한글문서분류에 SVD를 이용한 BPNN 알고리즘†

(BPNN Algorithm with SVD Technique for Korean Document Categorization)

리 청 화*, 변 동 률*, 박 순 철*
(Chenghua Li, DongRyul Byun and Soon Choel Park)

요 약 본 논문에서는 역전파 신경망 알고리즘(BPNN: Back Propagation Neural Network)과 Singular Value Decomposition(SVD)를 이용하는 한글 문서 분류 시스템을 제안한다. BPNN은 학습을 통하여 만들어진 네트워크를 이용하여 문서분류를 수행한다. 이 방법의 어려움은 분류기에 입력되는 특징 공간이 너무 크다는 것이다. SVD를 이용하면 고차원의 벡터를 저차원으로 줄일 수 있고, 또한 의미있는 벡터 공간을 만들어 단어 사이의 중요한 관계성을 구축할 수 있다. 본 논문에서 제안한 BPNN의 성능 평가를 위하여 한국일보-20000/한국일보-40075 문서범주화 실험문서집합의 데이터 셋을 이용하였다. 실험결과를 통하여 BPNN과 SVD를 사용한 시스템이 한글 문서 분류에 탁월한 성능을 가지는 것을 보여준다.

핵심주제어 : 문서 분류, 역전파 신경망 알고리즘, SVD

Abstract This paper proposes a Korean document categorization algorithm using Back Propagation Neural Network(BPNN) with Singular Value Decomposition(SVD). BPNN makes a network through its learning process and classifies documents using the network. The main difficulty in the application of BPNN to document categorization is high dimensionality of the feature space of the input documents. SVD projects the original high dimensional vector into low dimensional vector, makes the important associative relationship between terms and constructs the semantic vector space. The categorization algorithm is tested and compared on HKIB-20000/HKIB-40075 Korean Text Categorization Test Collections. Experimental results show that BPNN algorithm with SVD achieves high effectiveness for Korean document categorization.

Key Words : Documents categorization, Back Propagation Neural Network Algorithm, SVD

1. 서 론

문서 분류는 문서를 미리 정의된 하나 이상의 범주로 분리하는 것이다[1]. 지식의 양이 많아지면서 인류는 정보의 체계화의 필요성을 느끼게 되었고, 그 방법 중 하나로 분류를 사용하고 있다. 아날로그 문서로 생성되던 지식들이 컴퓨터와 인터넷의 발달로 디지털 자료들로 생성형태가 바뀌면서 생산되는 문서의 양은 기하급수적으로 증가하게 되었다. 이와 더불어 수작업으로 하던 분류 방법은 너무 많은 시간을 소비하게 되었다. 따라서 수작업을 대체할 수 있는 효율적이며

† 이 논문은 2009년 대학산업기술지원단 안식·연구년 교수 심층 기업지원 사업의 지원(06A0906260050)에 의해 연구되었음.
* 전북대학교 전자정보공학부

빠른 자동 분류방법의 필요성이 대두되었다. 이에 따라 정해진 분류체계에 따라 분류하고자 하는 각 문서들을 가장 적합한 범주에 배정하는 자동 분류에 대한 연구가 활발하게 진행되고 있다. 문서를 자동으로 분류하기 위해서는 대상이 되는 문서의 특성을 파악하여 존재하는 범주와 적합도를 계산하고 가장 최상의 값을 가지는 범주로 문서를 분류하게 된다.

기존의 문서 분류에 사용되는 알고리즘은 단순히 단어들 관계를 가지고 있는 문서 벡터 모델을 이용 문서와 범주와의 유사성을 계산하여 문서를 분류하게 된다. 이것은 단어들 간의 관계, 단어들과 문서의 관계, 문서와 범주와의 관계를 제대로 표현할 수 없다.

본 논문에서 한글문서 분류에 제안하는 다층 구조의 신경망을 가지는 BPNN은 학습을 통하여 분류 시스템의 오류를 분석한다. 오류를 분석한 결과를 통하여 전위층의 가중치를 고친다. 이러한 학습 과정을 거쳐 최적의 분류기를 완성하게 된다. 또한 문서를 분류하기 위하여 문서의 특징을 가진 고차원의 벡터를 사용한다. 고차원의 벡터는 뛰어난 성능에 비하여 분류를 위한 학습시간과 분류시간이 크다는 단점을 가진다[2, 3, 4, 5]. SVD를 이용하면 고차원의 벡터를 문서의 의미정보를 가지는 저차원의 벡터로 분해할 수 있다.[6, 7]. 저차원의 벡터를 이용하여 역전파 신경망 알고리즘(BPNN)을 이용하여 분류 모델을 학습하고, 문서를 분류하게 된다.

본 논문은 2장에서 문서 분류 방법에 대한 관련 연구들을 살펴보고 3장에서는 BPNN에 특징과 문서분류에서 많이 사용되는 K-Nearest Neighbor(KNN) 알고리즘에 대하여 설명한다. 4장에서 시스템 전체구조에 대하여 설명하고 문서분류에 사용되는 특징을 추출하는 방법과 SVD를 이용한 문서 벡터 분해 방법을 설명한다. 5장과 6장에서는 실험 방법, 실험결과 및 결론에 대하여 기술한다.

2. 관련 연구

자동으로 문서를 분류하기 위하여 어느 범주로 문서를 분류할 것인지를 결정하기 위하여 문서 분류 규칙을 정한다. 문서 분류를 위한 학습에 사용되는 알고리즘은 규칙 기반 방법, 확률 기반 방법, 결정트리를 이용하는 방법, SVM을 이용한 방법 등이 있다.

2.1 규칙 기반 방법

규칙 기반 방법은 학습을 통하여 문서 분류 모델을 만드는 것이 아니라 사용자에 의해 규칙을 미리 정의하고 그 규칙에 따라 문서를 분류하는 것이다. 일종의 전문가 시스템을 구축하기 위한 방법이라 볼 수 있다. 규칙 기반 방법을 이용한 시스템으로는 Carnegie Group에서 개발한 CONSTRUE 시스템이 있다. 이 시스템은 로이터셋에 대하여 실험한 결과 정확률과 재현율이 90% 이상 되는 훌륭한 결과를 보였다. 그러나 규칙 기반 방법은 수작업으로 구축한 규칙이므로 다른 분야에 사용하거나 시스템을 확장할 때에 많은 시간과 비용이 요구된다[10].

2.2 확률 기반 방법

확률 기반 방법은 학습 문서 데이터 셋에 나타나는 단어들이 특정 범주의 문서에 나타날 확률값을 계산하고 그것을 통하여 새로운 문서가 속하는 범주를 예측하는 방법이다. 베이스 정리(Bayes' theory)를 이용하는 단순 베이지언 분류기(Naive Bayesian classifier)가 대표적인 확률 기반 방법이다[11]. 문서에 속해 있는 용어들과 범주와의 결합 확률값(joint probability)를 사용한다. n개의 용어로 구성된 문서를 $(t_1, t_2, t_3, \ldots, t_{n-1}, t_n)$의 벡터 형식으로 표현한 다음 $P(c_i | t_1, t_2, t_3, \ldots, t_{n-1}, t_n), c_i \in C$을 계산하여 가장 확률이 높은 범주에 문서를 할당하는 방식이다. 좋은 성능을 가질 수 있으나 많은 시간의 학습을 가져야 한다는 문제점이 있다.

2.3 결정 트리를 이용한 방법

결정 트리는 기계 학습 분야에서 널리 사용되는 규칙 표현 방법으로 객체를 분류하는 규칙들이 트리 형태로 나타난다. 모든 예제에 대하여 그 예제가 속하는 범주가 결정되고, 속성을 의미하는 각각의 노드는 속성 값에 따라서 서로 다른 링크로 분리되므로, 트리의 뿌리에서부터 단말 노드까지의 경로는 하나의 규칙으로 변환이 가능하다[12].

2.4 SVM(Support Vector Machine)을 이용 방법

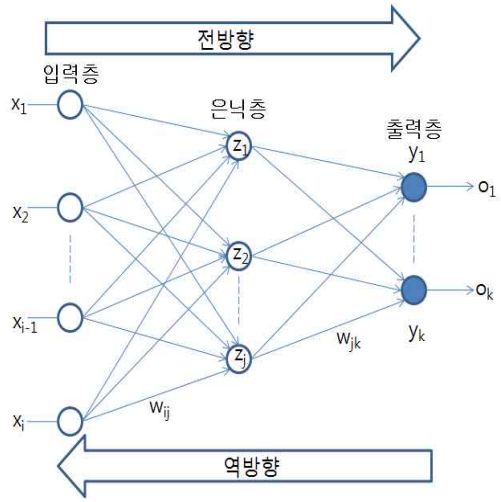
SVM은 학습 문서 데이터 셋을 통하여 생성되는 양성 자질(positive feature)과 음성자질(negative feature)을 벡터 공간으로 표현한다. 이들 간의 차이를 극명하게 나타내는 벡터를 찾는 방법이다. 최근 패턴 인식 연구 분야에서 많이 이용되고 있다[13, 14].

3. BPNN 알고리즘과 KNN 알고리즘

신경망 알고리즘은 인간의 신경조직을 응용하여 만들어진 알고리즘이다. 신경조직의 기본 구성요소는 뉴런이며 뉴런의 생물학적 특성을 수학적으로 표현한 것이 신경망 처리소자인 퍼셉트론이다. 신경조직이 다수의 뉴런으로 구성되듯이 신경망은 다수의 퍼셉트론으로 구성되어진다. 하나의 퍼셉트론은 입력되는 값에 가중치를 부여하고, 활성함수를 이용하여 출력값을 조절한다. 이미 답을 알고 있는 문제 여러 가지를 입력하여 주어진 문제에 대한 해답을 찾을 때까지 학습을 수행한다. 해답을 찾아가는 과정에서 가중치와 활성함수를 조정하여 분류 시스템을 완성한다[11, 12, 14].

역전파 신경망(BPNN) 알고리즘은 <그림 1>에서와 보이는 바와 같이 다층 퍼셉트론이다. 입력층과 출력층 사이에 은닉층이 있으며, 각 층은 가중치로 연결된다.

역전파 신경망은 전방 단계(forward)와 후향(back) 단계를 반복적으로 수행하며 총 오차의 합이 정해진 오차의 기중치에 도달할 때까지 실행한다. 각 층의 연결강도 조절은 일반 델타 규칙을 이용한다. 학습을 위하여 학습용 문서 데이터 셋을 입력하고 전방 계산(forward computation)으로 출력을 산출할 수 있다. 목표 출력값과 실제 출력값 사이의 오류를 계산한 후 출력층에서 시작하여 반대방향으로 전진하며 오류를 줄이는 쪽으로 연결 가중치를 갱신하며 전파(back propagation)한다. 연결강도를 조정한 후에 다시 입력값을 넣어 계산하여 나온 출력값의 오차는 처음 입력했던 값보다 작은 값이 나온다. 이러한 과정을 반복하여 총 오차의 합이 정해진 오차의 기준치까지 도달할 때까지 실행하여 시스템을 안정화시킨다.



<그림 1> 역전파 신경망의 구조

3.1 입력값과 출력값

전방계산에 사용되는 입력값과 출력함수는 아래 식을 사용하여 계산한다. 퍼셉트론 j에 net_j가 입력되며 식 (3.2)를 통하여 출력값 O_j을 구한다.

$$net_j = \sum_i w_{ij} O_i \tag{3.1}$$

$$O_j = f(net_j + \theta_j) \tag{3.2}$$

w_{ij}는 j번째 퍼셉트론에 들어오는 i번째 퍼셉트론의 가중치를 나타내며, $f(net_j + \theta_j)$는 퍼셉트론의 활성함수이다. O_i와 O_j는 i와 j에 들어오는 이전 퍼셉트론의 출력값이다. θ_j는 퍼셉트론의 입력바이어스 값이다.

3.2 에러 계산

출력값으로 계산되는 절대오차 E는 식 (3.3)을 통하여 구할 수 있다.

$$E = \frac{1}{2} \sum_l \sqrt{(t_l - O_l)^2} \tag{3.3}$$

식 (2.4)를 통하여 평균 절대 오차 Em을 구한다. 여기서 n은 입력되는 학습데이터 수이다.

$$E_m = \frac{1}{2n}\sum_l \sqrt{(t_l - O_l)^2} \quad (3.4)$$

평균 절대 오차는 학습률을 평가하는데 사용된다. 평균 절대 오차가 최소화 될 때까지 학습은 진행된다.

3.3 역방향 오차 수정

오차에 따라 연결가중치를 조정하는 델타규칙은 식 (3.5)와 (3.6)으로 나타낼 수 있다.

$$\delta_l = \lambda(t_l - O_l)f'(O_l) \quad (3.5)$$

$$\delta_j = \lambda\sum_i \delta_l w_{ji} f'(O_j) \quad (3.6)$$

연결가중치는 w_{ij}와 바이어스 θ_j는 식 (3.7)과 (3.8)을 통하여 구할 수 있다. 여기에서 k는 에포크(epoch)의 수, η는 학습률을 나타낸다.

$$w_{ji}(k+1) = w_{ji}(k) + \eta\delta_j y_i \quad (3.7)$$

$$\theta_i(k+1) = \theta_i(k) + \eta\delta_i \quad 3.8)$$

3.4 KNN알고리즘

문서분류에 사용되는 대표적인 알고리즘으로 분류할 문서 X와 범주별 표본문서 Y사이의 유사도를 통하여 문서를 분류한다. 유사도를 계산하는 방법은 여러 가지가 있으며 유클리드거리와 Cosine Similarity를 이용하여 계산한다. 문서 X와 분류 표본 문서들간의 거리를 식 (3.9)로 계산한다. 문서 X와 분류 표본 문서들간의 Cosine Similarity는 식 (3.10)로 계산한다.

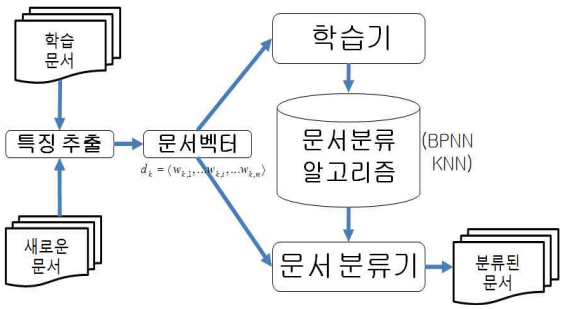
$$Sim_{XY} = \sqrt{\sum_{i=1}^{n}(X_i - Y_i)^2} \quad (3.9)$$

$$Sim_{XY} = \frac{\overrightarrow{d_X} \cdot \overrightarrow{d_Y}}{|d_X| \times |d_Y|} \quad (3.10)$$

새로운 문서와 모든 문서서의 유사도를 계산한 후 계산된 문서들 중 가장 가까운 K개의 문서가 위치한 범주로 분류할 문서 X를 할당한다[8, 9, 10].

4. 문서분류의 전체 구조

자동 분류 알고리즘은 문서의 특징을 추출하는 단계, 추출된 특징을 분석하여 학습하는 단계, 학습을 완료하고 새로운 문서를 분류하는 단계로 이루어진다.



<그림 2> 문서분류 과정

문서의 특징을 추출하는 단계에서는 문서의 내용을 가장 잘 나타내는 특징을 선택하는 것이 중요하다. 문서의 특징을 나타내는 것으로는 TF/IDF, 정보이득(information gain), 상호정보(mutual information)등이 사용된다. 추출된 특징은 벡터화하여 저장하고 인공지능 알고리즘을 이용하여 학습을 수행하게 된다. 학습이 완료되면 입력되는 문서의 특징을 분석하여 정해진 범주에 배정하게 된다.

4.1 특징추출과 문서벡터

본 논문에서 사용된 문서의 특징으로 TF 값을 사용하였다. TF(Term Frequency)는 문서내의 단어의 빈도수를 나타낸다. TF값을 이용하여 문서내의 단어의 가중치를 계산하여 문서의 특징을 가지는 문서 벡터를 생성한다. k번째 문서 벡터 d는 n개의 단어 가중치를 가진다.

$$d_k = \langle w_{k,1}, \ldots w_{k,i}, \ldots w_{k,n} \rangle \quad (4.1)$$

4.2 Singular Value Decomposition(SVD)

식 (4.1)로 생성된 문서벡터는 m×n개의 행렬형태가 된다. 원본 벡터를 이용하여 KNN과 BPNN을 수행했을 때 가장 좋은 결과를 보인다 하지만 원본 벡터의 크기가 대단히 커서 학습과 분류 속도는 현저히 떨어

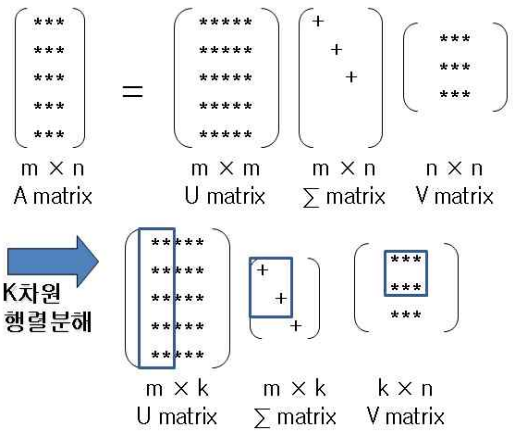
지게 된다. 원본 벡터의 의미는 유지하면서 벡터의 크기는 줄이기 위하여 본 논문에서는 SVD를 이용하였다. SVD를 사용하여 원본 벡터의 차원을 줄이면 분류 성능과 속도 모두를 만족할 수 있다.

원본 벡터 A는 식 (4.2)에 의하여 $U\sum V^T$ 로 분해할 수 있다.

$$A = U\sum V^T \tag{4.2}$$

Σ는 단일 값을 갖는 m×n 대각 행렬(diagonal matrix)이다. U는 단어 간의 상관행렬(association matrix)를 나타내는 m×m 고유 벡터 행렬(orthogonal vector matrix)이다. V는 문서간의 상관행렬(association matrix)를 나타내는 n×n 고유 벡터 행렬이다. m보다 작은 k를 선택하면 식 (4.3)을 이용하여 원본 벡터와 가장 유사한 저차원 벡터 A_k를 얻을 수 있다.

$$A_k = U_k\sum_k V_k^T \tag{4.3}$$



<그림 3> SVD를 이용한 저차원 행렬분해

SVD는 Latent sematic indexing(LSI)에서 사용되는 수학개념이다. LSI는 정보검색 모델로 제안되었으며 현재 분류분야에서 사용되고 있다. LSI 모델에서는 사용자의 질의도 문서와 같이 단어들의 집합인 k차원의 행렬로 나타낼 수 있다.

$$\hat{q} = q^T U_k\sum_k^{-1} \tag{4.4}$$

또한 각각의 문서들은 다음 식과 같이 나타낼 수 있다.

$$\hat{d} = d^T U_k\sum_k^{-1} \tag{4.5}$$

본 논문에서는 LSI모델에서 사용하는 $\sum_k^{-1}$ 를 빼고 문서벡터에 단어 간의 의미정보를 가지고 있는 U 벡터만을 곱하여 입력 가중치 벡터 $\hat{d}$를 k개의 차원으로 줄여서 사용한다.

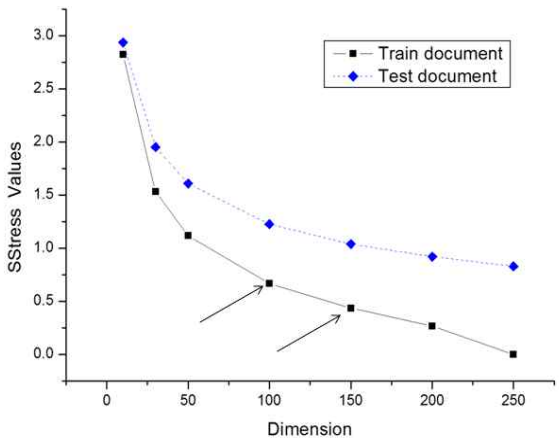
$$\hat{d} = d^T U_k \tag{4.6}$$

본 논문에서는 원본벡터보다 $\hat{d}$ 벡터를 이용하여 보다 적은 차원의 입력 벡터를 이용하여 같은 성능의 분류기를 만들었다.

4.3 SStress Test

저차원의 벡터 $\hat{d}$를 만드는 데는 감소된 차원 k값을 선정하는 방법이 중요하다. LSI 모델에서는 $\sum_k^{-1}$ 행렬이 대각행렬의 형태임을 이용하여 대각에 값을 가지고 있는 k값을 선정하여 새로운 차원으로 사상한다.

본 논문에서는 직교 k값을 선정하는 데에 사용한 방법은 전체 변동에 대한 공헌 정도를 고려하여 <그림 4>와 같은 그래프를 통해 판독한다.



<그림 4> 각각 250개의 학습문서와 실험 문서 SStress 곡선

$$SSTRESS = \left\{ \frac{\sum_i \sum_j (d_{ij} - \hat{d}_{ij})^2}{\sum_i \sum_j (d_{ij})^2} \right\}^{\frac{1}{2}} \quad (4.7)$$

그래프에서 사용된 그래프는 식 (4.7)의 Kruskal이 사용한 SSTRESS 측정 알고리즘이다. 이 식을 이용하면 특이값 분해 후의 특이값 설명정도를 그림 4와 같은 그래프를 통하여 나타낼 수 있다. 특이값의 수를 선택할 때 일반적인 기준은 특이값 간의 설명력 차이가 커 급격하게 그래프의 기울기가 감소하다가 다시 완만해지기 전까지를 택하는 것이다.

5. 실험 데이터 셋과 실험결과

이 장에서는 분류에 사용된 실험 데이터 셋의 특성과 분류 주제에 대하여 설명하고, SStress 곡선과 문서 분류 결과의 상관 관계를 다양한 실험결과를 통하여 보여줄 것이다.

5.1 실험 데이터 셋

분류시스템의 학습과 분류실험을 위하여 사용된 데이터 셋은 '한국일보-20000'(HKIB-20000)이다[13]. 한국일보-20000 실험문서집합은 한국일보-40075 집합의 기사 중 20,000건을 별도로 추출하여 분류체계를 보다 현실적으로 수정하였으며, 3단계 분류체계의 모든 노드에 기사를 할당하여 구축한 계층적 분류체계의 문서범주화용 실험문서집합이다.

한국일보-40075 실험문서집합은 한국일보가 제공한 1998~1999년의 2년간 신문기사를 바탕으로 40,075개의 각 문서별로 3단계 분류체계의 말단 범주를 부여하여 구축하였다. 이 문서집합들은 충남대학교 이석훈 교수 연구실과 한국과학기술정보연구원이 공동으로 제작한 것으로 한국일보-40075는 비계층적 분류체계를 가지는 단일범주 실험문서집합인 반면, 한국일보-20000은 계층적 분류체계를 가지는 다중범주 실험문서집합으로 요약할 수 있다.

<표 1>은 본 논문에서 문서분류를 위하여 선택한 140개 이상의 문서를 가지는 10개의 범주이다. 10개의 범주에서문 학습문서와 실험문서의 개수를 달리하여

<표 1> 실험에 사용된 범주

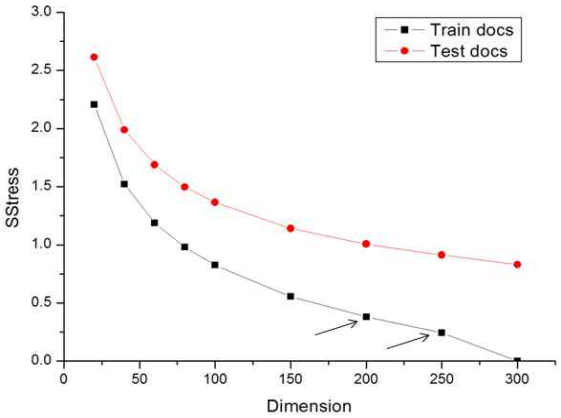
범주 이름	문서갯수
건강의학_의약학_질병(암외의질병)	196
경제금융_은행(금융업계동향)	282
과학자연과학_화학	188
문화종교_스포츠_야구	325
문화종교_공연_방송연예	518
문화종교_종교_불교	141
사회사회질서_군대	774
산업건설업_토목	272
산업제조업_컴퓨터	481
여가실외_여행관광	182

10차례 이상의 분류 실험을 하였다. 분류 성능을 측정하기 위하여 재현율과 정확률을 이용하는 평균 F1 값을 사용하였다.

$$F_1 = \frac{2 \times (Recall \times Precision)}{Recall + Precision} \quad (5.1)$$

5.2 실험 결과

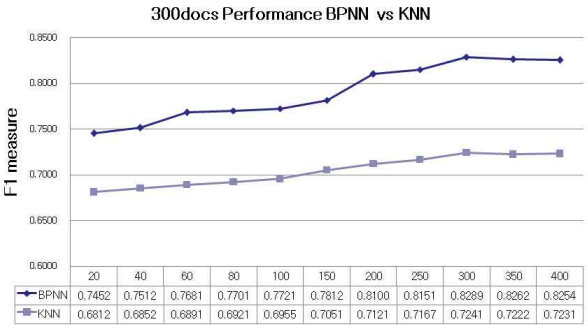
<그림 5>는 각각 300개의 문서를 SVD를 통하여 벡터 분해할 때 k를 구하기 위한 SStress 그래프를 이다. 이 그래프를 따르면 k가 200이나 300일 때 최적의 차원 감소 효과를 가지게 된다.



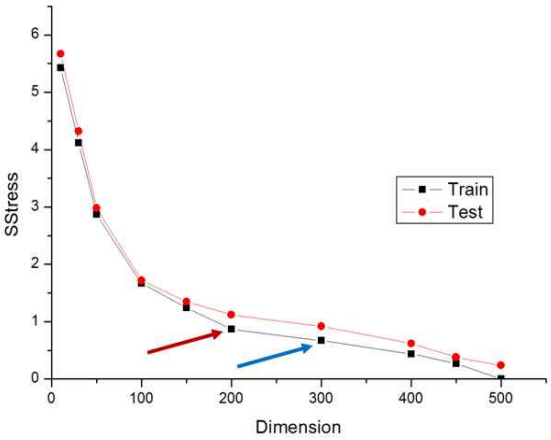
<그림 5> 300개의 학습문서와 300개 실험문서의 SStress 곡선

<그림 6>은 300개의 문서를 통하여 학습을 수행하고 300개의 문서를 분류기를 통하여 분류를 했을 때

KNN과 BPNN 알고리즘의 성능 결과를 보여준다. 실험 결과에 따르면 차원 200이후 성능이 크게 나아지고 있음을 볼 수 있다. 이것은 k를 선정함에 SStress 그래프가 유용함을 증명한다.



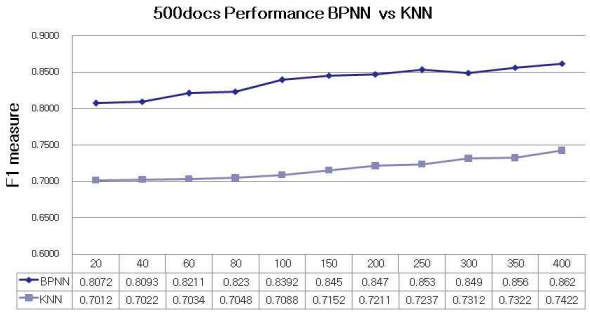
<그림 6> 300개의 문서 분류 성능



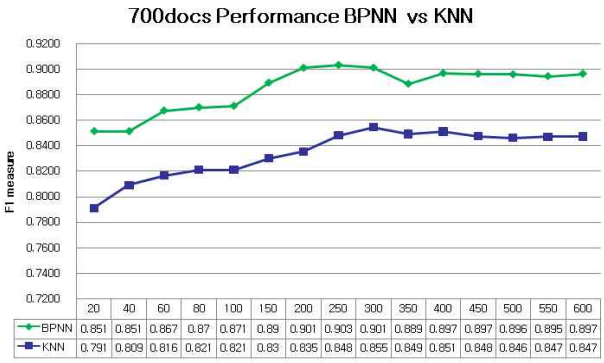
<그림 7> 500개의 학습문서와 500개 실험문서의 SStress 곡선

<그림 7>은 각각 500개의 문서를 SVD를 통하여 벡터 분해할 때 k를 구하기 위한 SStress 그래프를이다. 이 그래프를 따르면 k가 200이나 300일 때 최적의 차원 감소 효과를 가지게 된다.

<그림 8>은 500개의 문서를 통하여 학습을 수행하고 500개의 문서를 분류기를 통하여 분류를 했을 때 KNN과 BPNN 알고리즘의 성능 결과를 보여준다. KNN보다는 뛰어난 성능을 보인다. 차원이 200과 300일 때의 성능이 비슷하게 나오는 것은 SStress를 통하여 차원을 정하는 것이 유용함을 증명한다.

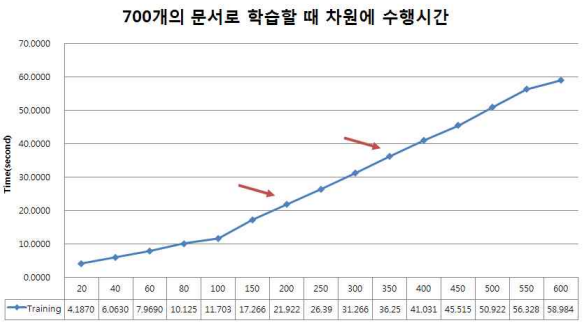


<그림 8> 500개의 문서 분류 성능



<그림 9> 700개의 문서 분류 성능

<그림 9>는 700개의 문서를 통하여 학습을 수행하고 700개의 문서를 분류기를 통하여 분류를 했을 때 KNN과 BPNN 알고리즘의 성능 결과를 보여준다. KNN보다는 뛰어난 성능을 보인다. SStress 곡선에 따르면 차원이 200과 400일 때 최적의 값을 가지게 된다.



<그림 10> 차원에 따른 학습 시간

<그림 10>는 700개의 문서를 이용하여 학습을 수행할 때 차원에 따른 시간을 보여준다. 원본벡터를 SVD

를 이용하여 저차원으로 줄이면 속도가 현저히 빨라짐을 볼 수 있다. <그림 9>에서 보듯이 차원이 200과 600의 성능이 비슷하다. 학습 속도를 비교하면 <그림 10>과 같이 속도가 200에서 약 2배정도 빨라지는 것을 볼 수 있다.

6. 결 론

일반적으로 문서분류에 있어서 사용되는 알고리즘은 KNN이다. KNN은 문서간의 거리를 비교하여 가까운 거리에 있는 문서들이 많은 범주에 문서를 분류한다. 실험 결과에서 볼 수 있듯이 KNN은 비정형 문서에서는 그다지 좋은 결과를 보여주지 못한다. 이에 반하여 BPNN은 90%가 넘는 분류성능을 보여주었다. KNN에 비교하여 보았을 때 모든 테스트 셋에서 약 10%정도 향상된 성능을 보인다.

학습과 분류에 원본 문서 벡터를 이용하면 좋은 분류 성능을 얻을 수 있다. 하지만 원본 벡터를 이용하면 많은 계산을 수행하여 느린 속도가 단점이었다. 본 논문에서는 SVD를 이용하여 같은 의미를 가지면서 원본 문서벡터보다 작은 차원의 벡터를 이용하여 높은 분류성능과 빠른 수행속도를 가진 분류기를 완성하였다.

참 고 문 헌

[1] C. Apte and F.Damerau, "Automated learning of decision rules for text categorization", *ACM Transactions on Information Systems*, Vol. 12, No.3, pp.233-251, 1994

[2] E. D. Wiener. A neural network approach to topic spotting in text. *Master's thesis*, Department of Computer Science, University of Colorado at Boulder, Boulder, US, 1995.

[3] D. E. Rumelhart, R. Durbin, R. Goldenand, and Y. Chauvin. Backpropagation: The basic theory. In M. C. Mozer and D. E. Rumelhart, editors, *Mathematical Perspectives on Neural Networks*, Lawrence Associates, Hillsdale, NJ, pp 533 - 566. 1996.

[4] D. E. Rumelhart and J. L. McClelland. Parallel distributed processing: exploration in the microstructure cognition, volume vols. 1 & 2. *MIT Press*, 1986.

[5] Ruiz, M. E., Srinivasan, P. Automatic Text Categorization Using Neural Network, in: *Proceedings of the 8th ASIS SIG/CR Workshop on Classification Research*, pp. 59-72. 1998.

[6] Noorinaeini, A. and Lehto, M.R. "Hybrid singular value decomposition; a model of human text classification", *Int. J. of Human Factors Modelling and Simulation*, Vol. 1,No.1, pp.95 - 118. 2006.

[7] Y. Yany, "Noise reduction in a statistical approach to text categorization," in *Proc. of the 18th ACM International Conference on Rexorch ond Development in Informorion Retrieval*, New York, pp. 256.263. 1995.

[8] Dudani, S.A. The distance-weighted k-nearest -neighbor rule. *IEEE Trans. Syst. Man Cybern.*, SMC-6: 325 - 327, 1976.

[9] Li BL, Yu SW, Qin Lu. An improved k-nearest neighbor algorithm for text categorization. In: Sun MS, Yao TS, Yuan CF, eds. *Proc. of the 20th Int'l Conf. on Computer Processing of Oriental Languages*. Beijing: Tsinghua University Press, 2003.

[10] Songbo Tan. An effective refinement strategy for KNN text classifier. *Expert Systems with Applications*, Volume 30, Issue 2, February, pp. 290-298. 2006.

[11] 김대수, *신경망 이론과 응용(I, II)*, 하이테크 정보, 1992

[12] 오일석, *패턴 인식*, 교보문고, 2008

[13] 한국일보 문서범주화 실험문서집합. http://www.kristalinfo.com/TestCollections/readme_hkib.pdf

[14] Nakayama, M., & Shimizu, Y. Subject categorization for web educational resources using MLP. In *Proceedings of 11th European symposium on artificial neural networks*, pp. 9 - 14. 2003.

리 청 화 (Chenghua Li)

• 학생회원
• 2004년 : 중국 풍남민족대학교 컴퓨터 공학 학사 졸업
• 2006년 : 전북대학교 정보통신학과 석사 졸업
• 2009년 8월 : 전북대학교 정보통신공학과 박사 졸업
• 관심분야 : 정보검색, 시멘틱 웹, 온톨로지

변 동 률 (Dong Ryul Byun)

• 학생회원
• 1998년 2월 : 전북대학교 화학공학과 (화학공학사)
• 2003년 2월 : 전북대학교 정보통신학과 (정보통신석사)
• 2005년 2월 : 전북대학교 정보통신공학과 (박사 수료)
• 2005년 3월 ~ 현재 : 전북대학교 박사과정 중
• 관심분야 : 정보검색, 시멘틱 웹, 온톨로지

박 순 철 (Soon Chol Park)

• 평생회원
• 1979년 2월 : 인하대학교 공과대학 (공학사)
• 1991년 12월 : (미국)루이지아나 주립대학(전산학박사)
• 1991년-1993년 : 한국전자통신연구원 근무
• 1993년 ~ 현재 : 전북대학교 전자정보공학부 교수
• 관심분야 : 정보검색, 시멘틱 웹, 온톨로지

논 문 접 수 일 : 2010년 03월 11일
1차수정완료일 : 2010년 04월 12일
2차수정완료일 : 2010년 05월 03일
게 재 확 정 일 : 2010년 06월 10일