제36회 한국정보처리학회 추계학술발표대회 논문집 제18권 제2호 (2011. 11)

반전역(Semi-Global) 문자 정열을 이용한 비속어 수집 기법

김성환, 조환규
부산대학교 컴퓨터공학과
e-mail: {sunghwan, hgcho}@pusan.ac.kr

A Collecting Method for Swear Words Using Semi-Global Text Alignment

Sung-Hwan Kim, Hwan-Gue Cho
Dept of Computer Science, Pusan National University

요 약

단어 필터링 기법의 개발에 있어서 가장 큰 난제는 정상단어를 금지어로 인식하여 이를 차단하는 오탐지 문제이다. 이를 방지하기 위하여 다양한 문장에 대한 면밀하고 광범위한 검사가 필수적이나 일반적으로 쉽게 접근할 수 있는 데이터는 주로 단어 위주로 구성된 검증 데이터로 문장 또는 구절로 구성된 데이터의 양은 실제 검증에 활용하기에는 충분하지 못하다. 본 논문에서는 웹에서 수집한 방대한 양의 말뭉치 데이터에 반전역정열(Semi-Global Alignment)을 적용하여 주어진 금지어가 사용되거나 금지어와 유사한 단어가 존재하는 구간을 탐색함으로써 단어 필터링 시스템에서 범용적으로 사용될 수 있는 문장형 평가 데이터를 수집하는 시스템을 제안하고 해당 기법을 통하여 수집한 문장 단위 데이터를 분석하고 단어 필터링 시스템이 오탐지 방지를 위해 가져야할 요소들에 대하여 검토해 본다.

1. 서론

단어 필터링은 불건전한 정보의 유통이나 사용자 간의 무분별한 욕설로 인한 분쟁을 막기 위한 가장 기본적인 방법이다. 인터넷 커뮤니티 게시판, 온라인 게임, 메신저 등 사이버 공간 상에서 의사소통 기능을 갖춘 많은 서비스들이 기본적인 단어 필터링 기능을 탑재하고 있다.

단어 필터링 시스템 개발이 어려운 이유는 금지어로 등록된 단어들을 차단하는 것은 물론 사용자간의 정상적인 의사소통을 방해하지 않도록 해야 하기 때문이다. 특히 오탐지 문제는 필터링 시스템에 대한 사용자 만족도에 크게 영향을 미치는 문제이며, 정상적인 대화를 차단하는 비율이 약간만 높아지더라도 사용자들이 큰 불편을 느껴 실제 시스템에 적용하는 것이 불가능해지게 된다.

스팸메일이나 유해 문서와는 달리 온라인 게임 채팅이나 웹 커뮤니티의 댓글은 유해성을 검증하기에 충분한 길이를 가지고 있지 않기 때문에 문장 내에 금지어가 존재하는지를 이용해 차단 여부를 판단한다. 그러나 인터넷 상의 다양한 변칙 단어 형태에 따른 금지어 목록에 대한 데이터 수집 및 배포[1]에 비하여 실제 사용자 간의 대화 내용을 이용하여 필터링 시스템의 성능을 측정하기 위한 데이터에 대한 수집이나 배포는 미비한 편이다. 기존의 단어 필터링 시스템은 주로 해당 시스템이 정상 단어 사전에 있는 단어를 필터링하는지 여부를 이용하여 오탐지에 대한 성능을 측정해왔다[2]. 그러나 이러한 단어 위주의 검증은 문장 단위의 필터링에서 성능 하락을 야기한다.

<표 1> 단어 필터링 오탐지 사례. 단어나 조사의 조합으로 우연히 금지어가 나타나 필터링 될 수 있다.

금지어	오탐지 사례
음부	처**음부**터 다시 해
시발	내가내일다**시발**표할것같아
게자식	그**게자식**사랑인거야
염병	화**염병** 날아다니고 난리 났던데

표 1은 단어 필터링으로 인한 오탐지의 전형적인 사례를 보여준다. 이러한 사례들은 해당 금지어만으로는 유추하기가 힘들며, 기존의 정상 단어 사전을 이용한 성능 측정 방법으로는 “화염병”의 경우 외에는 성능 평가에 반영되지 않는 문제점이 있다. 따라서 실제 사용자간의 대화 내용을 통하여 검증하여야 하나, 실제 인터넷 서비스 상의 사용자간 대화 데이터를 입수한다고 하더라도 방대한 양의 데이터를 일일이 분류하는 것은 사실상 불가능하다. 따라서 필터링 시스템에 의하여 탐지될 것으로 추측되는 구간에 대한 후보군 추출을 통해 해당 구간이 정상 대화인지 필터링 대상인지에 대한 분류를 보다 쉽고 빠르게 수행할 수 있도록 하는 시스템이 필요하다.

본 논문에서는 웹에서 수집한 말뭉치 및 사용자 대화 기록과 금지어 간에 반전역정열을 수행하여 주어진 금지어가 사용되는 구간 및 금지어와 유사한 구간을 추출함으로써 문장 평가 데이터 집합의 구축에 용이한 수집 기법을 제안하고 제안 기법을 통하여 문장형 평가 데이터를 수집 및 검토한다.

2. 반전역정열을 이용한 후보군 추출

서열정열(Sequence Alignment)은 본래 생물정보학에서 염기서열 간의 유사도를 판별하기 위해 사용하는 기법으로 각 문자열에 적당한 갭(gap)을 삽입하여 주어진 조건에 맞도록 두 문자열을 정열하는 것을 말한다. 서열정열 기법은 유전자 염기서열 뿐만 아니라 일반적인 문자열 간의 유사도 판별에도 응용할 수 있다. 그 중 반전역정열(Semi-Global Alignment)[4]은 두 문자열 간에 겹치는(overlap) 구간을 찾아내기 위해 사용되는 방법으로 특히 한 문자열이 다른 문자열에 비하여 길이가 짧은 경우에는 짧은 문자열과 유사한 구간들이 긴 문자열의 어느 위치에 존재하는지 알아낼 수 있다.

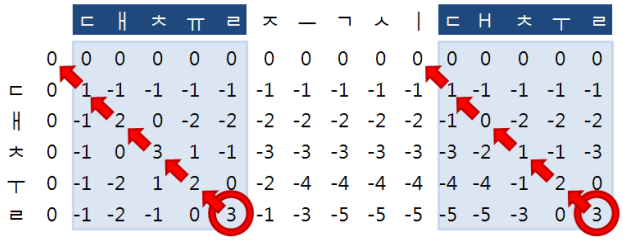
길이 m, n인 두 문자열 $\langle \alpha_i \rangle$, $\langle \beta_j \rangle$에 대하여 반전역정열을 수행한다는 것은 크기 $(m+1)\times(n+1)$인 2차원 배열 M의 성분을 채워나가는 문제로 설명할 수 있다. 배열 M의 각 성분은 아래와 같이 재귀적으로 정의된다. 이 때, $\sigma(\alpha_i,\beta_j)$는 α_i, β_j 간의 유사도 점수를 의미한다. "-" 표기는 갭(gap)을 의미한다.

$$M(0,j) = M(i,0) = 0$$

$$M(i,j) = \max\begin{Bmatrix} M(i-1,j-1)+\sigma(\alpha_i,\beta_j) \\ M(i,j-1)+\sigma(\alpha_i,-) \\ M(i-1,j)+\sigma(-,\beta_j) \end{Bmatrix}$$

배열 M의 모든 성분에 대한 값을 구한 다음에는 배열의 가장 마지막 행과 가장 마지막 열에 있는 성분 중 최댓값을 찾아 해당 값이 구해진 방향을 배열 상에서 역추적해가면 두 문자열이 서로 겹치는 부분을 알아낼 수 있다. 본 논문에서는 이 방법을 약간 변형하여 특정한 임계값 θ이상인 모든 성분에 대하여 역추적을 함으로써 긴 문장과 짧은 단어 간에 겹치는 모든 부분을 추출해 내도록 한다.

한글은 특성상 하나의 문자가 초중종성으로 구성되어 있고 이들 음소에 쉽게 변형이 일어나기 때문에 문자열을 구성하는 최소 단위를 음절이라 두면 문자열 간의 유사도 결과가 직관과 일치하지 않는 결과가 발생한다. 예를 들어 "컴퓨터"와 "콤푸타"는 서로 매우 유사하지만 대응되는 모든 음절이 서로 다르기 때문에 두 단어는 완전히 다른 단어가 되어 버린다. 이 문제를 해결하기 위하여 초중종성을 차례로 나열하는 방법[3]과 음소와 음절 단위의 치환연산으로 구분하는 방법[5]등이 있는데 본 논문에서는 초중종성을 차례로 나열하는 방법을 사용하도록 한다. 즉 "컴퓨터"와 "콤푸타"는 각각 "ㅋㅓㅁㅍㅠㅌㅓ"와 "ㅋㅗㅁㅍㅜㅌㅏ"로 표현되어 최소한 절반의 음소가 서로 일치하게 된다.



(그림 1) 두 문자열 "대출즉시ㄷH출"과 "대출"의 반전역정열 결과. 붉은 원은 마지막 행에서 θ=3을 초과한 지점을 나타낸다. 해당 지점들로부터 시작하여 점수를 얻기까지의 과정을 역추적한 결과를 붉은 화살표로 나타내었다. 반 전역정열을 통하여 확인할 수 있는 두 문자열 간에 겹치는(Overlapped) 구간을 사각형을 이용하여 표시하였다.

그림 1은 두 문자열 "대출즉시ㄷH출"과 "대출"의 반전역정열 결과를 나타낸다. 각각의 문자열의 초중종성을 분해 후 나열한 문자열 "ㄷㅐㅊㅠㄹㅈㅡㄱㅅㅣㄷHㅊㅜㄹ"과 "ㄷㅐㅊㅜㄹ"에 대하여 정열을 수행한다. 음소 간의 유사도 점수 σ의 값은 두 음소가 같을 때는 +1점, 다른 경우에는 -1점, 갭(gap)과의 유사도는 -2점, 그리고 θ=3으로 하였다. 그림 1에서 역추적 방향에 해당하는 두 문자열의 위치를 통하여 문자열 간에 겹치는 구간을 알아낼 수 있다. 겹치는 구간을 사각형으로 표현하였다. 이를 살펴보면 문자열 "대츌즉시ㄷH출"에서 주어진 단어 "대출"과 유사한 구간에 해당하는 단어는 "대츌", "ㄷH출" 임을 알 수 있다. 이 때 θ값을 적당히 조절함으로써 오차 범위를 조절할 수 있고, 또한 해당 오차 범위 내에 있는 모든 유사한 구간을 탐색할 수 있다. 후보 구간은 이와 같은 과정을 통하여 추출된 유사한 구간을 말한다.

유사도 함수 σ와 임계값 θ가 주어졌을 때 문자열 $C=\langle c_i \rangle$와 단어 $W=\langle w_i \rangle$간의 반전역정열을 통하여 추출된 C 상의 후보 구간 집합을 $S(C,W,\sigma,\theta)$라고 정의하자. 후보 구간은 문자열 C를 초중종성 분해하기 전 문자열 상에서 첫 문자의 위치를 1로 하였을 때의 상대위치로 표현하도록 한다. 예를 들어 그림 1에서는 문자열 C와 단어 W는 각각 "대츌즉시ㄷH출"과 "대출"에 대응되고 이 때 후보 구간에 해당하는 C상의 위치는 "대츌"과 "ㄷH출"로, $S(C,W,\sigma,\theta)=\{\ [3,4],\ [7,9]\ \}$이다. 분류를 위해서 $S(C,W,\sigma,\theta)$의 각 구간에 대하여 해당 구간을 포함하는 전후의 문자열이 사용자에게 주어진다.

3. 사용자 평가를 위한 시스템 구축

주어진 데이터로부터 추출된 금지어 후보 구간이 실제 필터링 되어야 할지에 대한 여부는 사용자 평가를 통하여 결정하여야 한다. 평가 시에는 전후의 문맥이나 어감이 중요하므로 $S(C, W, \sigma, \theta)$를 통해 구한 후보 단어 구간뿐만이 아니라 해당 구간을 포함한 적당한 길이의 앞뒤 문자열을 동시에 제시하여 사용자 평가를 수행하도록 한다. 사용자는 주어진 문자열에 대하여 필터링 여부를 판정하도록 한다. 주의할 점은 후보 구간과 인접한 문자열들은 후보 구간이 원래 금지어의 의미와 동일한지 여부를 판단하기 위한 것으로 인접 문자열에 속한 다른 불건전한 단어들로 인하여 필터링 여부를 결정하지 않도록 하여야 한다는 것이다. 예를 들어 금지어가 "대출"인데 주어진 문장이 "대출상담[대추]나무옆건물"이고 후보구간이 문장의 가운데 있는 "대추"라면, 이 때의 "대추"는 "대출"과는 아무런 관련이 없는 단어로 사용자는 필터링 대상이 아니라는 판정을 내리도록 하여야 한다. 이때 "대추"를 쉽게 식별하지 못한다면 전체적인 문장 의미로 인하여 필터링 대상으로 판정을 내리게 될 지도 모른다. 따라서 주어진 문자열에서 후보 구간을 식별하기 쉽도록 다른 색상으로 표기하여 해당 후보 구간 이외의 요소로 인하여 필터링 판정에 직접적인 영향을 끼치지 않도록 한다.

4. 실험 및 검토

실험을 위하여 인터넷 커뮤니티의 글 177,311개와 그에 따른 댓글 1,007,640개가 수록된 크기 200MB의 데이터에 대하여 제안 방법을 이용해 수집하는 실험을 수행하였다. 실험 환경은 Intel(R) Core(TM) i5 CPU 3.33GHz, 4GB 이었으며, 표 2에 초중종성 분해 시 길이가 4 내지 6인 대표적인 금지어에 대한 실험에 따른 추출된 후보 구간 개수 및 후보 구간 추출을 위한 수행 시간을 나타내었다.

<표 2> 대표적인 금지어에 대한 후보구간개수와 수행시간

	금지어	후보구간개수	수행시간(ms)
1	대출	2,200	29,158
2	씨발	11,733	29,447
3	지랄	4,564	29,713
4	개새끼	2,137	31,557
5	자지	11,149	26,563
6	보지	10,536	26,448
7	병신	11,679	32,408
8	야동	3,501	29,621

표 3에 실험을 통하여 발견한 오탐지 사례를 유형별로 정리하였다. 숫자로 표기는 금지어가 문장내에 존재하게 된 원인별로 분류한 것이며, 영문 알파벳은 해당 금지어 구간의 외형적인 특징에 따라 분류한 것이다. 두 분류는 직교적인 것으로 1A, 2C와 같이 병행하여 표현할 수 있다. 표 4에 대표적인 금지어와 유사한 단어들이 문장의 부분문자열로서 포함된 문장들의 일부를 유형과 함께 나타내었으며 지면 관계상 실험 결과의 일부만을 수록하였다.

수집 결과를 살펴보면 "생활정보지", "바보짓", "왁자지껄" 등 정상 단어 사전을 이용하여 사전에 성능 검사를 할 수 있는 경우도 있는 반면 두 개의 단어가 서로 결합하는 경우, 접사의 결합으로 인하여 금지어와 유사한 형태가 되는 경우, 띄어쓰기나 오타의 잘못으로 인한 경우 등 다양한 형태의 사례가 있다. 일부 문자열의 경우는 필터링 대상이 되는 불건전한 용법보다는 정상 문장에서 더욱 자주 등장하는 횟수가 많은 경우도 있었다. "질할"같은 경우에는 행위나 일을 뜻하는 접미사 "-질"과 "하다"의 의미가 결합되는 형태로 주로 사용되어 대부분의 경우에 정상 문장으로 분류되었지만, 일부의 경우에 해당 단어를 비속어로 사용한 용례도 있어 "질할" 자체를 필터링 예외 단어로 설정할 수도 없음을 확인할 수 있었다.

이러한 오탐지 사례들을 확인한 결과 주어진 단어에 대한 금지어 판단을 기본적인 원리로 채택하는 필터링 시스템은 실제 커뮤니티 게시판이나 채팅 등에 응용하기에는 오탐지로 인한 정상 대화의 방해 수준이 상당할 것으로 판단된다. 따라서 문맥 기반 및 다양한 예외 상황에 대한 규칙 추가를 통한 보다 지능적인 필터링 시스템의 개발 및 개선이 시급하다. 특히 음소 단위의 검사를 통한 필터링 방식을 사용하여 탐지 성능을 향상시키는 경우 단어 변형에는 유연하게 대응할 수 있지만 발생할 수 있는 오탐지 사례가 더욱 교묘하고 사전 탐지가 어려운 형태가 많아 세밀한 성능 검사가 요구됨을 확인할 수 있었다.

<표 3> 유형별 오탐지 사례

유형	설명
1	한 단어 내 금지어 존재
2	단어 또는 어절 간 결합
3	조사 및 접사와의 결합
A	완전 일치 또는 공백, 특수문자만 존재
B	유사한 형태의 단어
C	음소 단위 일치

<표 4> 실험을 통하여 수집된 오탐지가 예측되는 문장 데이터의 일부

	검출단어	오탐지가 예측되는 문장	유형		검출단어	오탐지가 예측되는 문장	유형
1	대출	사람들이 정말 삼류**대출**신에	2A	26	지랄	쓰는 것과 마찬가**지랄**까	3A
2	데 출	오늘 토요일인**데 출**근했음	2B	27	질할	고마워 나 열심히 삽**질할**께	3B
3	대출	도서**대출** 프로그램같은거야	1A	28	질알	뭔소스인**질알**아야	2B
4	대출	난 C 코드는 **대출** 읽을줄 알지	2A	29	지라ㄹ	이름 역시 모**질라로** 정하려다가	3C
5	때출	avr에서는 1일**때출**력이던데.	2B	30	지라 ㄹ	계속 바뀌는**지라 런**타임 중에	2C
6	개 새기	직업 참을인 한 백**개 새기**면서	2B	31	시 발	복구기능이 역**시 발**표되었습니다	2A
7	게생기	무섭**게생긴**듯	2C	32	시 바	그거**시 바**로 머신러닝	2B
8	게 새끼	저럴게 나오**게 새끼** 클래스를 완성	2B	33	시 빠	오!! 역**시 빠**른 답변	2B
9	보지	무슨 생활정**보지** 펴보면	1A	34	심받	관**심받**고 싶었성	3B
10	보/지	정**보/지**식등이 먼저 수반되어야	2A	35	시 파ㄹ	함수호출**시 파라**미터를	2C
11	보지	아직 어리고 초**보지**만 그래도	3A	36	시받	해결책을 제**시받**는 것보다	3B
12	보지	원칙을 지키는게 바**보짓**이라고	3C	37	시 빨	여학생일 **시 빨**간펜 첨삭지도 가능	2B
13	보지	진**보진**영인거랑 군대랑 무슨	2C	38	식발	정**식발**매했음	1B
14	보지	팔딱팔딱 뛰기 일**보직**전	1C	39	쉬 바ㄹ	플래**쉬 바로**가기 복사하신후에	2C
15	병신	차라리 특기**병신**청해서가는게	2A	40	시발	한국과 일본동**시발**매이고	1A
16	ㅂ영신	현역 **입영신**청을 해버리면	2C	41	시 발	바퀴를 다**시 발**명하지말라	2A
17	자지	일종의 투**자지**	3A	42	시 바	혹**시 바**로가기한거 아니냐고	2B
18	자지	사용**자지**정 형식란에	2A	43	씨 밝	흑인 아저**씨 밝**은 곡을 되게 무거운	2B
19	자지	바람쐬러 나오니 왁**자지**껄하고	1A	44	시 빠	속도는 역**시 빠**른거 같구	2B
20	자 지	스페이스 누르면 글**자 지**워지는데	2A	45	시 팔	사서 다**시 팔**고 해서 손실을	2A
21	자 지	니 혼**자 지**뢰찾기게임이나	2A	46	시받	지**시받**은것도 제대로 못하고	3B
22	자지	초보**자지**만 공부도 해볼겸	3A	47	야 동	하고싶은 분**야 동**아리가 업ㅂ자	2A
23	짜지	초**짜지**만 자주 들르겠습니다	3B	48	야 동	입력으로 해**야 동**시 입력이 가능	2A
24	자지	이스라엘의 가**자지**구 침공	2A	49	야 동	깔려있어**야 동**작하지 않을까	2A
25	자지	한**자지**원변경이 안되거든	2A	50	야도ㅇ	일부 예외의 분**야도있**다고 하심	2C

5. 결론

본 논문에서는 단어 필터링 시스템의 성능 평가를 위하여 실제 웹 커뮤니티 등지에서 사용되는 방대한 데이터로부터 필터링 후보 구간을 추출함으로써 사용자 평가를 보다 용이하게 수행할 수 있는 방법을 제안하였다. 제안하는 기법은 반전역정열을 이용하여 주어진 금지어와 유사한 구간을 추출하고 해당 후보 구간을 포함한 전후의 문자열을 함께 제시하여 사용자가 해당 구간에 속한 문자열이 실제 문장 내에서 지니는 의미를 보다 용이하게 파악할 수 있도록 하였다.

제안 방법을 이용한 실험을 통하여 여러 가지 오탐지 및 오탐지 예상 사례를 실제 커뮤니티 사이트의 사용자 대화 기록을 통하여 수집하였다. 수집 결과를 통하여 필터링 강도나 방법에 따라 발생할 수 있는 다양한 유형의 오탐지 사례를 파악할 수 있었다.

본 논문에서 제안한 기법은 하나의 금지어와 임계값 , 유사도 함수가 주어진 상태에서 문자열 상에서 후보 구간을 추출하는 것을 기본으로 하고 있기 때문에 금지어가 다수 존재하는 경우에 대한 개선이 필요하다. 또한 후보 구간 추출을 통하여 검사해야 할 분량을 감소시킬 수는 있었으나 대규모 말뭉치에 대한 수행 결과에서 여전히 수천에서 수만개의 후보 구간이 존재하여 사용자 평가에 많은 비용이 소모되었고, 따라서 사용자 친화적인 평가 환경 제공에 관한 추가적인 연구 및 개선이 필요하다.

Acknowledge

이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. 2011-0003157)

참고문헌

[1] 한국게임산업진흥원, “게임언어 건전화 지침서 연구”, 2008.

[2] 윤태진, 정우근, 조환규, “제한된 한글 입력환경을 위한 음소기반 근사 문자열 검색 시스템”, 정보과학회논문지:소프트웨어및응용, 제37권, 제10호, pp. 788-801, 2010.

[3] 윤태진, 조환규, “반 전역 정렬을 이용한 온라인 게임 변형 욕설 필터링 시스템”, 한국콘텐츠학회논문지, 제9권, 제12호, pp. 113-120, 2009.

[4] S. Coull, J. Branch, B. Szymanski and E. Breimer, "Intrusion Detection: A Bioinformatics Approach," In Proc. of 19th CSAC, pp. 24-33, 2003.

[5] 노강호, 김진욱, 김은상, 박근수, 조환규, “한글에 대한 편집 거리 문제”, 정보과학회논문지:시스템및이론, 제37권, 제2호, pp. 103-109, 2010.