•IDF* 만으로 SVM 을 학습시켰을 때의 결과이고, 'Syntactic'은 각 문서를 표 1 과 같이 표현했을 때의 분류 결과이다. Reuters-21578 의 주요 10 개 토픽에 대해, 어휘정보를 사용한 경우가 통사정보를 사용한 경우보다 정확도가 높게 나타났다. 또한, 두 자질을 더하여 학습했을 때에도 어휘정보만 사용했을 때에 비해 성능개선의 효과가 거의 없었다. 'Earn', 'Corn' 과 'Grain' 토픽에 대해서만 약간 좋아졌을 뿐, 나머지 토픽에 대해서는 성능 개선이 없었다. 'Acq' 토픽에 대해서는 오히려 나쁜 성능을 보였다. 이는 Reuters-21578 말뭉치가 주로 경제 분야에서 추출된 기사들이기 때문에, 토픽에 따라 문체적 특징이 거의 나타나지 않기 때문이다.

Class	Accuracy		
	Syntactic	Lexical	Both
Earn	91.42%	95.09%	95.30%
Acq	78.21%	93.76%	93.73%
Money-fx	94.75%	96.15%	96.15%
Grain	95.48%	95.48%	95.51%
Crude	94.27%	97.00%	97.00%
Trade	96.45%	97.79%	97.79%
Interest	96.03%	97.18%	97.18%
Ship	97.30%	98.15%	98.15%
Wheat	97.85%	98.91%	98.91%
Corn	98.30%	99.12%	99.18%

표 2. 단순히 어휘정보와 통사정보를 더하여 학습했을 때의 문서분류 성능.

어휘정보만 사용한 분류기와 통사정보만 사용한 분류기를 각각 따로 학습시킨 후, 3.3 절에서 제시한 바와 같이 더 큰 margin 을 갖는 분류기의 추정을 믿는 방법으로 두 분류기와 합쳤을 때에는 어휘정보만 사용했을 때보다 높은 정확도를 보였다. 표 3 에서 Increase 열은 어휘정보만 사용했을 때보다 좋아진 정확도이다. 평균적으로 어휘정보만 사용했을 때보다 0.63% 좋아졌으며, 특히 단순히 자질을 더했을 때 나빠졌던 'Acq' 토픽에 대해서, 가장 많이 1.48% 좋아졌다. 'Wheat'나 'Corn' 토픽과 같은 경우에는 어휘정보만 사용했을 경우에도 98.91%와 99.12%라는 높은 정확도를 보였음에도 불구하고 추가적인 정확도 증가가 있었다.

6. 결론

본 논문에서는 어휘정보와 통사정보를 모두 이용하는 자동문서분류 방법을 제시하였다. 대량의 말뭉치로부터 학습한 단위화 모듈을 사용하여 주어진 문서를 단위화한 후, 이를 이용하여 문서분류를 하는 Support Vector Machine 을 학습시켰다. 기존의 어휘정보만 이용하여 학습한 Support Vector Machine 과 함께 사용했을 때, Reuters-21578 말뭉치에 대해 평균적으로 97.52%의 높은 정확도를 얻을 수 있었다. 이 결과는 어휘정보만 사용했을 때보다 0.63% 높은 결과이다.

한 문서에 대해 서로 다른 두 개의 독립된 관점으로 학습을 하였다. 따라서, 제시된 방법은 자연스럽게 Co-Training[8]에 적용될 수 있을 것으로 예상된다. Co-Training 은 레이블이 없는 데이터를 추가적으로 학습하기 위해 두 개의 서로 다른 관점으로 하나의 데이터를 분석하는 방법이다. TREC 의 filtering 문제처럼 일반적으로 문서분류 문제는 수 많은 레이블이 없는 데이터(unlabeled data)를 포함한다. 예를 들어, 웹 페이지를 분류할 때, 도처에 널려 있는 다른 수많은 웹 페이지를 학습에 사용할 수 있다.

Class	Accuracy	Increase
Earn	96.61%	1.31%
Acq	95.21%	1.48%
Money-fx	97.12%	0.97%
Grain	95.51%	0.00%
Crude	97.67%	0.67%
Trade	98.42%	0.63%
Interest	97.67%	0.49%
Ship	98.58%	0.43%
Wheat	99.15%	0.24%
Corn	99.27%	0.09%
Average	97.52%	0.63%

표 3. **제시된 방법의 문서분류 성능.**

감사의 글

이 논문은 한국과학재단(KOSEF)의 첨단정보기술연구센터(AITrc)와 교육부 BK 21 사업에 의하여 지원되었음.

참고문헌

[1] E. Stamatatos, N. Fakotakis, and G. Kokkinakis, "Automatic Text Categorization in Terms of Genre and Author," *Computational Linguistics*, Vol. 26, No. 4, pp. 471-495, 2000.

[2] T. Kodoh and Y. Matsumoto, "Use of Support Vector Learning for Chunk Identification," In *Proceedings of CoNLL-2000 and LLL-2000*, pp. 142-144, 2000.

[3] E. Brill. Rule based tagger. http://www.cs.jhu.edu/~brill/.

[4] T. Joachims. *SVM^{light}* version 3.02. http://www.-ai.cs.uni-dortmund.de/SOFTWARE/SVM_LIGHT/svm_light.eng.html.

[5] T. Joachims, "Text Categorization with Support Vector Machines: Learning with Many Relevant Features," In *Proceedings of ECML 98*, pp. 137-142, 1998.

[6] D. Biber, "Dimensions of Register Variation: A Cross-Linguistic Comparison," *Cambridge University Press*, 1995.

[7] A. McCallum and Andrew Kachites. "Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering." http://www.cs.cmu.edu/~mccallum/bow. 1996.

[8] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," In Proceedings of 11th Annual Conference of Computational Learning Theory, pp. 92-100, 1998.