다중 구조적응 자기구성지도의 퍼지결합을 이용한 웹 문서 분류

김경중, 조성배
연세대학교 컴퓨터과학과
uribyul@candy.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

Web Documents Classification with Fuzzy Integration of Multiple Structure-Adaptive Self-Organizing Maps

Kyung-Joong Kim, Sung-Bae Cho
Dept. of Computer Science, Yonsei University

요 약

웹 문서를 분류하는 목적은 특정 주제별로 중요한 문서들을 구분하려는 것과 사용자의 선호도를 바탕으로 개인화를 하려는 것으로 나누어 볼 수 있다. 특히, 웹의 효율적인 탐색을 위해 사용자가 관심 있어 할 웹 문서를 분류하는 것은 중요하다. 일반적으로 하나의 웹 문서는 특징 추출방법에 의해 문서 벡터로 표시되며 사용자의 선호여부나 주제번호를 클래스로 삼는다. 사용자가 선호도를 표시한 웹 문서를 사용하여 새로운 웹 문서의 선호 여부를 예측하기 위해 자기 구성지도(SOM)를 사용하면, 시각적으로 구조를 보여주어 데이터 사이의 관계를 효과적으로 이해할 수 있다. 그러나 SOM은 노드의 개수와 구조를 자동적으로 결정하지 못하는 단점이 있기 때문에, SOM의 장점을 활용하면서 자동적으로 구조를 결정하기 위해 구조적응 자기구성지도(SASOM)를 이용한다. 보다 나은 성능과 다양한 해석을 위해, 여러 개의 SASOM을 서로 다른 특징추출 방법을 이용하여 학습시킨 후 사용자가 주관적으로 분류기의 중요도를 결정할 수 있는 퍼지적분을 사용하여 결합하였다. UCI Syskill & Webert 데이터에 대한 실험결과 기존의 DT, MLP, naive Bayes 분류기 보다 향상된 성능을 보였다.

1. 서 론

웹 문서의 자동분류는 Yahoo와 같은 디렉토리 서비스를 자동으로 구축하는 등의 목적으로 유용하게 사용될 수 있다. 웹 문서의 양이 사람의 수작업을 통해 처리될 수 있는 한계를 넘어서면서 다양한 기계학습 기법들이 유용한 대안으로 등장하였다. 특히, 개인화를 위해서 웹 문서 분류가 응용될 수 있는데, 사용자가 선호하는 웹 문서를 분류기로 학습한 후 새로운 웹 문서의 선호여부를 예측하면 된다.

자기구성지도(SOM)는 고차원의 데이터를 시각화하여 지식을 추출하기 위한 매우 유용한 신경망이며 데이터를 클러스터링하는 효과적인 도구이다. 다른 신경망과 유사하게 SOM의 단점은 신경망의 크기와 구조를 결정하기 어렵다는 점이다. 이러한 문제를 해결하기 위해 이전연구에서 SOM의 노드를 동적으로 분할하는 효율적인 패턴 인식기를 제안하였다[1]. 구조적응 자기구성지도(SASOM)는 하나 이상의 클래스를 포함하는 노드를 여러 개의 자식노드로 분할한다.

한편 하나의 분류기보다는 여러 개의 서로 다른 분류기를 결합해서 사용하는 것이 더 좋은 성능을 얻을 수 있다. 이때 각 분류기를 서로 다르게 만드는 것과 결과를 효과적으로 결합하는 것이 중요한 문제이다. 웹 문서는 보통 수천~수만 개의 단어를 포함하고 있으며 분류기는 이것을 특징으로 사용한다. 정보이득, TFIDF, Odds ratio의 세 가지 서로 다른 특징추출 방법을 사용하면 서로 다른 분류기를 위한 학습 데이터를 생성할 수 있다.

일반적인 분류기 결과의 결합방법은 객관적인 근거를 바탕으로 한다. 이러한 접근 방법은 분류기를 설계하는 사람의 의견이 반영될 수 있는 부분이 적기 때문에 한계가 있다. 퍼지적분은 사용자가 주관적으로 평가한 분류기의 중요성과 각 분류기의 클래스에 대한 신뢰정도를 통합하여 결과를 낸다. 그림 1은 제안하는 웹 문서 분류기의 구조이다.

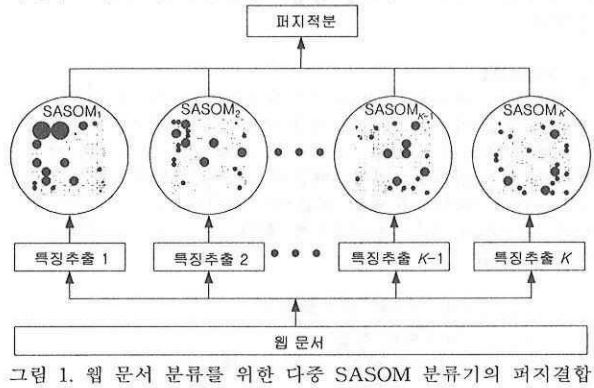


그림 1. 웹 문서 분류를 위한 다중 SASOM 분류기의 퍼지결합

제안하는 방법의 유용성을 평가하기 위해 UCI Syskill & Webert 데이터를 사용한다[2]. 사용자는 Syskill & Webert 시스템을 사용하여 웹 문서에 대한 선호도("Hot","Cold")를 표시한다.

2. 다중 SASOM의 퍼지결합

2.1 특징 추출

특징추출은 빈도수나 의존도와 같은 정보를 이용하여 특징의 순위를 매기는 과정이다. 텍스트 분류에서는 하나의 단어가 특

징에 해당하며 문서에 존재하는지 여부를 바탕으로 0 또는 1의 값을 가진다. 일반적으로 20개의 웹 문서에 5000에서 6000개 정도의 특징이 있기 때문에 특징 추출과정이 필요하다. 모든 특징을 사용하는 것은 분류기의 학습 시간을 늘릴 뿐만 아니라 성능도 저하시킨다. 정보이득, TFIDF, Odds ratio의 세가지 서로 다른 특징 추출 방법이 사용되었다.

정보이득은 정보이론에 기초한 방법이다. 여기에서 S는 웹 문서의 집합이며 E는 기대 정보이득이다. $E(W,S)$는 단어 W의 문서집합 S에 대한 기대값이다.

$$E(W,S) = I(S) - P(W=present)I(S_{w=present}) + P(W=absent)I(S_{w=absent})$$

$$I(S) = \sum_{c \in \{hot, cold\}} -p(S_c)log_2(p(S_c))$$

TFIDF는 텍스트 분류에서 자주 사용되는 특징추출방법이다. (TF=Term Frequency, DF=Document Frequency)

$$TFIDF = TF \times log\frac{1}{DF}$$

Odds ratio는 하나의 클래스에만 유용한 특징을 추출하는 방법이다.

$$OddsRatio(F) = log\frac{odds(W=present|\ C_1)}{odds(w=present|\ C_2)}$$

Odds(X_i)의 계산은 아래와 같다.

$P(X_i)=0$	$P(X_i)=1$	$P(X_i)\neq 0, P(X_i)\neq 1$
$\frac{\frac{1}{n^2}}{1-\frac{1}{n^2}}$	$\frac{1-\frac{1}{n^2}}{\frac{1}{n^2}}$	$\frac{P(X_i)}{1-P(X_i)}$

웹 문서로부터 '<' 와 '>' 같은 알파벳이나 숫자가 아닌 경우에 해당하는 것을 제거한다. 위의 세 가지 서로 다른 특징 추출 방법을 사용하여 중요한 특징만을 선택한 후 문서벡터를 구성한다.

2.2 SASOM

SOM은 고차원의 공간을 시각화하는데 자주 사용되며 위상보존 특성을 가지고 있는 신경망 모델이다. 기본적으로 SOM은 구조를 고정하고 하나의 노드에 하나 이상의 클래스로 구성된 데이터를 포함하고 있기 때문에 분류 성능이 높지 않다. SASOM은 이러한 문제를 해결하여 하나의 노드에 모인 데이터의 클래스가 모두 동일하도록 한다. 알고리즘은 구체적으로 아래와 같다.

① 지도를 4×4 크기로 초기화 한다.
② SOM 알고리즘으로 학습시킨다.
③ 지도의 노드들 중 여러 클래스의 데이터가 섞인 노드를 찾는다.
④ 찾아낸 노드들을 2×2 크기의 노드로 분화시킨다.
⑤ 분화된 노드들을 LVQ 알고리즘으로 학습시킨다.
⑥ 분화된 노드들 중, 학습에 참여하지 않는 노드를 삭제한다.
⑦ ③~⑥의 과정을 종료 조건이 만족될 때까지 반복한다.

여기서 두 가지 학습이 필요한데, 첫 번째는 일반적인 SOM 알고리즘을 이용하여 학습하는 것이고, 두 번째는 교사 학습 방법을 혼합한 LVQ방식의 학습이다. 종료조건은 모든 노드가 각각 하나의 유일한 클래스만을 표시하는 것이다. 분화된 노드의 가중치는 다음과 같이 결정한다. C는 분화된 자식 노드의 가중치이며 P는 분화되기 전 노드의 가중치이다. N_c는 자식노드의 이웃노드들의 가중치이다. S는 N_c+2이다. 그림 2는 SASOM의 노드 분화과정을 보여준다. 그림 3에서 P_4 노드는 C_0에서 C_3 노드로 분화하였다.

$$C = \frac{(P\times 2) + \sum N_c}{S}$$

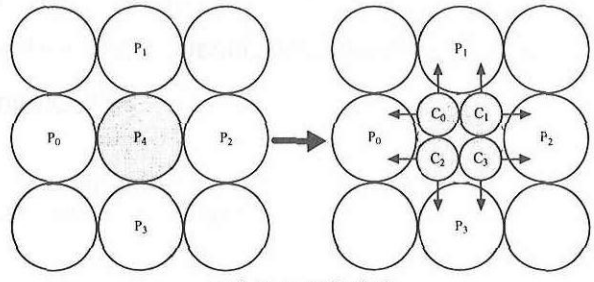


그림 2. 노드의 분할

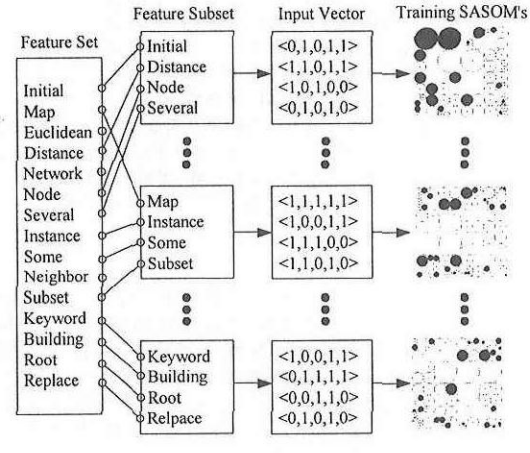


그림 3. 서로 다른 특징 집합을 사용한 SASOM 학습의 예

그림 3은 서로 다른 특징 추출 방법을 사용하여 SASOM을 학습시키는 과정을 예를 들어 보여준다. 특징 집합에는 15개의 특징이 있으며 이중에서 가장 중요하다고 생각된 4개의 특징을 뽑는다. 특징 추출 방법에 따라 선택된 특징이 달라진다. 각 특징의 존재 유무와 사용자가 문서에 대해 표시한 선호도를 바탕으로 입력 벡터를 구성한다. 입력벡터는 SASOM을 학습하는데 사용되며 각 SASOM은 서로 다른 위상구조를 가진다.

2.3 퍼지적분

다중 분류기의 결정을 위해 많은 방법이 사용된다. 일반적으로 분류기의 중요도는 모두 동일하다고 가정하거나 객관적인 방법으로 설정한다. 반면 퍼지적분은 사용자의 주관적인 평가값을 바탕으로 결합을 수행한다. 이 방법은 사용자에 의해 주관적으로 정의된 분류기의 중요도와 각 클래스에 대한 분류기의 신뢰정도를 혼합하여 결과를 낸다.

퍼지적분은 다음과 같이 정의된다. 퍼지기준은 X의 부분집합에 대해 0과 1사이의 값을 할당한다. X를 유한집합이라 할때, $h: X \rightarrow [0,1]$은 X의 퍼지 부분집합이며 함수 h에 대한 X의 퍼

지적분은 퍼지기준 g에 대해서 다음과 같이 정의된다.

$$h(x) \circ g(\cdot) = \max_{E \subseteq X}\left[\min\left(\min_{x \in E} h(x), g(E)\right)\right]$$

$Y = \{y_1, y_2, \ldots, y_n\}$ 이라고 할때, $h: Y \rightarrow [0,1]$은 함수이다. $h(y_1) \geq h(y_2) \geq h(y_3) \geq \ldots \geq h(y_n)$이라고 가정할 때 Y에 대한 퍼지기준 g에 대한 퍼지적분 e는 다음과 같다.

$$e = \max_{i=1}^{n}[\min(h(y_i), g(A_i))]$$

$A_i = \{y_1, y_2, \ldots, y_i\}$라고 할때, λ는 다음과 같은 수식으로 계산된다.

$$\lambda + 1 = \prod_{i=1}^{n}(1 + \lambda g^i) \quad \lambda \in (-1, +\infty) \text{ and } \lambda \neq 0.$$

λ는 두 부분집합의 합집합에 대한 퍼지기준 값을 계산할때 사용된다. 여기서 y_i는 분류기에 해당하고 $h_k(y_i)$는 k 클래스에 대한 분류기 y_i의 신뢰도값이다. 모든 클래스에 대해 퍼지적분 값을 계산하여 가장 높은 값을 가지는 클래스로 결정한다.

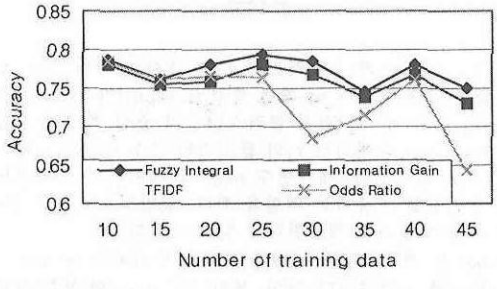
```
<A NAME="EL_SOB"></A>
<TITLE>EL SOB</TITLE>
<CENTER>
<H1>
EL SOB
</H1>
<A HREF="/IUMA-2.0/ftp/volume2/EL_SOB/EL_SOB.jpg">
<IMG WIDTH=101 HEIGHT=124 BORDER=2 SRC="/IUMA-2.0/ftp/volume2/EL_SOB/sm-EL_SOB.gif"></A>
<P><BR></P>
</CENTER>
<CENTER>
<I><FONT SIZE=5>Skin a Cat</FONT></I><BR>
```

Excerpt of HTML text (File name "1")

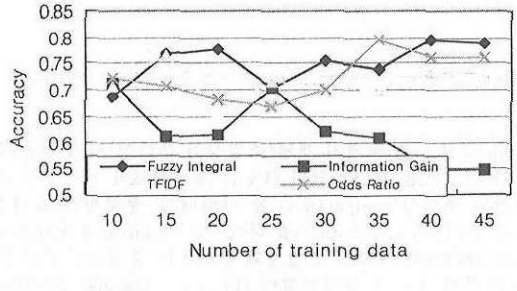
```
1|cold|http://www.iuma.com/IUMA-2.0/ftp/volume2/EL_SOB/|Fri Oct 13 15:21:56 PDT 1995|EL SOB
2|hot|http://www.iuma.com/IUMA-2.0/ftp/volume3/Lead_Pipe_Cinch/|Tue Oct 17 09:01:56 PDT 1995|Lead Pipe Cinch
3|hot|http://www.iuma.com/IUMA-2.0/ftp/volume2/Porter,_JL/|Tue Oct 17 09:05:01 PDT 1995|Porter, JL
4|cold|http://www.iuma.com/IUMA-2.0/ftp/volume3/Dr._Octojoculus/|Tue Oct 17 09:11:23 PDT 1995|Dr. Octojoculus
5|cold|http://www.iuma.com/IUMA-2.0/ftp/volume7/Adam_Bomb/|Tue Oct 17 09:12:24 PDT 1995|Adam Bomb
6|cold|http://www.iuma.com/IUMA-2.0/ftp/volume1/Russlee/|Tue Oct 17 09:15:45 PDT 1995|Russlee
```

Syskill & Webert ratings

그림 4. Syskill & Webert 데이터



(ㄱ) Bands



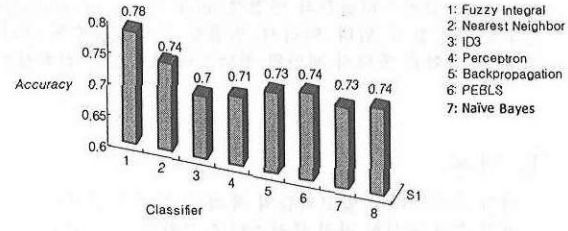
(ㄴ) Goats

그림 5. 단일 분류기와 퍼지적분 결합 모델의 성능비교

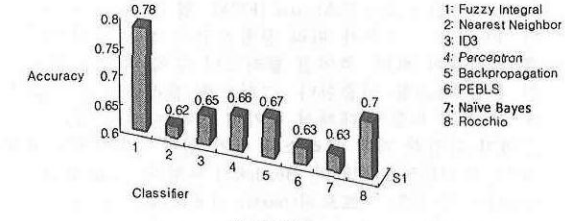
3. 실험 및 결과

UCI KDD 데이터베이스의 Syskill & Webert 데이터는 네 가지 서로 다른 주제에 대해 구성되어져 있으며 이 중 "Goats"와 "Bands" 데이터에 대해 실험을 수행한다. "Goats"는 염소에 관한 70개의 HTML 문서와 사용자의 평가가 포함되어 있으며, "Bands"는 음악밴드에 관한 61개의 HTML 문서와 사용자의 평가가 포함되어 있다. 그림 4는 데이터의 예를 보여준다. 윗부분은 실제 HTML 파일이고 아래 부분은 HTML 파일이름, 사용자의 선호도, 시스템 정보가 기록되어 있다. 초기 전처리 과정을 통해 빈도수가 높은 600개의 단어는 분류성능을 높이기 위해 제거하고, 128개의 중요한 특징을 각 특징 선택 방법을 사용하여 선택하였다[3].

그림 5와 6은 10번 반복한 실험결과이다. 그림 5는 학습 데이터의 수를 바꾸어 가면서 성능을 측정한 것이다. 특징추출방법에 따라 성능에 차이가 생겼으며 퍼지적분 결합을 통해 성능향상을 얻을 수 있었다. 데이터의 종류에 따라 좋은 성능을 내는 특징추출방법이 다르다는 것을 알 수 있다. (ㄱ)에서는 IG가, (ㄴ)에서는 TFIDF가 학습 데이터의 수에 관계없이 안정적으로 동작하였다.



(ㄱ) Bands



(ㄴ) Goats

그림 6. 다양한 분류기와의 비교 실험결과 (학습 데이터=20개)

4. 결론 및 향후연구

웹 문서는 수가 방대하고 개인의 취향에 따라 선호도가 크게 차이가 난다. 개인의 성향에 맞도록 문서를 분류하기 위해서는 개인의 주관적인 부분이 반영되어야 하며 높은 성능을 얻을 수 있어야 한다. 다양한 특징 추출방법과 개선된 SASOM 분류기를 퍼지적분으로 결합하여 향상된 성능을 얻을 수 있었다. 향후연구는 각 SASOM의 맵의 구조를 분석하는 것이다.

감사의 글

이 연구는 2002년 중소기업 기술혁신 개발사업의 위탁연구개발비에 의해 지원되었음.

참고문헌

[1] S.-B. Cho, "Self-organizing map with dynamical node splitting: Application to handwritten digit recognition," *Neural Computation*, vol. 9, no. 6, pp. 1343-1353, 1997.

[2] S. Hettich and S. D. Bay, The UCI KDD Archive, http://kdd.ics.uci.edu.

[3] M. Pazzani and D. Billsus, "Learning and revising user profiles: The identification of interesting web sites," *Machine Learning*, vol. 27, pp. 313-331, 1997.