韓國 컴퓨터情報學會 論文誌
第11卷 第2號, 2006. 5.

2006-11-2-1-3

텍스트 마이닝을 이용한 XML 문서 분류 기술

김 천 식*, 홍 유 식**

Classification Techniques for XML Document Using Text Mining

Cheonshik Kim*, You-Sik, Hong**

요 약

인터넷에는 많은 문서가 있고 지금도 새로운 문서가 만들어지고 있다. 따라서 인터넷에 존재하는 문서를 의미 있게 분류하는 것은 향후 문서의 관리 및 질의처리에서 중요한 문제이다. 하지만 지금까지 대부분은 키워드에 기초한 문서 분류 방법을 사용하고 있다. 이 방법은 문서를 효율적으로 분류하지 못했다. 또한 의미를 포함한 문서의 분류를 하지 못한다. 사람이 문서를 꼼꼼하게 읽어서 문서를 분류하는 방법이 최선이지만, 시간적인 면이나 효율성에 문제가 있다. 따라서 본 논문에서는 신경망 알고리즘과 C4.5 알고리즘을 이용하여 문서를 분류하고자 한다. 실험 데이터로 XML로 만들어진 이력서 데이터를 사용하여 실험하였다. 실험결과 문서 분류에 가능성을 보였다. 또한, 다양한 문서 분류 응용에 적용하여 좋은 결과를 얻을 것으로 기대한다.

Abstract

Millions of documents are already on the Internet, and new documents are being formed all the time. This poses a very important problem in the management and querying of documents to classify them on the Internet by the most suitable means. However, most users have been using the document classification method based on a keyword. This method does not classify documents efficiently, and there is a weakness in the category of document that includes meaning. Document classification by a person can be very correct sometimes and often times is required. Therefore, in this paper, We wish to classify documents by using a neural network algorithm and C4.5 algorithms. We used resume data forming by XML for a document classification experiment. The result showed excellent possibilities in the document category. Therefore, We expect an applicable solution for various document classification problems.

▸ Keyword : data mining, neural network, C4.5

• 제1저자 : 김천식
• 접수일 : 2006.02.6, 심사완료일 : 2006.05.18
* 안양대학교 디지털미디어공학전공 교수, ** 상지대학교 컴퓨터정보공학부 교수

Ⅰ. 서 론

인터넷에는 현재 수많은 문서가 존재한다. 이렇게 많은 문서에 정보 검색을 위한 질의를 할 경우, 질의 결과 또한 엄청난 양 일 것이다. 이렇게 많은 검색된 문서는 사용자가 검색하기를 원하는 데이터라면 좋겠지만 대부분 검색의도와는 관계가 없는 데이터가 검색되는 경우도 많다.

왜냐하면 현재 대부분의 웹 문서가 구조적인 형태를 갖추고 있는 웹 문서가 아니므로 대부분의 검색 엔진은 결국 질의 방법으로서 키워드에 기초한 문서의 검색 및 결과를 얻게 된다. 키워드에 기초한 검색 방법은 검색 대상이 되는 웹 페이지가 사용자가 질의한 키워드를 얼마나 많이 포함하는가에 전적으로 의존하게 된다. 하지만 키워드를 많이 포함하더라도 정작 사용자가 원하는 웹 페이지가 아닐 수 있다.

현재 대부분의 웹 문서가 HTML로 되어 있다. HTML 언어는 웹 문서를 보기 좋게 만들어주는 언어의 측면을 갖고 있다. 따라서 구조적인 어떤 면을 기대하기는 어렵다. 이러한 이유 때문에 웹에서 무엇인가를 검색하려고 할 때, 가장 중요시 될 수 있는 요인은 결국 사용자의 질의에 대하여 키워드를 얼마나 많이 갖고 있느냐가 주요시 될 수밖에 없다. 하지만 현재 웹 문서의 표준으로서 XML[1]이 채택되었고, XML은 구조적인 문서 기술력인 자기서술적인 특성을 갖고 있다.

XML이 구조적인 특성을 갖고 있다고는 하지만 관계형 데이터베이스와 같은 정도의 구조적인 특성은 아니다. XML과 관계형 데이터베이스의 차이점은 XML 문서는 그래프형태를 갖지만, 관계형 데이터베이스는 테이블 형태라는 것이다. XML의 이러한 구조적인 면을 활용하여 대량의 문서를 효과적으로 분류할 수 있다[21].

하루에도 전 세계적으로 수 십 만개의 웹 문서가 생성된다. 또한 각 기관에서는 종류를 알 수 없는 수 십 만개의 문서를 키워드로 식별해서 문서를 분석하여 분류하고 있다. 물론, 이 방법은 보편적으로 많이 사용하고 있고 분류가 간단하다는 장점이 있다. 하지만 이와 같은 문서 분류 방법은 정확성이 좋지 않은 단점이 있다. 따라서, 이를 개선한다면 기업이나 각 기관에서는 보다 정확한 문서분류 결과를 얻을 것으로 기대한다.

따라서 본 논문에서는 문서 분류의 실험 대상으로서 회사의 온라인 입사 지원서를 사용한다. 대기업의 경우 입사 지원이 상대적으로 크기 때문에 입사지원서에 대한 평가나 분류가 쉽지 않다. 그러므로 본 논문에서는 기계학습[9,10,11, 12,13]을 통해서 문서를 효율적으로 분류할 수 있는 방법을 제안하고 향후 다양한 분야에서 이를 활용할 수 있는 방안을 제안하고자한다.

Ⅱ. 관련 연구

(P. Willet)[2]의 논문은 문서를 응집하는 여러 가지 알고리즘을 소개했다. 점진적인 계층적 문서 응집 알고리즘은 현재 가장 많이 사용하는 방법일 것이다. K-평균[3], kNN[19][20] 알고리즘과 같은 선형시간 알고리즘은 온라인 문서 응집에 장점이 있다. 단어의 순서가 의미가 있다고 가정한다면 구(clause)는 인터넷의 많은 문서를 군집화(Clustering)하는데 유용하게 사용된다[4]. 비트맵 인덱스는 질의를 최적화하는데 유용하게 사용된다[5,6,7].

일반적으로 텍스트마이닝 도구는 사용자가 시나리오를 작성하고 데이터준비와 분석을 도울 수 있도록 상용 혹은 비상용으로 많이 개발되어 있다[15][16] [17][18]. 그러나, 아직까지 XML 문서 분류에 효율적인 기술이 제안된 적이 없다.

따라서, 본 논문에서는 신경망과 C4.5 알고리즘을 이용하여 XML 문서 분류를 실험했다. 실험을 위한 목표는 문서 분류에 오차를 최소 줄이는 방법을 찾는 것이다. 이와 같은 실험을 위해 사용한 데이터는 XML 이력서 데이터이다.

2.1 신경망(neural networks)

신경망은 분명 강력한 예측모델이다. 그러나 사용자의 어려움, 결과해석의 어려움 등을 동시에 수반하기도 한다. 신경망 모델은 비록 전문가라 할지라도 완벽히 이해할 수 없는 아주 복잡한 모델이다. 그 안에서는 예측변수 값을 이용하여 엄청난 양의 계산들이 행하여지며, 모델 자체도 이들 수치에 의해서 표현된다. 신경망 모델이 제공하는 예측결과 또한 수치로 표현되기 때문에 만약 목표변수의 값이 범주 형이라면 결과수치를 변환하는 작업이 필요하다. 예를 들어 의류제조

업체에서 고객의 청바지 색상 선호도를 예측하고자 한다면 색상 선호도라는 목표변수가 취하는 '파란색', '흰색', '검정색' 등의 값은 학습을 위해 수치로 변환시켜야 하며, 예측결과를 해석하기 위해서는 다시 본래의 값으로 되돌려야 한다.

이와 같은 복잡성을 줄이고 모델에 대한 이해도를 높이고자 하는 연구와 노력이 꾸준히 진행되어 왔으나 아직 초보단계에 불과하다. 그러나 신경망을 포함한 다수의 데이터 마이닝 기법들이 실제로 현업에서 사용되고 있으며, 이들이 예측하는 결과를 토대로 많은 기업들이 엄청난 액수의 자금을 투자한다는 점을 감안하면 모델의 이해도와 성능을 증대시키고자 하는 노력은 계속되어야 하며 한시라도 간과되어서는 안 된다.

예측을 위해 신경망은 입력마디라 불리는 예측변수들로부터 값들을 받아들인다. 그리고 이 값들에 링크에 지정된 값, 즉 강도를 곱하고 서로 합한 후 그 결과를 출력마디에 보낸다. 마지막으로 사전에 정의된 한계함수를 적용하여 최종 결과를 도출하는데, 이것이 바로 예측치가 된다.

2.2 의사결정트리(decision tree)

1970년대 후반 J.Ross Quinlan 박사는 'ID3'라는 의사결정나무 알고리즘을 발표하였는데, 이 알고리즘은 간단하면서도 탁월한 성능으로 인해 아직까지도 학습이나 추론에 기반을 둔 시스템에 거의 독보적으로 사용되고 있다. 특히 초창기의 ID3는 학습을 통한 체스게임 전략 수립에 주로 사용되었으나, 차츰 각종 산업분야를 망라한 다양한 비즈니스 영역에 폭넓게 사용되었으며, 이후 여러 번의 수정 및 보완 작업을 거쳐 오늘날의 모습을 갖추게 되었다.

ID3는 정보수익 이라는 수치에 근거하여 질문, 즉 예측변수와 해당 값을 선택하고 데이터를 분할하여 나무를 확장한다. 여기에서 정보수익이란 "데이터를 분할하기 전에 레코드들을 올바로 분류하기 위해 필요한 정보 양과 분할 후 필요로 하는 정보의 양"의 차이로 정의한다. 따라서 데이터를 분할한 후 각 세부군 에 속한 레코드를 올바로 분류하기 위해 필요한 정보의 양이 분할 전에 비해 급격히 줄어들었다면 각 세부군의 데이터 동질성이 상당히 높아졌다는 것을 의미한다.

C4.5는 ID3를 보완하고 다음과 같은 새로운 기능을 추가한 버전이다.

예측변수에 일부 결손 값이 존재해도 무방
예측변수가 연속형 값을 취해도 무방
나무 가지치기 기능
나무를 규칙으로 전환하는 기능

III. 데이터마이닝 실행

SPSS, NCR, Daimler-Benz 등 여러 업계의 선도 회사들이 3년여 데이터 마이닝 작업의 표준화를 연구하여 1999년 발표한 것이 CRISP-DMCross-ndustry Standard Process for Data Mining, 데이터 마이닝 표준 실행과 정)이다. 현재 전 세계 데이터 마이닝 프로젝트의 50% 이상은 CRISP-DM을 따라 수행되고 있다[11].

CRISP-DM은 다섯 단계로 구성된다. 모든 과학적 과정이 그렇듯이 작업 단계들은 항상 전진하는 것이 아니라 때로는 후진하기도 하고 전체가 사이클이 되어 돌기도 한다.

① 비즈니스 이해

해당하는 업무를 이해하는 단계이다. 예를 들어 보험업, 신용카드 업, 유통업 등 그 분야의 기본 지식을 각종 참고 자료와 현업 책임자와의 커뮤니케이션을 통하여 이해해야 한다. 그 중에서 데이터 마이닝으로 접근할 수 있는 문제를 파악한다.

② 데이터 이해

현업이 보유. 관리하고 있는 데이터를 이해하는 단계이다. 레코드의 수, 변수의 종류, 자료의 질, 데이터 관리 체계 등을 파악한다. 흔히 변수의 종류가 수백 가지에 이르고 데이터가 여러 개의 컴퓨터 서버에 분산되어 있기 때문에 한 조직의 정보체계를 정확히 이해하는 데는 시간이 걸리기 마련이다.

③ 데이터 준비

자료를 컴퓨터 서버로부터 내려 받고나서 분석 가능한 상태로 만들기 위하여 데이터 정제작업을 한다. 예컨대 고객이름과 주소, 전화번호를 1개의 표준 형태로 정리한다. 이 단계에 많은 일손이 필요로 하므로 전체 프로젝트 일정의 50% 이상을 차지하는 것이 보통이다. 또한 우리가 필요로 하는 변수를 만들어내는 일도 필요하다. 예컨대 고객별 총 거래액이 필요하다면 고객별로 개별 거래기록을 묶는 작업을 해야 한다.

④ 모델링

자료기술과 탐색을 포함하여 필요로 한 각종 모델링을 한다. 여기에는 신경망, 나무형 모델(decision tree) 등의 지도학습(supervised learning), 군집화(clustering), 연관성

(association)분석 등의 비지도학습(unsupervised learning)이 포함된다.

⑤ 평가

앞 단계에서 생성된 모형이 잘 해석되는지, 독립적인 새 자료에 적용되는 경우도 재현가능한지를 검토한다.

⑥ 전개

검토가 끝난 모형을 실제 현업에 적용하는 단계이다. 예컨대 모든 고객에 대하여 이탈 가능성 점수를 산출하여 고객 관리자에게 전달하여 필요한 조치를 취하는 작업 등이다.

```
<!-- Standalone Parameter Entity Declarations used in the DTD -->
<!ENTITY % doctype "이력서">
<!ENTITY % p.date "년|월|일">
<!ENTITY % p.bth.day "나이|(%p.date;)">
<!ENTITY % p.pi.num "앞자리,뒷자리">
<!ENTITY % m.fig "EMPTY">
<!ENTITY % m.ph.addr
"국명|시|도|군|구|읍|면|동|우편번호|사서함번호|전화번호|팩스번호|전자우편번호">
<!-- Mixed Parameter Entity Declarations used in the DTD -->
<!ENTITY % m.addr "#PCDATA | %m.ph.addr;">
<!ENTITY % m.lib "(%p.date;)*, 내용,비고?">
<!-- Entities for common attributes -->
<!ENTITY % a.id "ID ID #IMPLIED">
<!ENTITY % a.lang "(한글|영어|한자|일본어) '한글'">
<!ENTITY % b.type "(양력|음력) '양력'">
<!-- Notation Declarations -->
<!NOTATION jpg PUBLIC "">
<!-- Element Delclarations -->
<!ELEMENT %doctype;
(이름+,사진?,주민등록번호,생년월일,주소,호적관계,(학력|경력|특기|자격|토익)*)
>
<!-- 이름 -->
<!ELEMENT 이름 (#PCDATA)>
<!ATTLIST 이름 언어 %a.lang;>
<!-- 사진 -->
<!ELEMENT 사진 %m.fig;>
<!ATTLIST 사진 %a.id;
NAME ENTITY #IMPLIED>
<!-- 주민등록번호 -->
<!ELEMENT 주민등록번호 (%p.pi.num;)>
<!ELEMENT 앞자리 (#PCDATA)>
<!ELEMENT 뒷자리 (#PCDATA)>
<!-- 생년월일 -->
<!ELEMENT 생년월일 (%p.bth.day;)+>
<!ATTLIST 생년월일 책력 %b.type;>
<!ELEMENT 나이 (#PCDATA)>
<!ELEMENT 년 (#PCDATA)>
<!ELEMENT 월 (#PCDATA)>
<!ELEMENT 일 (#PCDATA)>
<!-- 주소 -->
<!ELEMENT 주소 (%m.addr;)*>
<!ELEMENT 국명 (#PCDATA)>
<!ELEMENT 시 (#PCDATA)>
<!ELEMENT 도 (#PCDATA)>
<!ELEMENT 군 (#PCDATA)>
<!ELEMENT 구 (#PCDATA)>
<!ELEMENT 읍 (#PCDATA)>
<!ELEMENT 면 (#PCDATA)>
<!ELEMENT 동 (#PCDATA)>
<!ELEMENT 우편번호 (#PCDATA)>
<!ELEMENT 사서함번호 (#PCDATA)>
<!ELEMENT 전화번호 (#PCDATA)>
<!ELEMENT 팩스번호 (#PCDATA)>
<!ELEMENT 전자우편번호 (#PCDATA)>
<!-- 호적관계 -->
<!ELEMENT 호적관계 (호주성명,관계)>
<!ELEMENT 호주성명 (#PCDATA)>
<!ELEMENT 관계 (#PCDATA)>
<!-- 학력, 경력, 특기, 자격 -->
<!ELEMENT 학력 (%m.lib;)>
<!ELEMENT 경력 (%m.lib;)>
<!ELEMENT 특기 (%m.lib;)>
<!ELEMENT 자격 (%m.lib;)>
<!ELEMENT 토익 (%m.lib;)>
<!ELEMENT 내용 (#PCDATA)>
<!ELEMENT 비고 (#PCDATA)>
```

그림 1. 이력서 DTD
Fig. 1 resume DTD

(그림 1)은 본 논문에서 사용할 이력서 데이터의 구조를 표현하는 XML DTD이다. 이 구조의 데이터를 만들어 문서를 분류하는데 이용할 것이다.

Ⅳ. 신경망 기법을 문서 분류

본 논문에 사용할 문서 분류를 위한 예측은 다음과 같으며, X축은 분서 분류에 이용할 요소이고, Y축은 변수의 값(과거 데이터 값)을 의미 한다.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \cdots + \varepsilon \quad \cdots\cdots\cdots\cdots (1)$$

단, Y : 1차 서류 분류(합격, 불합격)

X_1 : 종속변수에 영향을 주는 요인1

X_2 : 종속변수에 영향을 주는 요인2

X_3 : 종속변수에 영향을 주는 요인3

⋮

X_{10}: 종속변수에 영향을 주는 요인10

본 논문에 사용된 학습 신경망 구조는 다음과 같다.

① offsets, weight를 초기화 한다

② input, target의 패턴을 신경망에 제시

③ 출력 신경세포들의 에러와 델타를 구해서 은닉 층으로 역 전파 한다.

ej = tj - aj .

δj = aj (1 - aj) ej

④ 역 전파된 델타로부터 은닉층 신경세포들의 에러와 델타를 구해서 역 전파한다.

$e_j = \sum_k w_{jk}\,\delta_k$

$\delta_j = a_j (1 - a_j) e_j$

⑤ 델타 규칙에 의해서 연결가중치를 조절한다.

W(new)ij = W(old)ij + αδiaj + βΔwij(old)

bias(new)ij = bias(old)ij + αδ. 1 + βΔbiasij(old)

⑥ 1-5 의 과정을 모든 입력패턴에 대해서 반복한다.

⑦ 4 과정을 신경망이 완전히 학습 될 때 까지 반복한다.

표 1 신경망을 이용한 XML 문서 분류
Table. 1 XML document classification that use neural net

입력 데이터 조건	
변수1	학력(상/중/하)? A11 : 상, A12 : 중, A13 : 하
변수2	경력 기간(상/중/하) ? A21 : 5년이하 상, A22 : 3년이하 중, A23 : 1년 이하
변수3	특기 유무? A31 : Yes, A32: No
변수4	관련 자격증 유무? A41: Yes, A42: No
변수5	토익 점수 (상, 중, 하)? A51: 1000점 이하, A52: 800점 이하, A52: 600점 이하
변수6	자기소개서? A61: 상, A62:중, A63:하

<표 1>은 1차 지원서의 통과 여부 결정을 예측하기 위한 6가지 서로 다른 조건을 입력 하였을 때 최종 1차 서류 통과 여부의 결정을 예측하는 과정을 나타내고 있다.

신경망 학습의 초기값을 설정하는 것은 중요한 문제다. 초기값을 적절하게 선택함으로써 학습오차가 작고 학습과정이 빠르게 수렴될 수 있기 때문이다. 일반적으로 신경망의 학습은 특정 초기값에서 시작한다.

그리고 학습률은 모수 값들을 어떻게 선택하느냐에 따라서 학습오차가 작으면서 학습과정이 빠르게 수렴 할 수도 있고 조기 포화점에 빠질 수도 있다. 그렇기 때문에 분석하고자 하는 자료에 적당한 모수를 선정하여 오차가 최소 값이면서 학습과정이 빠르게 수렴될 수 있게 학습하도록 하는 것은 매우 중요한 문제다.

그래서 제한적이지만 κ, θ, φ, μ (kappa, theta, phi, mu)만을 가지고 각 범위 0.1, 0.3, 0.5, 0.7, 0.9에 따라 모든 경우를 고려해서 임의의 경우로 실험을 해보았다. 그리고 학습시간을 각각 500회로 제한하였다.

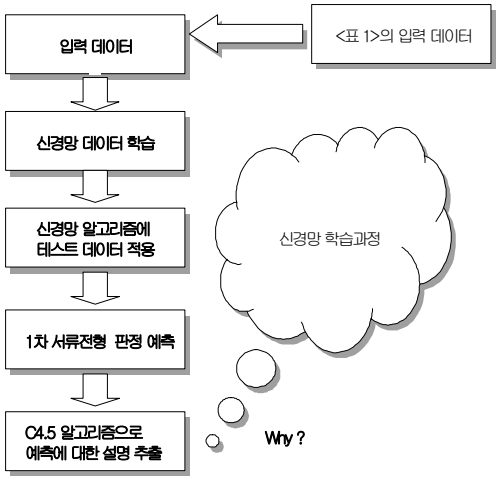

그림 2 문서 분류 판정 예측 과정
Fig. 2 Document classification decision estimate process

(그림 2)는 다음과 같은 처리를 수행한다.

① 신경망을 이용하여 6개의 서로 다른 조건 테스트 데이터를 학습시킨다.

② 6개의 테스트 데이터에 대하여 예측을 한 뒤 테스트 데이터와 예측 데이터의 오차를 계산한다.

$Z_1,\ Z_2,\ Z_3,\ \cdots,\ Z_n$: 테스트 데이터

$\hat{Z}_1,\ \hat{Z}_2,\ \hat{Z}_3,\ \cdots,\ \hat{Z}_n$: 예측값

$e_i = Z_i - \hat{Z}_i$ ..... (1)

i시점 시계열 테스트 자료와 예측 값에 대한 차이

$$Z_j' = Z_j + W(Z_j) \quad \cdots\cdots (2)$$

여기서, Z_j'는 j번째 특이 값으로 식별된 테스트 데이터 Z_j의 수정된 값을 의미한다.

③ 후처리로 데이터 마이닝의 C4.5 알고리즘을 이용한다.

신경망 알고리즘을 적용한 결과 <표 2>와 같은 결과를 얻는다. 즉, 입력 변수에 대해 가장 영향이 큰 요소가 intro 변수라는 결론을 얻었고, 신경망 알고리즘의 정확도는 72.886%이다.

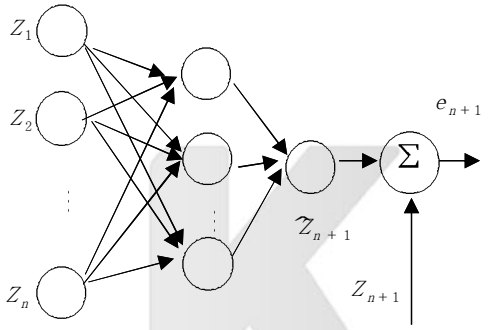

그림 3. 신경망을 이용한 문서 분류 결정 모형
Fig. 3 Document classification decision model that use neural net

표 2. 신경망 알고리즘 실험결과
Table. 2 Neural net algorithm experiment result

입력변수	영향
intro	0.666486
career	0.0318188
license	0.0212744
adademic	0.0211459
toeic	0.00488975
talent	0.00405935

V. C4.5 알고리즘을 이용한 문서 분류

본 논문에서는 신경망 알고리즘을 이용해서 최적의 입사 지원서 분류를 목적이다. 신경망 알고리즘은 자료 분석 분야에서 복잡한 구조를 가지고 있는 자료에 대하여 예측 문제를 해결하기 위한 유연한 비선형 모형의 하나로 분류될 수 있다. 인간의 신경생리학과 유사성 때문에 일반적으로 다른 통계적 예측모형에 비해 보다 흥미롭게 연구 되어지고 있다. 특히, 예측 기법으로써 로지스틱 회귀분석 보다 신경망의 우수함을 비교한 연구들이 고려되고 있다.

그러나 신경망은 미래의 목표 값을 예측하는데 있어 입력벡터의 값의 수나 형태를 결정할 수 있는 체계적인 방법의 결여와 모델의 분류가 어떻게 이루어지는지 명확하게 이해 할 수 없는 단점이 제시되고 있다, 이러한 단점을 해결하기 위하여 신경망에서 상징적 분류 규칙을 찾거나 의사결정 나무를 통하여 이해 할 수 있는 해석을 얻고자 하는 연구 등이 이루어졌다.

C4.5의사결정 나무를 형성하기 위하여 처음 수행하는 작업이 분할정복이다. 입력되는 훈련 집합이 성공적으로 분할되도록 모든 하부 집합에 하나의 클래스가 속하는 경우들로 구성될 때까지 나무를 형성한다. 노드를 분리하는 기준으로 정보이익비율(Information gain ratio)이 사용된다. 나무 구조의 결정 규칙을 생성하기 위하여 각 단계에서 p개의 설명변수 중 어느 것에 의하여 가지분리를 할 것인 가를 선택해야 한다. 이 때 결정 규칙들은 각기 다른 기준을 쓰는데, C4.5는 엔트로피 기준을 사용한다. 엔트로피(entropy)는 열역학에서 쓰는 개념으로 무질서도에 대한 측도이다. 자료집합 T가 Y에 의하여 k개의 범주로 분할되고 범주 비율이 p1, ..., pk라고 하자. T의 엔트로피는

$$Entropy(T) = -\sum_{l=1}^{k} p_l \log p_l$$

로 정의된다[11].

C4.5 모형은 엔트로피 기준에서 가장 엔트로피를 낮추는 분리 변수를 찾고자 한다. 분리변수를 찾음으로서 가장 성

취도가 좋은 변수 및 수준을 찾는 것이 나무규칙 생성 알고리즘이다. 본 논문에서는 클레멘타인 패키지를 이용하여 C4.5 알고리즘을 적용하여 신경망 알고리즘에서 설명할 수 없는 문서 분류에 영향을 주는 요인을 알아낼 수 있다.

실험 데이터는 <표1>의 데이터를 이용하였고, (그림 4)의 과정을 거쳐 (그림 5), (그림 6), (그림 7)의 결과를 얻었다.

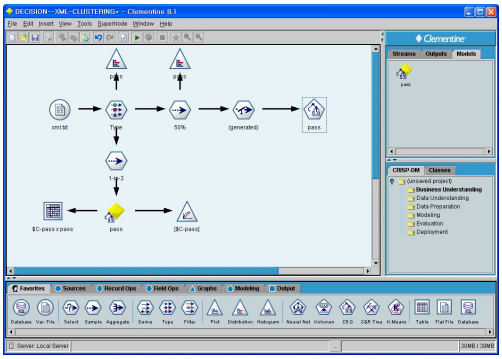

그림 4. SPSS를 이용한 실험과정
Fig. 4 Experiment process that use SPSS

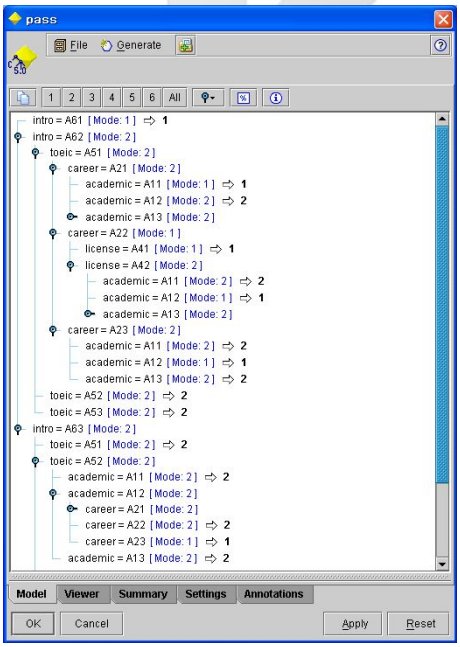

그림 5. 실험결과 생성된 Rule
Fig. 5 Rule created according to experiment result

(그림 5)는 입사 지원 분류에 가장 중요한 분류기준이 변수 intro임을 알 수 있다. 즉, 이력서에 포함한 자기소개서를 어떻게 작성하느냐가 중요한 요소임을 알 수 있다.

```
<?xml version="1.0" encoding="UTF-8"?>
<PMML version="2.1" xmlns="http://www.dmg.org/PMML-2_1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.dmg.org/PMML-2_1 pmml-2-1.xsd">
<Header copyright="Copyright (c) 2002 Integral Solutions Ltd., All Rights
Reserved.">
<Application name="Clementine" version="8.1"/>
</Header>
<TreeModel modelName="pass" functionName="classification"
algorithmName="C5">
<MiningSchema>
<MiningField name="intro" usageType="active"/>
<MiningField name="toeic" usageType="active"/>
<MiningField name="career" usageType="active"/>
<MiningField name="academic" usageType="active"/>
<MiningField name="license" usageType="active"/>
<MiningField name="talent" usageType="active"/>
<MiningField name="$C-pass" usageType="predicted"/>
<MiningField name="$CC-pass" usageType="supplementary"/>
</MiningSchema>
<Node score="2" recordCount="743">
<Extension name="x-nodeId" value="0" extender="spss"/>
<True/>
<ScoreDistribution value="2" recordCount="380"/>
<ScoreDistribution value="1" recordCount="363"/>
<Node score="1" recordCount="183">
<Extension name="x-nodeId" value="1" extender="spss"/>
<SimplePredicate field="intro" operator="equal" value="A61"/>
<ScoreDistribution value="2" recordCount="0"/>
<ScoreDistribution value="1" recordCount="183"/>
</Node>
<Node score="2" recordCount="282">
<Extension name="x-nodeId" value="2" extender="spss"/>
<SimplePredicate field="intro" operator="equal" value="A62"/>
<ScoreDistribution value="2" recordCount="190"/>
<ScoreDistribution value="1" recordCount="92"/>
<Node score="1" recordCount="8">
</Node>
</TreeModel>
</PMML>
```

그림 6 결정트리의 규칙 PMML
Fig. 6 Rule PMML of decision tree

(그림 6)은 PMML(Predictive Model Markup Language)[14]이다. PMML은 통계 모델과 데이터 마이닝 모델을 기술하기 위한 XML 언어이다.

PMML은 재정, 경제, 마케팅, 제조, 방위분야의 어플리케이션을 포함한 다양한 분야를 위해 사용된다.

(그림 6)은 이력서 분류를 위한 모델링을 XML의 응용인 PMML로 표현한 것으로, 데이터마이닝에 사용된 변수

에 대한 정보와 이 변수에 의해서 분류된 데이터의 개수 등의 SCORING 정보를 나타내고 있다.

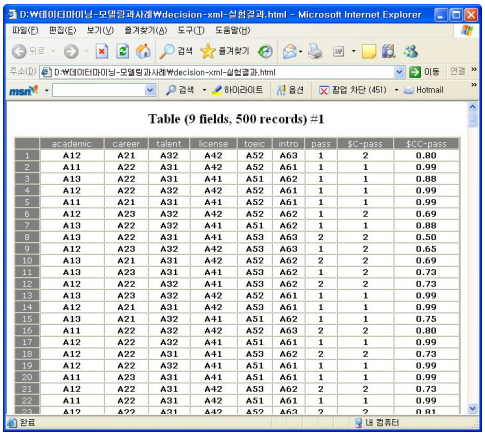

그림 7 결정트리 알고리즘의 분석결과
Fig. 7 Analysis result of decision tree algorithm

(그림 7)은 분석 결과 데이터를 웹 브라우저로 나타낸 것이다. 이 그림에서는 사용자가 잘못 예측하여 합격 판정한 것을 볼 수 있다. 그러나 결정 트리 알고리즘이 다시 정확하게 판단함을 알 수 있다.

VI. 결론

최근에 인터넷 문서의 표준문서인 XML 문서가 대량으로 생성되고 있고 이러한 문서 중에서 관심 있는 문서의 분류는 쉽지 않다 이렇게 많은 문서가 내가 원하는 분류체계를 갖고 있다면 문서의 검색과 이용에 큰 도움이 될 것으로 판단하여 제안한 알고리즘을 적용하여 실험하였다.

실험에 사용한 데이터는 이력서 데이터이고 문서의 개수는 1000개를 시험에 이용하였다. 기존의 텍스트 마이닝을 이용한 방법에서는 약 20% 정도의 XML 문서 분류 오차가 있었다. 그러나 본 논문에서 신경망과 C4.5를 이용한 시험을 이용할 경우에 문서 미분류 오차는 5% 정도로 줄었다.

데이터마이닝과 신경망을 이용한 문서분류 기술이 전통적인 문서 분류의 결과보다 더욱 효율적인 기술임을 확인할 수 있었다. 따라서, 대용량의 XML 문서 분류기술로 본 논문에서 제안한 방법이 효과적일 것으로 기대한다. 실험결과 문서의 구조를 이용한 분류방법은 XML 문서에서 유용한 분류 기준이 될 것으로 판단한다. 향후 제안한방법이 상업적으로 이용할 수 있도록 알고리즘과 편리한 인터페이스를 개발하는 것은 향후 연구과제로 남겨둔다.

참고문헌

[1] W3C, Extensible Markup Language(XML) 1.1, http://www.w3.org/
W3C Working Draft, April, 2002. (xml)

[2] P. Willet, Recent Trends in Hierarchical Document Clustering : a Critical Review, Information Processing and Management, 24:577-97, 1988.

[3] D. Hill, A Vector Clustering Technique, Mechanised Information Storage, Retrieval and Dissemination, North Holland, Amsterdam, 1968.

[4] J. Pei, J. Han, B. M. Asi, H. Pinto, "PrefixSpan : Mining Sequential Pattern Efficiently by Prefix-Projected Pattern Growth," Int, Conf, Data Engineering (ICDE), 2001.

[5] C. Chan and Y. Ioannidis, Bitmap Index Design and Evaluation, Proc. of Int'1 ACM SIGMOD Conference, 1998, 355-366

[6] P. O'Neil and S. Quass, Improved Query Performance with Variant Indexes, Proc. of Int'1 ACM SIGMOD Conference, 1997, 38-49.

[7] M. Wu, Query Optimization for Selections using Bitmaps, Proc. Int'1 ACM SIGMOD Conference, 1999, 227-238.

[8] Minos N. Garofalakis, Aristides Gionis, Rajeev Rastogi, S. Seshadri, and Kyuseok Shim. XTRACT: A System for Extracting Document Type Descriptors from XML Documents. In Proc. ACM SIGMOD, Dallas, Texas, USA, pages 165-176. ACM, 2000.

[9] 이상원, “학습하는 기계 신경망”, Ohm사, p.412, 1995.
[10] 장남식, 홍성완, 장재호, “데이터 마이닝”, 대청, p202, 1999
[11] 허명회, 이용구, 데이터마이닝 모델링과 사례, 아카데미 출판사, 2003.7.
[12] Tian Zhang, Rahu Ramakrishnan, and Riron, "Data Mining and Knowledge Discovery," p141-182, 1997.
[13] Tom M.Mitchell, MC Graw Hill "MACCHINE LEARNING," p414, 1997.
[14] PMML, http://www.dmg.org/
[15] IBM Text Mining, http://www-4.ibm.com/software/data/iminer/fortext/download/whiteweb.ht ml
[16] SEMIO Text Mining White Paper, http://www.semio.com/products/whitepapers_list.html
[17] Megaputer Text Mining White Paper, http://www.megaputer.com/tech
[18] SPSS, Clementine, http://www.spass.com
[19] 오승준, “범주형 시퀀스 데이터의 K-Nearest Neighbor 알고리즘”, 한국 컴퓨터정보학회 논문지 10권 2호, 2005.5
[20] 오승준, 원민관, “텍스트마이닝 기법을 이용한 컴퓨터 네트워커의 침입 탐지”, 한국컴퓨터정보학회 논문지 10권 5호, 2005.11.
[21] 김천식, XML 데이터 관리기술, 한국학술정보(주), 2005.12

저 자 소 개

김천식

1995년 안양대학교 전자계산학과 (공학사)
1997년 한국외국어대학교 컴퓨터 및 정보통신공학과 (공학석사)
2003년 한국외국어대학교 컴퓨터 및 정보통신공학과 (공학박사)
2000년~2003년 경동대학교 정보통신공학부 교수
2004년~현재 안양대학교 디지털미디어학부 교수
<관심분야> 데이터베이스, 데이터마이닝, 유비쿼터스, 텔리매틱스, TPEG, DMB, 홈네트워크

홍유식

1984년 경희대학교 전자공학과 (학사)
1989년 뉴욕공과대학교 전산학과 (석사)
1997년 경희대학교 전자공학과 (박사)
1985년-1987년 대한항공(NY.지점 근무)
1989년-1990년 삼성전자 종합기술원 연구원
1991년-현재 상지대학교 컴퓨터정보공학부 교수
<관심분야> 퍼지시스템, 전문가시스템, 신경망, 교통제어