http://dx.doi.org/10.7236/JIWIT.2012.12.5.243
JIWIT 2012-5-31

단어패턴 빈도를 이용한 단문 오피니언 문서 분류기법의 실험적 평가

An Experimental Evaluation of Short Opinion Document Classification Using A Word Pattern Frequency

장재영*, 김일민**

Jae-Young Chang, Ilmin Kim

요 약 데이터 마이닝의 문서분류 기술에서 발전된 오피니언 마이닝은 이제 국외뿐만 아니라 국내 산업에서 중요한 관심분야로 자리잡아가고 있다. 오피니언 마이닝의 핵심은 문서에서 감정 단어를 추출하여 긍정/부정 여부를 얼마나 정확하게 판별하느냐를 평가하는 것이다. 국내에서도 이에 관련된 많은 연구가 이루어 졌으나 아직 실용적으로 적용할 만큼의 분류 정확도를 보이지 않고 있다. 한국어의 경우 비문법적 표현, 감정단어의 다양성 등으로 인해 문서의 극성을 판별하기가 쉽지 않기 때문이다. 본 논문에서는 문법적 요소를 최대한 배제하고 단어패턴의 빈도만을 고려한 새로운 오피니언 문서 분류기법을 제안한다. 제안된 방법에서는 문서를 단어들의 리스트로 추상화한 후, 패턴들의 빈도를 이용하여 기계학습 알고리즘을 적용한다. 이후에 적절한 스코어 함수를 적용하여 문서의 극성을 판별한다. 또한 제안된 기법의 정확도를 평가하기 위해서 실험결과를 제시한다.

Abstract An opinion mining technique which was developed from document classification in area of data mining now becomes a common interest in domestic as well as international industries. The core of opinion mining is to decide precisely whether an opinion document is a positive or negative one. Although many related approaches have been previously proposed, a classification accuracy was not satisfiable enough to applying them in practical applications. A opinion documents written in Korean are not easy to determine a polarity automatically because they often include various and ungrammatical words in expressing subjective opinions. Proposed in this paper is a new approach of classification of opinion documents, which considers only a frequency of word patterns and excludes the grammatical factors as much as possible. In proposed method, we express a document into a bag of words and then apply a learning algorithm using a frequency of word patterns, and finally decide the polarity of the document using a score function. Additionally, we also present the experiment results for evaluating the accuracy of the proposed method.

Key Words : Data Mining, Opinion Mining, Classification, Sentiment Analysis

Ⅰ. 서 론

오피니언 마이닝(opinion mining)은 정형화되지 않은 문서에서 주제를 찾아내는 기술인 텍스트 마이닝(text mining)의 한 분야이다. 텍스트 마이닝이 어떠한 문서의 주제를 찾아내고 계층화 한다면, 오피니언 마이닝은 그

*정회원 : 한성대학교 컴퓨터공학과
**정회원 : 한성대학교 컴퓨터공학과
접수일자 2012년 8월 27일, 수정완료 2012년 9월 27일
게재확정일자 2012년 10월 12일

Received: 27 August 2012 / Revised: 27 September 2012 / Accepted: 12 October 2012
*Corresponding Author: jychang@hansung.ac.kr
Dept. of Computer Engineering, Hansung University, Korea

문서를 작성한 사람의 감정(sentiment)을 추출해 내는 기술이다. 즉, 문서의 주제가 무엇인지 보다 그 문서를 작성한 사람이 주제에 대해 어떠한 감정을 가지고 있는가를 판단하여 분석한다. 오피니언 마이닝 분야가 본격적으로 연구되기 전에도 텍스트 마이닝에 필요한 요소로써 감성 분류(sentiment classification)등의 주제로 많은 연구가 진행 되었다[1-5]. 최근에 이르러서는 글쓴이의 감정이 좋고 나쁘고를 넘어 어떠한 부분에 대하여 좋아하는지 아니면 어떠한 부분에 대하여서는 싫어하는 지뿐만 아니라 어느 정도 좋아하는지 까지 분석하는 정도에 이르고 있다.

현재 국제적으로 오피니언 마이닝과 관련한 활발한 연구가 진행되고 있으며[1-5], 국내에서도 한국어를 대상으로 한 다양한 연구 결과들이 발표되고 있다[6-11]. 그러나 국내의 경우 아직 실용적으로 적용할 만큼의 분류 정확도를 보이지 않고 있다. 한국어의 경우 비문법적 표현이 많아 형태소 분석기[12]를 통해 정확한 문법적 구조를 파악하기 쉽지 않을 뿐만 아니라, 다양하고 변화가 많은 감정단어를 갖고 있어 문서의 극성을 판별하기가 쉽지 않기 때문이다.

이러한 배경을 바탕으로 본 논문에서는 한글 오피니언 문서를 자동으로 분류하는 기술을 제안한다. 제안된 분류기술은 단문(short document)으로 구성된 오피니언 문서를 대상으로 한다. 기존의 한국어를 대상으로 하는 방법들은 대부분 문법에 지나치게 의존적이고, 감정단어와 객관적 사실에 대한 단어를 구분하기 위해서 의미사전을 미리 구축해야하는 문제점을 안고 있다. 그러나 네이버 영화평과 같이 40자 이내의 단문으로 구성된 오피니언 문서들은 많은 경우 문법에 어긋나며, 형용사 형태의 감정단어 보다는 한두 개의 명사를 이용하여 자신의 감정을 표현하는 경우도 흔히 찾아볼 수 있다. 그리고 단문의 특성상 객관적인 문장은 거의 찾아볼 수 없다. 따라서 감성단어를 판별하고 이를 이용하여 극성을 판별하는 기존의 분류기법을 직접적으로 적용하기에는 한계가 있다. 따라서 본 논문에서는 문법적 요소를 최대한 배제하고 단어패턴의 빈도만을 고려한 분류기법을 채택하였다. 제안된 방법에서는 감성단어 사전을 미리 구축하지 않고 문서를 단어들의 리스트 추상화하여 패턴들의 빈도로 학습한 후 적절한 스코어 함수를 적용하여 문서의 극성을 판별한다. 또한 부정어 처리를 위한 특별한 조치 없이 unigram부터 n-gram까지 고려한 단어패턴을 탐색한다.

본 논문에서 제안하는 분류 방식은 감정 단어를 추출하는 기존의 오피니언 분류 기법 보다는 일반적인 키워드 빈도 기반의 문서 분류 기법에 더 가깝다고 볼 수 있다. 그 이유는 본 논문이 가정하는 분류 대상이 비교적 단문이기 때문이다. 단문이 아닌 경우의 오피니언 문서에서는 주관적 문장과 객관적 문장이 혼용되어 이들 중에서 주관적 문장을 선별하는 것이 중요한 이슈가 된다. 그러나 한두 문장으로 이루어진 단문의 경우 대부분의 내용이 주관적인 내용이므로 일반적인 문서 분류 기법을 적용하는 것이 오히려 유리한 측면이 있다. 이를 증명하기 위해 본 논문에서는 실험을 실시하여 제안된 기법의 정확도를 평가하였다. 실험은 본 논문이 제안한 방법을 이용하여 긍정과 부정 문서로 분류한 후 분류 정확도를 측정하였으며, 이를 통해 제안된 방법을 우수성을 증명하였다.

II. 관련연구

오피니언 문서를 분류하는 방법으로는 크게 자연어 처리기법과 통계학적 접근법이 있다. [1]에서는 기계 학습(machine learning) 및 자연어 처리 기술을 활용하여, 상품평 데이터에 대한 감성분석 및 분석결과 요약 기법을 제시하고 있으며, 결과물로서 연구목적의 Opinion Observer라는 명칭의 시스템을 개발하였다. 그러나 실제 시스템 개발을 위해 필요한 인프라적 측면을 소홀히 하고 있어 상용화 시스템 개발을 위한 방법론 측면으로서는 미흡한 면이 있다. 미국 카네기멜론 대학교에서는 RedOpal 시스템을 개발한 사례가 있으며[2], 이는 상품평 데이터와 사용자 평가점수를 활용하여 요약 보고서를 생성하는 기법을 제안하였다. 이 연구에서는 상품 속성과 평가 점수에 대하여 다차원 분석 결과를 보여주고 있지만, 주관적 긍정/부정 평가를 수행하지는 않고 있다. [3]에서는 문장 구조와 문장 사이의 관계, 문장성분의 패턴 정보 등의 언어 규칙을 이용한 통계학적 방법으로 오피니언 마이닝에 접근하고 있으며, [4]에서는 워드넷(WordNet)을 활용하여 어휘의 긍정이나 부정적 의미를 판단하고, 이를 센티워드넷(SentiwordNet)으로 응용하여 감정의 폭을 정량화하는 방법을 제시하고 있다. [5]에서는 사용자의 질의어에 대해 가장 관련 있는 문서의 우선순위를 정하는 기법을 제안하였다. 이 방법에서는 주관적 혹은 객관적 문서인가를 고려하지 않고 타 사용자

들의 해당 문서가 얼마나 도움이 되는가를 정량적으로 평가한 수치를 가지고 오피니언 문서의 가치를 계산하는 방법을 이용하였다.

한국어에 관련된 오피니언 마이닝 연구에서는 대부분 한국어 문법구조와 의미 사전을 이용하였다[6, 7, 8, 10, 13, 14]. [10]에서는 한국어 문법구조와 의미 사전을 이용하여 감성분석을 시도하였다. 그러나 이 논문에서는 대체로 문법이 제대로 지켜지지 않고 문장 분리가 쉽지 않은 한글 문서의 특징을 고려하지 않고 있다. [6]에서는 본 논문과 같이 단문을 대상으로 자동 분류를 시도하였으나 긍정 및 부정 패턴 추출 과정이 지나치게 임의적이고 분류 정확도도 만족할 만한 수준이 되지 못하고 있다. [7]에서는 특성(feature)들을 미리 정해놓은 상태에서 이 단어들에 근접한 형용사를 찾아 감정단어를 추출하는 방법을 이용하였으나, 정해진 한글 문법의 패턴에 의존적이어서 이를 지키지 못한 문장에는 적용이 불가능하다. 이외에도 한글 오피니언 문서에 대한 여러 분류 기법을 제안하였으나 대부분 문법에 의존적이고 의미사전을 사전에 구축해야하는 경우를 가정하고 있다. 반면에 본 논문에서는 자유도가 높은 한글의 특성을 고려하여 문법구조를 고려하지 않고 문서를 단순한 단어들의 리스트로 단순화하여 이 문제를 해결하였다.

Ⅲ. 오피니언 문서 분류기법

본 논문에서 제안하는 오피니언 문서 분류방법은 네이버 40자 영화평과 같이 비교적 단문으로 구성된 문서를 대상으로 한다. 따라서 한글의 문법적 구조를 가정하지 않고, '아니다', '않다'와 같이 부정어에 대한 처리도 별도로 가정하지 않는다. 또한 명백한 불용어(stopwords)를 제외하고는 사전에 존재하지 않는 비표준 단어라 할지라도 형태소 분석기를 통해 나타나는 모든 패턴을 분류 모델에 사용한다. 이러한 가정 하에 본 논문에서 제안하는 오피니언 문서 분류를 위한 전체 과정은 그림 1과 같다. 이 그림에서 보는 바와 같이 우선 학습을 위한 오피니언 문서 집합으로부터 이모티콘 등 불필요한 단어들을 제거하고 형태소분석기를 통해 문장을 단어 리스트로 분리해낸다. 일반적으로 형태소 분석기는 문장을 분석하여 단어와 해당 품사를 결과로 출력한다. 본 논문에서는 단어의 품사 중에서 조사와 같이 의미를 부여할 수 없는 불용어만을 제외하고 나머지 모든 단어에 대한 리스트(bag of words)를 유지한다. 또한 신조어나 약자 등 사전에 존재하지 않는 단어들도 리스트에 포함시킨다.

다음 단계로 각 문장의 단어패턴들에 대한 빈도수를 계산하여 뒤에 설명할 분류방법에 의해 분류 모델을 생성한다. 단어패턴은 문법구조를 고려하지 않고 단순히 단어들의 리스트에서 unigram부터 n-gram까지의 패턴을 생성한다. 분류 모델은 현재 문서 분류에 가장 많이 사용하고 있는 베이지안(Bayesian) 분류 기술을 적용하였으며, 이를 이용하여 단어패턴들의 출현 확률을 계산하고 스코어 함수(score function)를 생성한다. 마지막으로 테스트 문서 집합에 대해서 분류 모델을 적용하여 분류 정확도를 평가한다.

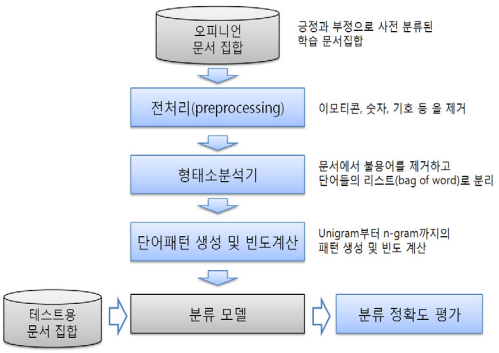


그림 1. 오피니언 문서 분류 과정
Fig 1. Process of Opinion Document Classification

그림 1의 과정들을 위해 오피니언 문서의 집합을 다음과 같이 정의한다. 우선 학습을 위해 수집된 전체 문서의 집합을 D라 하고 이 중에서 긍정 문서집합을 D_P, 부정 문서집합을 D_N이라 하자. 그러면 $D = D_P \cup D_N$이 성립한다. 또한 각 문서로부터 형태소 분석기를 통해 단어들을 추출하는데, 하나의 문서로 부터 추출된 단어를 이용하여 unigram부터 n-gram까지 단어패턴들의 집합을 생성할 수 있다. 하나의 문서 d로부터 생성된 단어패턴 집합을 $\overline{d}$로 정의한다. 예를 들어 d가 "내 생애 최고의 영화"인 경우 4-gram까지 단어패턴을 생성한다면 $\overline{d}$는 다음과 10개의 단어패턴으로 구성된다.

unigram : {(내),(생애),(최고), (영화)}
bigram : {(내, 생애), (생애 최고), (최고, 영화)

trigram : {(내, 생애, 최고), (생애, 최고 영화)}
4-gram : {(내, 생애, 최고, 영화)}

이와 같은 과정을 통해 D_P와 D_N은 각각 다음과 같이 단어패턴들의 집합인 $\overline{D_P}$와 $\overline{D_N}$으로 재정의할 수 있다.

$$\overline{D_P} = \bigcup_{d \in D_P} \bar{d} \quad (1)$$

$$\overline{D_N} = \bigcup_{d \in D_N} \bar{d} \quad (2)$$

다음으로 각 단어패턴 w에 대해서, 이 패턴이 $\overline{D_P}$와 $\overline{D_N}$에 각각 출현한 빈도수를 계산할 필요가 있다. 이를 위해 다음과 같이 $f_P(w)$와 $f_N(w)$를 정의한다.

$f_P(w)$ = $\overline{D_P}$에서 단어패턴 w의 출현 빈도 (3)

$f_N(w)$ = $\overline{D_N}$에서 단어패턴 w의 출현 빈도 (4)

이와 같은 정의에 따라 단어패턴 w는 $\overline{D_P}$와 $\overline{D_N}$에 각각 출현한 빈도수에 따라 그림 2와 같이 분류할 수 있다.

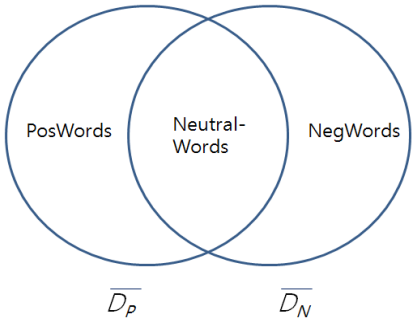


그림 2. 단어패턴의 분류
Fig 2. Classification of Word Patterns

이 그림에서 각 집합은 다음과 같이 정의된다.

$$Pos\ Words = \{w \mid f_P(w) > 0 \text{ and } f_N(w) = 0\} \quad (5)$$

$$Neg\ Words = \{w \mid f_P(w) = 0 \text{ and } f_N(w) > 0\}$$

$$Neutral\ Words = \{w \mid f_P(w) > 0 \text{ and } f_N(w) > 0\}$$

이 식에서 *PosWords*는 $\overline{D_P}$에만 나타나며, $\overline{D_N}$에는 나타나지 않는 단어패턴들의 집합이며, *NegWords*는 반대로 $\overline{D_N}$에만 나타나며, $\overline{D_P}$에는 나타나지 않는 단어패턴들의 집합이다. 따라서 이들은 순수하게 긍정패턴과 부정패턴으로 분류할 수 있다. 반면에 *NeutralWords*는 양쪽 모두에 최소한 한번이상 출현하는 패턴들의 집합이다. 기존 연구에서는 *NeutralWords*에 속하는 각 단어패턴 w에 대해서 $f_P(w)$와 $f_N(w)$중 어느 빈도가 어느 정도 높은가에 따라 일방적으로 하나의 극성으로 판단하였다[8]. 그러나 이러한 강제적인 극성 판단은 오히려 각 패턴이 갖고 있는 의미가 왜곡될 수 있으며, 의미 사전을 미리 구축해야하는 등 많은 제약이 존재한다.

본 논문에서는 각 단어패턴에 대해 강제적인 분류보다는 어느 방향에 가까운지에 대한 정량적인 수치만을 부여한다. 따라서 추후에 극성을 판단해야하는 문장 중에 해당 단어패턴이 나타날 경우, 각 단어패턴에 대한 이 수치를 스코어 함수에 적용한다. 정량적 수치를 부여하기 위해 가장 보편적인 방법은 베이지안 확률(Bayesian probability)을 이용하는 것이다. 즉, 단어패턴 w가 긍정 및 부정 단어패턴일 확률은 각각 다음과 같이 정의할 수 있다.

$$p(\overline{D_P}|w) = \frac{p(w|\overline{D_P})\ p(\overline{D_P})}{p(w)} \quad (6)$$

$$p(\overline{D_N}|w) = \frac{p(w|\overline{D_N})\ p(\overline{D_N})}{p(w)} \quad (7)$$

이 식에서 $p(\overline{D_P}|w)$는 w가 긍정 패턴일 사후확률(posterior probability)을 나타내며, $p(\overline{D_P}|w)$는 w가 부정 패턴일 사후확률을 나타낸다. $p(w|\overline{D_P})$와 $p(w|\overline{D_N})$은 각각 $\overline{D_P}$와 $\overline{D_N}$에서 w가 출현할 확률이다. 또한 $p(\overline{D_P})$와 $p(\overline{D_N})$은 각각 전체패턴 중 $\overline{D_P}$와 $\overline{D_N}$의 비중을 나타내는 사전확률(prior probability)이다. 마지막으로 $p(w)$는 전체 단어패턴에서 w가 차지하는 비율을 나타낸다.

이와 같은 분류 방법에 따라 마지막으로 필요한 것은 주어진 문장이 긍정적인 문장인지 아니면 부정적인 문장인지 판별하는 것이다. 기존의 일부 연구에서는 긍정과

부정으로 분류된 각 의미사전의 단어에 대해 1과 -1과 같은 일방적인 방향만을 지정하기도 하고, 정해진 수치로 가중치를 부여하는 등 임의의 방식으로 단어에 대해 점수를 부여하고 있다[6]. 본 논문에서는 이를 해결하기 위해 식 (6)와 (7)에서 계산된 확률 값과 단어패턴들의 출현 빈도를 이용하여 단어의 극성을 계산한다. 즉, 주어진 문서 d에 대해 극성을 판단하기 위한 스코어 함수는 다음과 같이 계산할 수 있다.

$$Pscore(d) = \sum_{w \in \vec{d}} (f_P(w) \times p(\overline{D_P}|w)) \quad (8)$$

$$Nscore(d) = \sum_{w \in \vec{d}} (f_N(w) \times p(\overline{D_N}|w)) \quad (9)$$

이 식에서 $Pscore(d)$는 문서 d가 긍정 문서일 가능성에 대한 점수이며, 반대로 $Nscore(d)$는 반대일 경우에 대한 점수이다. 따라서 이 두 값을 비교하여 더 높은 값으로 d의 극성을 판단할 수 있다.

식 (8)과 (9)에서 제시한 스코어 함수는 출현하는 모든 단어패턴에 대해서 일률적으로 계산한 방법으로 보다 정확한 분류를 위해서는 여러 가지 변수에 대한 고려가 필요하다. 우선 단어패턴 w가 긍정과 부정 패턴에 모두 나타나는 경우, 즉 w가 *NeutralWords*에 나타나는 경우의 처리방법을 결정해야한다. 이 경우에 가장 좋은 방법은 w가 긍정 혹은 부정 패턴에 속할 확률의 차이에 의해 판단하는 것인데, 확률의 차이는 다음의 식에 의해 계산할 수 있다.

$$\alpha = |p(\overline{D_P}|w) - p(\overline{D_P}|w)| \quad (10)$$

여기서 계산된 α값에 대해서 이 값 이상인 경우만을 대상으로 스코어 함수를 적용한다. α가 0인 경우는 형태소 분석기를 통해 출현하는 모든 단어패턴들에 대해 적용하는 것이고, 1.0인 경우는 *NeutralWords*에 나타나는 모든 단어패턴을 스코어 함수에서 배제하는 것이다. 각각의 경우에 대해서 장단점이 있는데 α가 0에 가까울 경우에는 긍정과 부정에 고르게 나타나는 패턴, 즉 중립 단어들에 대해서도 스코어 함수를 적용해야하는 문제점이 있는 반면, 1에 가까울수록 적용가능한 단어패턴의 수가 줄어들어 극성을 판단하기 불가능한 경우가 발생할 가능성이 높게 된다. 본 논문의 4장에서는 이 값의 변화에 따른 분류 정확도에 대한 실험 결과가 제시되어 있다.

지금까지 제시한 문서 분류 방법은 감정 단어를 추출하는 것이 핵심인 기존의 오피니언 분류 기법보다 오히려 키워드 빈도에 기반한 일반적인 문서 분류 기법에 더 가깝다고 볼 수 있다. 서론에서 언급한 바와 같이 단문이 아닌 경우에는 문장들에서 주관적 표현을 분리하는 것이 중요한 문제가 된다. 하지만 단문의 경우에는 표현자체가 대부분 주관적인 문장이므로 일반적인 문서 분류 방식을 적용하는 것이 오히려 분류 정확도를 높일 수 있다.

구체적으로 본 논문에서 제안한 오피니언 문서 분류 방법의 특징은 다음과 같이 요약할 수 있다. 우선 기존 연구[6-10]와는 달리 한국어의 문법적 구조를 고려하지 않으므로 맞춤법 오류, 띄어쓰기 오류, 비속어 사용 등과 같은 문제에도 유연하게 대처할 수 있다. 다만 이러한 단어들의 추출은 형태소 분석기의 성능에 절대적으로 의존한다. 일부 형태소 분석기의 경우에는 사전에 없는 단어의 경우에도 분석 결과에 포함시키는 경우가 있다. 이 경우 올바른 형태소로의 분리가 되지는 않지만 이러한 단어들의 출현 빈도가 높을 경우 분류에 어느 정도 기여할 수 있다. 또한 본 논문에서 제시한 방법은 unigram부터 n-gram까지 고려하여 '안', '않다'와 같은 부정어 대한 별도의 처리 방법이 필요 없다. 기존의 대부분의 감성 분류 기법은 부정어를 별도로 처리한다. 그러나 한국어의 특성상 다양한 부정어를 완벽히 처리할 방법은 사실상 불가능하다. 반면에 본 논문에서는 부정어를 n-gram까지의 단어패턴에 자연스럽게 포함되도록 처리하였다. 마지막으로 본 논문에서 제시한 기법은 감성단어와 같은 의미사전을 사전에 구축할 필요가 없다. 한국어의 경우 감성사전을 완벽히 구축하는 것은 불가능하다. 단문으로 구성된 문서의 경우 단순한 방법으로 감정 표현이 가능하기 때문이다. 예를 들어 "억지 감동", "쓰레기 영화", "적극 추천"과 같이 완벽한 문장이 아닌 축약된 방법으로 표현하는 경우가 많으나 이런 모든 경우를 감성사전으로 구축하는 것은 매우 어려운 일이기 때문이다.

Ⅳ. 실험 및 결과

본 논문에서 제안한 오피니언 문서 분류기법의 정확도를 평가하기 위해서 실험을 실시하였다. 실험은 네이버에서 제공되는 40자 영화평을 대상으로 하였다. 네이

표 1. 빈도수 상위 10개의 단어패턴들
Table 1. Word Patterns of Top 10

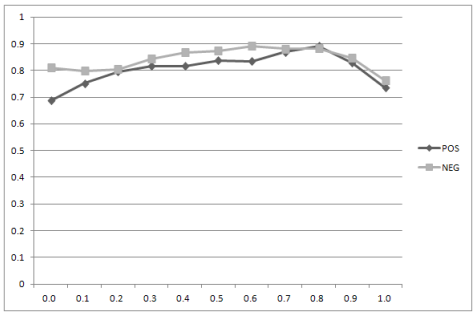
순위	unigram		bigram		trigram	
	긍정	부정	긍정	부정	긍정	부정
1	영화	영화	최고 영화	이 영화	말 필요 없	이 주 기
2	보	보	이 영화	최악 영화	보 하 영화	아까 울 영화
3	최고	이	좋 영하	쓰레기 영화	꼭 보 하	돈 아까 워
4	정말	없	영화 보	이 것	인생 최고 영화	하 말 없
5	하	하	수 있	이런 영화	보 영화 중	이 것 아니
6	이	것	하 영화	영화 보	이 영화 보	보 영화 중
7	감동	나	꼭 보	돈 아까	내 인생 최고	영화 중 최악
8	있	정말	것 같	아까 울	정말 좋 영화	이 걸 영화
9	좋	진짜	보 영화	재미 없	캡틴 마이 캡틴	돈 아까 울
10	것	1	이런 영화	아까 워	수 있 영화	쓰레기 같 영화

버 영화평은 평점을 1점부터 10점까지 부여하게 되어있으며, 이 중에서 학습 데이터를 위해 평점 10인 긍정 영화평과 평점 1인 부정 영화평을 각각 2만개씩 수집하여 그림 1에서 설명한 과정을 통해 처리하였다. 그리고 테스트를 위해 별도로 긍정/부정 영화평을 각각 1,000개씩 별도로 수집하여 정확도를 측정하였다. 다음으로 단어패턴의 길이를 제한하기 위해 각 문서에 대해서 unigram, bigram, trigram까지 생성하였다. 이렇게 생성된 총 단어패턴의 개수는 약 30만개이다. 표 1은 이렇게 생성된 단어패턴 중에서 최대 빈도를 나타내는 상위 10개의 단어패턴을 보여준다. 이 표에서 보듯이 긍정 및 부정패턴에서 최대 빈도를 나타내는 단어는 공통적으로 '영화'이다.

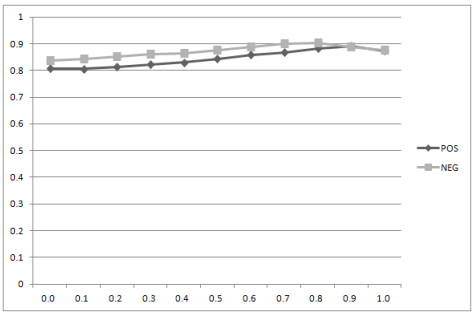
문서 분류의 정확도를 측정하는 방법은 다양하나 가장 보편적으로 사용하는 방법이 정확도(precision)와 재현율(recall) 그리고 이들의 평균인 F-value를 측정하는 것이다. 우선 첫 번째 실험으로 고정된 단어패턴 길이에 따라 문서 분류의 성능을 측정하였다. 본 실험에서는 단어패턴을 unigram, bigram, trigram으로만 분리하였으므로, 각각의 패턴에 대해서 실험을 실시하였으며, 이들 천제를 대상으로도 실시하였다. 또한 식 (10)에서와 같이 α 값을 0.0부터 1.0까지 변화하며 성능 변화를 관찰하였다.

그림 3은 정확도를 측정한 실험 결과이다. 이 그림에서 (a)(b)(c)는 각각 unigram, bigram, trigram을 별도로 분리하여 실험한 결과를 나타내며, (d)ALL은 전체를 대상으로 한 실험 결과이다. x축은 α의 변화를 나타내며, POS와 NEG는 각각 긍정과 부정 영화평에 대한 실험 결과이다. 이 그림에서 보는 바와 같이 trigram을 제외하고는 α가 증가함에 따라 분류정확도가 증가하지만 일정수준 이상에서는 오히려 떨어지는 것을 관찰할 수 있다. trigram의 경우는 α값에 큰 영향을 받지 않는 것으로 나타났다. 그 이유는 단어 패턴의 길이가 길수록 긍정과 부정에 대한 표현이 보다 명확히 나타나기 때문이다. ALL의 경우를 보면 unigram의 성능과 유사한 것을 알 수 있다. 그 이유는 전체적인 성능이 스코어 함수에 영향을 미치는 단어패턴의 빈도에 기인하는데 unigram에서의 단어패턴 빈도가 다른 것에 비해 현저하게 크므로 ALL의 성능이 unigram에서의 성능에 전적으로 의존하기 때문이다. 다만 ALL의 경우 0.8이상에서는 unigram과 비교하여 좋은 성능을 나타내는데, unigram에서는 정확한 분류가 되지 않았으나 bigram이나 trigram에서의 분류 정확도에 어느 정도 영향을 받은 것으로 추정된다.

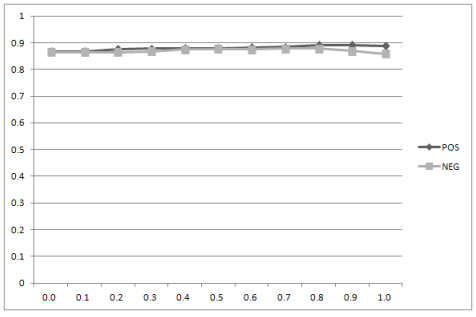
그림 4는 재현율을 측정한 실험 결과이다. 전반적으로 α가 증가함에 따라 재현율이 낮아지는 것을 알 수 있다. 그 이유는 α가 증가하게 되면 스코어 함수에 적용되는 단어패턴 집합이 점차 작아지므로 패턴이 발견되지 않아 분류할 수 없는 영화평들의 수가 점차 증가하기 때문이다. 또한 ALL의 경우가 전반적으로 가장 좋은 재현율을 보이고 있다. trigram의 경우는 정확도 측면에서는 좋은 성능 보였으나, 패턴을 찾지 못하는 경우가 많이 발생하여 α에 관계없이 재현율이 매우 낮은 것으로 측정되었다. 따라서 trigram만으로는 오피니언 문서 분류에 단독으로 사용하는 것은 어렵다고 하겠다. 그림 5는 그림 3과 그림 4의 결과를 하나로 나타낸 F-value에 대한 실험 결과이다. 이 그림에서 확인할 수 있듯이 ALL이 α에 관계없이



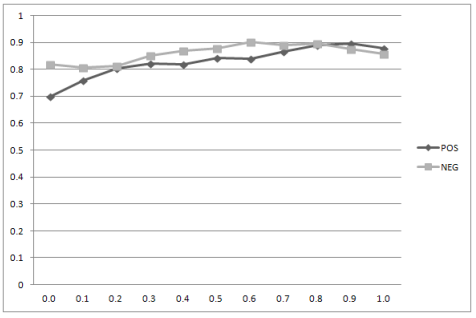
(a) unigram



(b) bigram

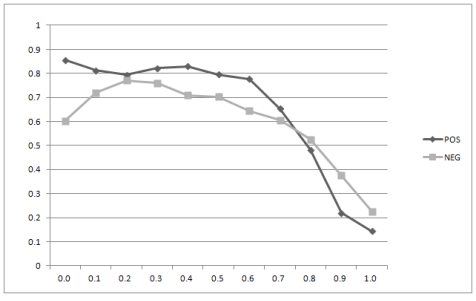


(c) trigram

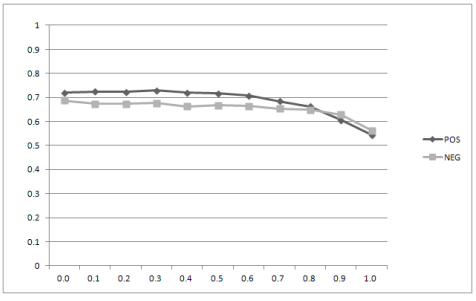


(d) ALL

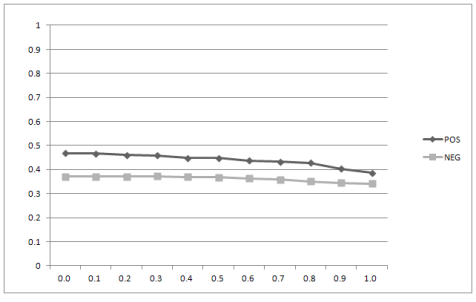
그림 3. 정확도 측정 결과
Fig 3. Experiment Result of Accuracy



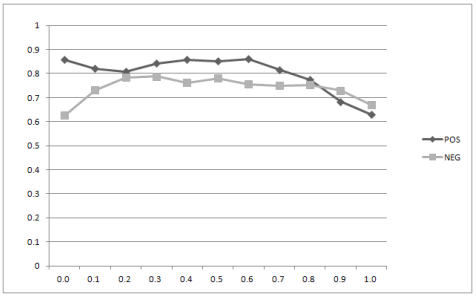
(a) unigram



(b) bigram



(c) trigram



(d) ALL

그림 4. 재현율 측정 결과
Fig 4. Experiment Result of Recall

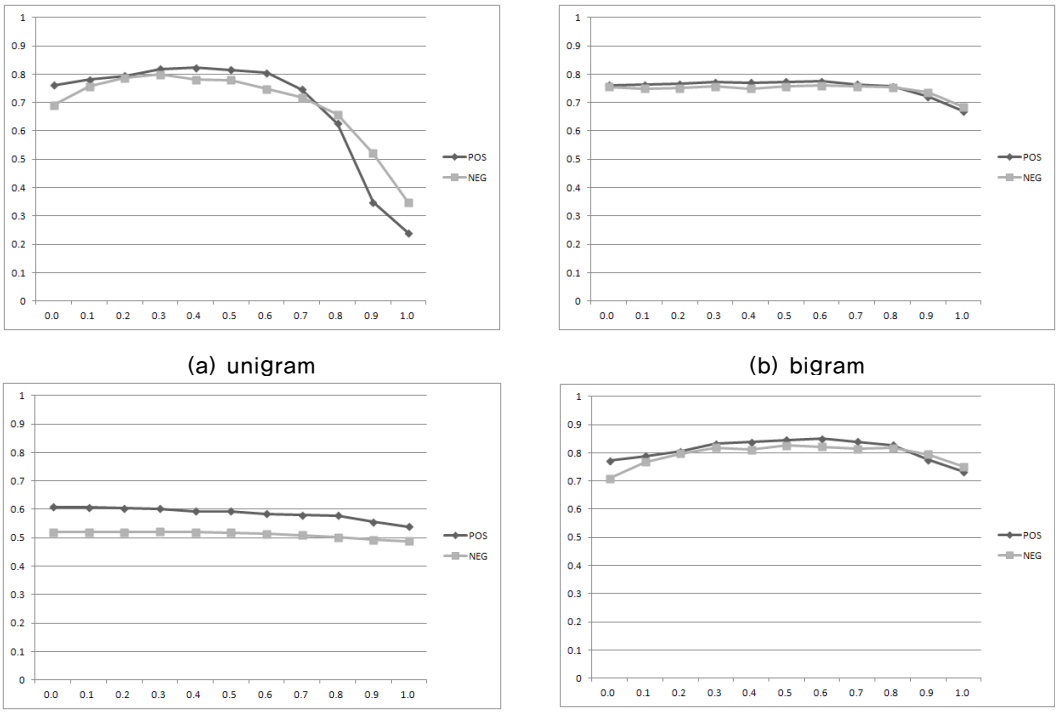


그림 5. F-value 측정 결과
Fig 5. Experiment Result of F-Value

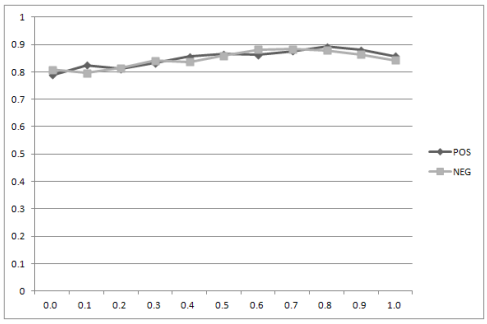
전체적으로 가정 좋은 성능을 보인다는 것을 알 수 있다. 특히 α값이 0.2~0.8에서는 0.8이상의 좋은 성능을 보이고 있다. 따라서 오피니언 문서 분류에 있어서 적당한 수준 에서의 α값을 사용하고, 고정된 길이의 단어패턴 보다는 다양한 길이의 패턴을 사용함으로써 전체적인 분류 성능을 높일 수 있다는 것을 확인할 수 있다.

그림 3, 4, 5의 실험결과를 보면 전반적으로 모든 단어패턴을 사용한 ALL의 경우가 가장 좋은 성능을 보인다는 것을 확인할 수 있었다. 그러나 이 방법의 문제점은 ALL의 정확도가 unigram의 성능에 크게 좌우된다는 것이다. 그 이유는 표 2와 같이 unigram의 단어패턴 빈도가 다른 것에 비해 매우 크기 때문이다. 이를 해결하기 위해 단어패턴의 길이에 따라 빈도수를 정규화(normalization)하여 그 편차를 보정하여 실험을 실시하였다. 이를 적용한 스코어 함수는 다음과 같다.

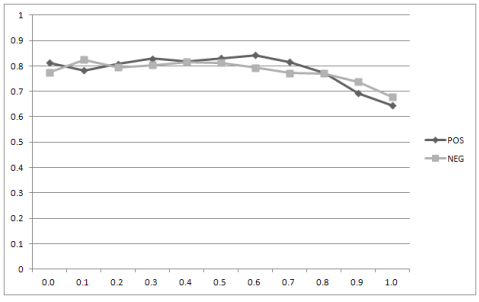
$$Pscore(d) = \sum_{w \in \vec{d}} \left(\frac{f_P(w)}{\max_{|w_i| = |w|}(f_P(w_i))} \times p(\overline{D_P}|w) \right) \quad (11)$$

$$Nscore(d) = \sum_{w \in \vec{d}} \left(\frac{f_N(w)}{\max_{|w_i| = |w|}(f_N(w_i))} \times p(\overline{D_N}|w) \right) \quad (12)$$

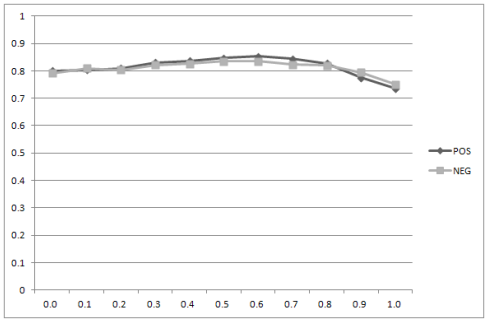
여기서 $|w|$는 단어패턴의 길이를 나타내는데. $|w|$가 1, 2, 3이면 각각 unigram, bigram, trigram을 나타낸다. 따라서 $\max_{|w_i| = |w|}(f_P(w_i))$와 $\max_{|w_i| = |w|}(f_N(w_i))$는 w와 단어패턴의 길이가 같은 것 중에 각각 긍정과 부정의 최대빈도수를 의미한다. 이 스코어 함수를 이용하여 동일한 실험을 실시한 결과는 그림 6과 같다. 이 그림에서 F-value를 보면 정규화 이전(그림 5의 F-value)의 경우와 비교하여 유의한 차이는 보이지 않고 있다. 그러나 (a)정확도와 (b)재현율을 정규화하기 이전 성능과 비교해봤을 때는 전반적으로 긍정(POS)와 부정(NEG)간의 차이가 거의 나타나지 않고 일관성있는 성능을 보이고 있다. 그 이유는 정규화 이전의 경우는 출현빈도가 높은 unigram의 일부 단어에 의해 성능이 크게 영향을 받는 경향이 있었으나 정규화를 함으로써 단어패턴의 길이에 따른 빈도의 편차에 성능이 왜곡되는 경향이 사라진 결과라고 볼 수 있다.



(a) 정확도



(b) 재현율



(c) F-value

그림 6. 정규화에 따른 실험 결과
Fig 6. Experiment Result after Normalization

지금까지의 실험에서는 스코어 함수를 적용할 때 단어패턴의 길이에 따라 독립적 혹은 하나의 동일한 집합으로 처리하였다. 그러나 이전 실험의 결과를 분석해보면 trigram의 경우 정확도면에서 가장 우수한 성능을 보이고, 재현율에서는 unigram이 비교적 좋은 성능 보이고 있다. 따라서 단어패턴의 길이가 높을수록 스코어 함수에서 가중치를 부여하게 되면 높은 정확도와 동시에 높은 재현율도 기대할 수 있다. 단어패턴의 길이에 따라 가중치를 부여한 스코어 함수는 다음과 같다.

$$Pscore(d) = \sum_{w \in \bar{d}} \left(\frac{f_P(w)}{\max_{|w_j|=|w|}(f_P(w_i))} \times p(\overline{D_P}|w) \right) \times weight(w) \quad (13)$$

$$Nscore(d) = \sum_{w \in \bar{d}} \left(\frac{f_N(w)}{\max_{|w_j|=|w|}(f_N(w_i))} \times p(\overline{D_N}|w) \right) \times weight(w) \quad (14)$$

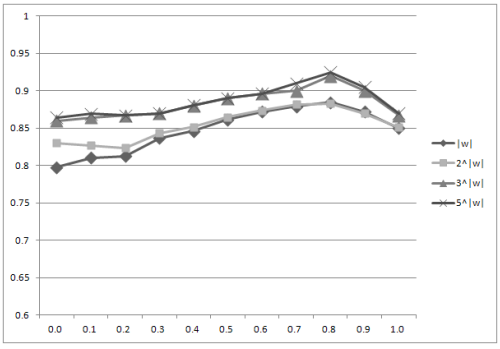
이 식에서 $weight(w)$는 단어패턴 길이에 따른 가중치를 의미한다. 이 수식에 따라 그림 7은 단어패턴의 길이에 따라 가중치를 부여한 실험에서의 성능을 보여준다. 실험은 $weight(w)$가 각각 $|w|$, $2^{|w|}$, $3^{|w|}$, $5^{|w|}$로 점차 가중치의 값을 높여가며 그 변화를 측정하였으며, 긍정과 부정 영화평에 대한 분류성능을 별도로 나누지 않고 통합하여 실시하였다. 이 그림에서 보는 바와 같이 정확도에서는 α값이 0.8일 경우 가장 좋은 성능을 보였으며, 재현율에서는 0.6에서 가장 좋은 성능 보였고, F-value에서도 0.6에서 가장 좋은 결과를 나타내고 있다. 또한 정확도, 재현율 모두에서 가중치가 커짐에 따라 좋은 성능을 나타내고 있다. 다만 가중치가 $3^{|w|}$과 $5^{|w|}$ 사이에서는 큰 변화가 없는 것으로 보아 그 이상의 가중치를 부여해도 더 이상의 성능 향상을 기대하기 어렵다는 것을 예측할 수 있다.

지금까지의 실험들을 종합해보면 고정된 단어패턴의 길이를 보다 다양한 형태의 패턴을 혼용하는 것이 좋은 성능을 보였다. 또한 단어패턴의 길이에 따라 가중치를 부여하는 것이 그렇지 않은 경우보다 좋은 성능을 기대할 수 있다. 본 논문에서 제안한 오피니언 분류 기법은 기존의 연구에서 실시했던 실험결과[6-10]와 비교했을 때 더 좋거나 유사한 성능을 보여주고 있다. 따라서 단문으로 구성된 오피니언 문서 분류에 있어서는 감성분석 분류기술을 굳이 사용하지 않더라고 기존의 문서분류 방식만으로도 만족할 만한 성능을 기대할 수 있다.

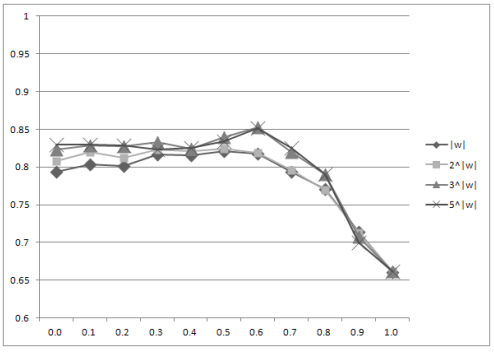
V. 결 론

본 논문에서는 단문으로 구성된 오피니언 문서에 대한 주관적 의견을 자동으로 분류할 수 있는 방법을 제안하였다. 제안된 방법은 한국어 문법을 고려하지 않고 부정어에 대한 특별한 처리를 고려하지 않는 등 기존의 방법과 다른 기준을 적용하여 처리하였다. 실험결과 F-value만을 보았을 때는 최고 86%이상의 높은 정확도

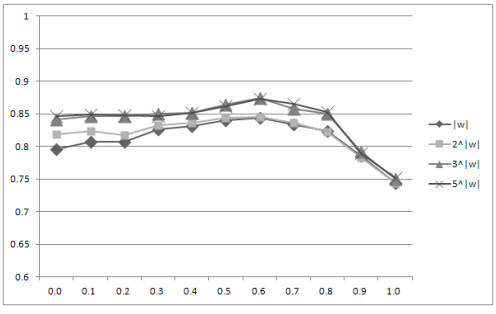
를 보였다. 직접적인 비교는 무리가 있지만 이 실험결과는 기존의 연구와 비교하여 유사하거나 더 좋은 분류 정확도를 가진다고 말할 수 있다. 따라서 단문의 오피니언 문서에서는 기존의 문서분류 방식만으로도 만족할 만한 성능을 기대할 수 있다. 향후에는 본 논문에서 적용한 기술에 기존의 감성분석 기술을 융합하여 분류 정확도를 더욱 향상시킬 수 있는 방안을 모색할 계획이다.



(a) 정확도



(b) 재현율



(c) F-value

그림 7. 가중치 변화에 따른 실험 결과
Fig 7. Experiment Results with Various Weights

참 고 문 헌

[1] B. Liu , M. Hu , and J. Cheng, “Opinion observer: analyzing and comparing opinions on the Web”, Proceedings of the 14th international conference on WWW, pp. 10-14, 2005.

[2] C. Scaffidi, K. Bierhoff, E. Chang, M. Felker, H. Ng, and C. Jin, “Red Opal: Product-Feature Scoring from Reviews”, Proceedings of the 8th ACM conference on Electronic commerce, pp. 11-15, 2007.

[3] Xiaowen Ding, and Bing Lui, “The Utility of Linguistic Rules in Opinion Mining”, SIGIR 2007, pp. 811-812, 2007.

[4] E. Courses, and T. Surveys, “Using SentiWordNet for multilingual sentiment analysis”, Data Engineering Workshop ICDEW 2008, 2008.

[5] Q. Miao, Q. Li, and R. Dai, “A sentiment mining and retrieval system”, Expert Systems with Applications, Vol.36, pp. 7192-7198, 2009.

[6] J. O. Kim, S. S. Lee, W, S, Yong, “Automatic Opinion Classification Of Korean Text”, Journal of KIISE: Database, Vol. 38, No. 6, Dec., 2011.

[7] J. S. Myoung, D. J. Lee, S. G. Lee, “A Korean Product Review Analysis System Using a Semi-Automatically Constructed Semantic Dictionary”, Journal of KIISE, Vol. 35, No. 6, 2008.

[8] H. H. Kang, S. J. Yoo, S. I, Han, “Automatic Extraction of Opinion Words from Korean Product Reviews Using the k-Structure”, Journal of KIISE, Vol. 37, No. 6, 2010.

[9] J. Y. Chang, “A Sentiment Analysis Algorithm for Automatic Product Reviews Classification in On-Line Shopping Mall”, Journal of Korea Society for E-Business Studies, Vol. 14, No. 4, 2009.

[11] J. Y. Chang, J. M. Kim, S, Y, Lee, “Automatic Classification of Korean Movie Reviews Using a Word Pattern Frequency”, Proc. of 2012 Korea Computer Congress, 2012.

[12] S. S. Kang, Korean Morpheme Analysis and Information Retrieval, HongRung Publishing Company, 2003.

[13] C. Park, D. Seong, K. Lee, “Automatic IPC Classification for Patent Documents using Machine Learning”, Journal of Korean Institute of Information Technology, Vol. 10, No. 4, 2011.

[14] J. Shim, H. C. Lee, “The Development of Automatic Ontology Generation System Using Extended Search Keywords” Journal of the Korea Academia-Industrial cooperation Society, Vol. 11, no. 6, 2009.

※ 본 논문은 한성대학교 교내연구비 지원과제임

저자 소개

장 재 영(정회원)

- 1992년 : 서울대학교 계산통계학과 (이학사)
- 1994년 : 서울대학교 계산통계학과 (이학석사)
- 1999년 : 서울대학교 계산통계학과 (이학박사)
- 2000년 ~ 현재 : 한성대학교 컴퓨터 공학과 교수

<관심분야 : 데이터베이스, 정보검색, 데이터마이닝>

김 일 민(정회원)

- 1984년 : 경북대학교 전자공학과
- 1995년 : 아리조나주립대학 전산학박사
- 1985년 ~ 1987년 : 한국전자통신연구원(ETRI) 연구원
- 1996년 ~ 1987년 : 삼성SDS 멀티미디어교육센터 책임
- 1997년 3월 ~ 현재 : 한성대학교 컴퓨터공학과 교수

<관심분야 : 운영체제, 분산처리>