인터넷 신문기사의 자체 생산량 측정 기술

김동주*, 김한우**
*안양대학교 컴퓨터공학과, **한양대학교 컴퓨터공학과
e-mail: djkim@anyang.ac.kr, kimhw@cse.hanyang.ac.kr

A Technique for Measuring the Self-Production of Internet Newspapers

Dong-Joo Kim*, Han-Woo Kim**
*Dept of Computer Engineering, Anyang University
**Dept of Computer Science & Engineering, Hanyang University

요 약

인터넷의 발달과 인터넷 문화의 보편화로 인하여 사용자들은 폭발적으로 증가하는 다양한 정보를 접할 수 있게 되었으며, 자체 생산하거나 다른 신문사들로부터 생산된 기사들을 단순 유통, 링크를 통하여 정보검색 사이트들뿐만 아니라 각종 포털 사이트, 인터넷신문사들은 다양한 경로로 기사를 제공할 수 있게 되었다. 이에 따라 인터넷신문을 규정하고 법적 테두리에 넣기 위한 법률이 제정되었으며, 인터넷신문사에 대해 기사의 자체 생산량이라는 요건 검증에 대한 요구가 증가하고 있다. 본 논문은 인터넷신문 자체기사 생산량을 측정하기 위해 필요한 기술들을 조사하고 타당성을 검토하여 이에 적합한 기술을 제시한다. 제시한 방법은 대량의 기사의 비교를 빠른 시간에 수행할 수 있도록 하기 위해 인간의 단어 인지와 관련한 경험적 정보의 반영을 통하여 변형한 편집거리 기반 방법이다. 제시하는 방법의 정확성을 검증하기 위해 실제 소량의 인터넷 신문 기사를 대상으로 실험하였다.

키워드: 인터넷신문(internet newspaper), 문서유사성(document similarity), 편집거리(edit-distance)

I. 서론

오늘날 인터넷의 발달로 인터넷 언론들의 수가 폭발적으로 증가하였으며 포털의 언론 영향력이 커지면서 사회적 책임을 묻기 위해 포털 사이트를 언론 기관으로서 법적 테두리에 묶기 위한 입법화가 이루어졌다. 그에 따라 제정된 신문법에서 여전히 논란이 되고 있는 부분은 인터넷신문에 관한 규정이고, 이 규정의 시행령에 따라 인터넷신문사는 독자 생산량이 30%를 넘어야 한다고 규정하고 있다. 이에 따라 인터넷신문사의 독자 생산량의 측정이 요구된다.

본 논문은 인터넷신문 자체기사 생산량 측정 시스템 개발을 위해 필요한 기술들을 조사하고 타당성을 검토하여 이에 적합한 기술을 제시한다. 특정 인터넷 사이트에서 제공되는 인터넷신문기사는 자체 생산되는 경우, 다른 인터넷신문사로부터 제공받아 단순 유통하는 경우, 제공받은 신문기사를 통해 재생산하는 경우, 도용하거나 표절하는 경우로 나누어 볼 수 있다. 이들 중 법률에서 규정하고 있는 인터넷신문으로서 지위를 인정받기 위해서는 유통하는 경우와 재생산하는 경우, 도용하거나 표절하는 경우를 제외한 자체 생산되는 기사의 수를 측정할 수 있어야만 한다. 즉, 다른 어느 인터넷신문사에서도 생산되지 않은 고유의 자체 기사를 확인할 수 있어야만 한다. 본 논문은 이들을 정량화하여 기사들이 서로 얼마나 유사한지를 판단하는 것을 목표로 하고 있다.

$$2\times 10{,}240\times\binom{10000}{2}=2\times 10{,}240\times\frac{10000!}{2!\times(10000-2)!} \quad (1)$$
$$=10{,}240\times 99{,}990{,}000$$
$$=1{,}023{,}897{,}600{,}000\,\mu s$$
$$\approx 284\,\text{hours}$$

이를 위해 가장 간단한 방법은 모든 신문 기사 쌍을 서로

비교하되 비교의 구체적인 대상을 단어의 표면적이 형태를 직접 비교하는 문자열 일치 방법(KMP, Boyer-Moore)을 사용하는 것이다. 그런데 이 방법을 사용할 경우 하나의 문자쌍을 비교하는데 1μ초가 걸린다고 가정하고 기사의 평균 길이가 10kbytes인 1만 개의 기사를 비교한다면 최악의 경우 계산식 (1)의 시간이 걸린다. 거의 12일이나 걸리는 이 시간도 물론 허용할만한 시간이 아니지만, 이것이 전부가 아니다. 단순 문자열 일치 알고리즘은 완전 일치 문자열을 찾아내는 것을 목표로 하는 알고리즘이기 때문에 부분적으로 유사한 재생산되는 경우나 부분 표절 같은 경우에 대해 적용하려면 문자열 중간에 임의의 길이의 불일치, 혹은 대치, 삭제, 재배열 등의 문자열을 허용해야만 한다. 이럴 경우 비교 시간은 기하급수적으로 증가할 것이다.

따라서 이와 같이 다양한 경우에 대해 적응력 있게 빠른 시간 안에 모방이나 표절을 판단하기 위해서는 고전적인 단순한 문자열 일치의 방법을 넘어서는 기법들이 요구된다. 이에 적용 가능한 몇 가지 방법들을 살펴보고, 자체기사 생산량을 측정하기 위한 방법으로서의 타당성을 조사한다. 그런 다음 각 방법에서의 몇 가지 장점을 취하여 자체기사 생산량 측정에 적합한 방법을 제시한다.

II. 문서간의 유사성 측정 방법

1.1. 정보검색 방법

기사의 독자 생산 여부를 판단하는 과정은 기준 문서를 다른 모든 문서와 일대일 비교를 수행한다는 점에서 정보검색 과정과 동일하다. 정보검색은 용어, 혹은 색인어, 키워드, 등과 같은 정보 식별자(information identifier)로 각각의 문서를 특징 지워주고, 이 특징 정보의 비교나 특징 정보들 간의 거리를 측정하여 문서들 간의 유사성을 판별한다.

정보검색 방법에서 용어의 빈도에 기반한 벡터 공간 모형[1]의 첫 번째 문제점은 충분한 문서가 이미 수집되어 있어야만 한다는 것이다. 즉, 통계량으로서 가치 있는 량의 문서가 수집되지 않는다면 벡터 공간 내에서의 문서들 간의 일대일 유사도 거리는 다른 문서와의 상대적인 관계에서 무의미해진다. 충분한 문서가 수집되어야만 여러 문서들간의 벡터 공간상의 거리가 가치 있어진다.

더욱 치명적이 문제점은 문서 내에서의 문맥을 반영하지 못한다는 것이다. 정보검색 기반 방법에서는 용어의 발생이 서로 독립적이라 가정하여 문서 내에 발생하는 용어 빈도만으로 문서를 벡터 공간 내에 사상하고 있다. 이러한 방법은 실제로 유사한 문서를 놓칠 가능성은 줄어들긴 하겠지만 본 논문에서 주요 쟁점이 유사하지 않은 문서를 구분하는 데는 매우 취약하다.

반면 정보검색 기반 방법의 장점은 표면정보에 민감하지 않다는 것이다. 이 장점은 마지막에 살펴본 독립 가정에 의한 빈도 기반 개념과 같은 것으로 장점이 되기도 하고 단점이 되기도 하는 문제이다. 즉, 정보검색 방법의 이러한 장점은 너무나 과도해 곧바로 단점이 되는 것이다. 따라서 정보검색 방법을 유사 문서 판단에 활용하기 위해서는 문맥 정보나 표면 정보에 좀 더 민감하게 변경하는 것이 바람직하다.

또 다른 장점은 비교를 점진적으로 수행할 수가 있고, 유사도 계산을 위해서 실제 비교시에 매우 빠른 속도로 수행될 수 있다는 것이다. 물론 유사도 계산을 위한 정보를 구축하는데, 즉, 색인어를 추출하여 역파일을 구성하는데 과도한 시간이 걸릴 수도 있기는 하지만 계산시 빠른 속도는 정보검색 기반 방법의 최대 장점이다.

1.2. 편집거리 방법

신문 기사를 포함한 일반 전자 문서의 유사성 검사를 위한 보다 직접적인 편집거리(edit distance)라는 방법[2-6, 8]이 존재한다. 편집거리는 비교하는 두 문자열의 유사성 정도(엄밀히는 비유사성 정도)를 문자의 표면 정보를 직접 비교하여 측정하는 방법이고, 각 응용 분야에 따라 다양한 변형 알고리즘이 존재한다.

이 편집 거리 알고리즘의 근간을 이루고 있는 개념은 만약 일련의 행위의 집합이 항목열에 의해 기호적으로 주어질 수 있다고 했을 때, 두 기호열 표현을 직접 비교에 의한 두 행위열의 비교의 필요성은 행위들의 패턴을 규명하는데 매우 유용할 수 있다는데 있다.

편집 거리 알고리즘에는 몇 가지 다른 알고리즘이 있는데, 본 절에서는 편집거리 알고리즘을 살펴보고 편집거리 알고리즘의 유사도 평가 알고리즘을 사용하여 신문 기사 내용의 독자 기사 판단 방법을 소개한다. 본 절에서 소개하는 알고리즘은 기본적으로 비교 단위가 문자인데, 물론 문자를 비교 단위로 하더라도 문서 비교 문제에 적용하는데 어려움이 없지만 자연언어 문장에서 의미의 기본 단위인 단어나 형태소를 비교 단위로 삼는 것이 직관적으로 더 가치 있을 것이다. 따라서 이 장에서 특별한 언급이 없는 경우는 문자를 단어나 형태소로 간주하고, 문자열은 문장, 혹은 문서를 의미하게 된다.

Hamming 거리

Hamming 거리척도[2]는 통신 분야에서 데이터가 전송될 때 발생하는 오류를 발견하고 수정하기 위한 알고리즘이고, 오류를 평가하는 방법으로 고정길이의 이진 코드에서 비트의 뒤바꿈 수로 판단하였다. 즉, 동일한 길이의 두 문자열 사이의 Hamming 거리는 가장 단순한 방법으로 대응하는 서로 다른 기호들의 수이다. 즉, 비교 대상의 두 문자열에 대하여 한 문자열을 다른 문자열로 변경하기 위해 필요한 대치 연산의 수이다. 가능한 대치 방법이 여러 가지가 존재하겠지만 여기서는 최소 개수를 의미한다.

Levenshtein 거리

두 문자열의 유사도를 계산하기 위한 Levenshtein 알고리즘[3]은 패턴 인식, 생물정보학, 철자오류 교정, 음성인식

등 다양한 분야에서 사용된다. 이 알고리즘은 개념적으로 한 문자열을 비교 대상이 되는 다른 문자열로 변환하는데 필요한 삽입, 삭제, 교체 연산의 최소 횟수를 계산하는 알고리즘으로 값이 크면 클수록 비유사성 정도가 커지는 문자열 비교를 위한 비유사성 척도이다.

Damerau-Levenshtein 거리

두 문자열의 비교를 위한 Damerau-Levenshtein 거리 척도[4]는 워드프로세서에서의 철자 오류 교정을 위해 제안되었다. 이 척도는 Levenshtein 거리 척도와 동일하게 비교 대상이 되는 두 문자열에 대해 하나의 문자열을 다른 문자열로 변환하는데 필요한 삽입, 삭제, 교체 연산의 최소 수를 의미한다. 그러나 Levenshtein 거리 척도와는 달리 한 문자열 내에서 이웃하는 두 문자의 교환 연산을 한 개의 연산으로 간주한다는 것이다.

서열 정렬 알고리즘

서열 정렬(sequence alignment) 알고리즘[5,6]은 생물정보학 분야에서 단백질이나 아미노 염기 서열에서 존재하는 부분 서열의 상관관계를 분석하는 방법이다. 서열 정렬의 목적은 관심 대상 서열과 상동성이 높은 서열들을 파악하여 서열의 기능을 추정하거나 관련 있는 서열들 간의 정량적 상관관계나 관련 기능 등을 예측하기 위한 것이다.

서열 정렬 알고리즘은 Levenshtein 거리 척도에서 삽입, 삭제 교체, 그리고 일치에 대한 가중치 집합을 일반화한 것이고, 엄밀한 의미에 비유사성(dissimilatiry) 척도인 Levenshtein 거리 척도와는 달리 서열 정렬 알고리즘에서 사용되는 척도는 유사성 척도이다. 따라서 Levenshtein 거리 척도에서의 각 연산에 대한 가중치는 해당 연산의 빈도수를 세는 역할만을 수행하지만 서열 정렬 알고리즘에서 삽입, 삭제, 교체 연산은 유사성 정도를 떨어뜨리는 역할을 하므로 벌점(penalty)이 주어지고, 일치에 대해서는 유사성 정도를 높이는 역할을 하므로 이점(advantage)을 주게 된다.

1.3. 점도표 방법

점도표(dotplot) 방법[7]은 원래 생물정보학학 분야에서의 연구가들이 DNA열의 자기유사성(self-similarity)를 연구하기 위한 방법을 찾는 데에서 시작된 방법으로 방대한 양의 디지털 정보에서 비교되는 두 문자열 매치의 패턴들을 시각화하기 위한 기술이다. 그런데 생물정보학 분야에서와 유사한 목적으로 일반 문서들 간에서도 유사한 서열(sequence)들을 찾는 것이 필요했고, 전체적인 유사한 부분을 한 눈으로 확인하기 위해 점도표라는 시각적인 접근을 시작하게 되었다. 문서들은 토큰화되고(즉, 띄어쓰기 단위, 혹은 형태소 단위 등으로 분리) 각 토큰의 쌍들은 쌍비교가 이루어진다. 토큰들이 일치하는 곳에 매트릭스의 원소 하나에 점을 찍거나, 혹은 유사성 정도를 나타내는 한가지의 색으로 한 점을 찍어 2차원의 점도표 공간을 구성하게 된다. 점도표의 패턴은 사각형과 대각선의 시각적 이미지를 통해 해석된다. 예를 들어, 만약 한 문서에서 세 번째 토큰이 또 다른 다섯 번째 토큰과 일치한다면 점도표 매트릭스의 (3, 5) 위치에 점 하나를 찍는 방식이다.

III. 제안하는 방법

지금까지 살펴본 여러 가지 알고리즘들과 기술들의 장단점을 토대로 구축할 수 있는 이상적인 시스템을 다음을 염두에 두어야만 한다.

○ 탐색 범위 축소, 문맥의 반영, 표면 정보의 민감성 완화, 비교 횟수 축소

탐색 범위 축소를 위해 먼저 빈도기반의 정보검색 방법론을 적용하여 비교 대상이 되는 문서의 그룹화를 수행한다. 이때, 수행하는 그룹화는 그룹이 서로 최소한 30~50% 정도는 중첩되어야만 한다. 또한 각각의 그룹은 하나의 중심점을 갖는다.

비교하고자 하는 문서의 입력이 들어오면 문서 벡터를 통하여 해당 문서와 가장 가까운 그룹의 중심점을 찾아낸다. 그 중심적이 소속되어 있는 그룹의 문서만을 대상으로 편집 거리 기반 방법을 적용하여 비교를 수행한다.

문맥정보를 반영하기 위해 연속 일치되는 부분은 길이에 비례하여 가중치를 높여주는 동적 가중치를 설정한다. 또한 표면정보의 민감성을 완화하기 위해 형태소 분석된 어간/어근의 품사열의 유사도를 통합적으로 계산한다.

마지막으로 비교 횟수를 감소시키기 위해 대명사, 부사, 기호 등을 포함하고 문서의 의미에 큰 역할을 하지 않는 어절은 비교 대상에서 제외한다. 또한 어절 내에서도 형식 형태소를 제거하고 남은 어근/어간에 대해서 첫 문자와 마지막 문자만을 비교한다.

실험 평가를 위해 구현한 소규모 시스템에서 문자열 비교는 편집거리 방법들 중 Needleman-Wunsch 알고리즘[8]에 기반한 전역적 비교 방법을 사용하였다. 표면 정보에 대한 민감성을 완화하기 위한 표제어 정보나 품사 및 의미 정보와 같은 언어적 정보를 사용하지는 않았지만 단어(어절)의 경계 문자만을 비교하였기 때문에 민감성 완화뿐만 아니라 비교의 빈도수까지도 줄여 더 빠른 속도로 동작할 수 있었다.

사용한 Needleman-Wunsch 알고리즘에서의 가중치 값들은 $w_m = 1$, $w_s = 0$, $w_d = -1$로 설정하였고, 계산 결과 $D_{m,n}$에 대하여 순위화를 위한 일반화는 다음 식에 의해 이루어졌다.

$$\mathrm{SIM}(X,Y) = \frac{D_{m,n} - w_d \times |m-n|}{w_m \times \min(m,n) - w_d \times |m-n|} \times 100(\%) \quad \cdots\cdots (2)$$

따라서 완전히 동일한 두 신문 기사 X, Y에 대한 유사도

는 100이라는 결과가, 그리고 내용이 완전히 다른 두 신문기사, 즉, 공통적으로 같이 사용하고 있는 동일한 단어가 단 한 개도 없는 두 신문 기사의 경우 0이라는 값이 나올 것이다.

IV. 성능평가 및 분석

실험을 위해 수집한 데이터는 4개의 신문사로부터 2007년 12월 13일 11시 경에 속보로 올라온 기사를 무작위로 50개씩 총 200개의 기사를 수집하였다. 표 1는 수집된 신문기사 내용 중에 있는 인용정보를 토대로 수작업으로 자체 생산량을 조사한 것이다.

표 1은 임의로 수집한 기사에 대한 통계이므로 자체 생산 기사 수와 타사 인용 기사 수에서 타사 기사 전체를 인용한 기사가 수집되지 않아서 수집된 기사 집합내에는 실제로 존재하지 않을 수도 있다. 즉 S사는 수집된 50개의 기사 중 28개의 기사를 Y사로부터 인용하였는데, 28개의 기사는 실제로 수집된 Y사의 기사 집합에 없을 수도 있다는 의미이다.

표 1. 수집한 신문기사의 수와 길이

괄호 안은 비율 (%)

		Y사	S사	J사	H사	전체 (Y사제외)	전체 (Y사포함)
자체생산 기사수		50 (100)	20 (40)	12 (24)	24 (48)	56 (37)	106 (53)
타사 인용	Y사	0	28 (56)	34 (68)	24 (48)	86 (58)	86 (43)
	E사	0	1 (2)	0	1 (2)	2 (1)	2 (1)
	N사	0	1 (2)	1 (2)	0	2 (1)	2 (1)
	P사	0	0	3 (6)	0	3 (2)	3 (2)
	K사	0	0	0	1 (2)	1 (1)	1 (0)

수집된 인터넷 신문 기사 데이터를 제안하는 시스템으로 얼마나 많은 기사가 자체 생산되지 않고 타사 기사를 인용한 것인지를 실험하였다.

먼저 Y사를 제외한 S사, J사, H사의 수집된 각각의 50개의 문서를 기준으로 다른 신문사 기사와의 유사도를 평가하였다. 표 2는 이에 대한 결과로 유사도 100%로 판정한 기사가 S사에 8개, J사에 5개, H사에 9개 있었고, 21~30%로 판정한 기사가 S사에 7개, J사에 8개, H사에 7개 있었다. 유사도 100%라고 판정한 각 신문사의 기사는 모두 예외 없이 수집된 Y사의 기사 집합 내에 존재하는 기사들이었다.

표 2. 유사도에 따른 기사 수

신문사 / 유사도 (%)	S사	J사	H사	전체
100	8	5	9	22
51~99	0	0	0	0
41~50	1	0	0	1
31~40	0	1	0	1
30이하	41	44	41	126
전체	50	50	50	150

유사도 100%라고 판정한 기사는 표 3에서 수작업으로 조사된 타사(Y사) 인용 기사수와 완전히 동일하므로 시스템의 정확률은 100%이다.

V. 결론

본 논문에서는 인터넷기사의 자체 생산량을 측정하기 위한 방법들을 조사하고 타당성을 검토하였다. 이들 방법들 중 본 논문에서는 탐색범위를 축소하기 위해 빈도를 이용하여 전혀 유사하지 않은 문서를 걸러내었다. 또한 표면 정보의 민감성을 완화하기 위해 어근/어간의 품사열 비교를 수행하였으며, 비교 횟수의 감소를 위해 문서에 큰 영향을 주지 않는 대명사, 부사, 기호의 비교를 제외하였으며, 단어의 경계문자만을 비교하였다. 이들 정보와 비교 기준을 통하여 최종적으로 비교를 수행하기 위한 알고리즘으로는 Needleman-Wunsch 알고리즘에 기반한 전역적 비교 방법을 사용하였다. 실험결과 자체생산된 기사를 엄격히 분간할 수 있었을 뿐만 아니라 부분적으로 유사한 내용의 기사도 알아낼 수 있었다.

그러나 본 논문에서 실험을 위해 사용된 기사는 주요 몇몇 신문사였으며, 특정 시간대에 게시된 매우 작은 량을 기사였을 뿐이다. 또한 추출한 기사는 무작위로 추출하였기 때문에 논문에 언급된 인터넷신문사들의 자체 생산량을 반영한다고 볼 수 없다. 따라서 향후 좀 더 광범위한 데이터 수집을 통한 장기적인 평가가 필요할 것이다.

참고문헌

[1] G. Salton, et al., "A Vector Space Model for Automatic Indexing," Communication of ACM, v.18, n.11, pp. 613-620, 1975.

[2] Richard W. Hamming, "Error Detecting and Error Correcting Codes," Bell System Technical Journal, v.26, n.2, pp.147-160, 1950.

[3] V. L. Levenstein, "Binary Codes Capable of Correcting Deletions, Insertions, and Reverables," Soviet Physics-Coklady Akademii

Nauk SSSR, v.163, n.4, pp.845-848, 1965.

[4] F. Damerau, "A Technique for Computer Detection and Correction of Spelling Errors," CACM, v.7, n.3, pp.659-664, 1964.

[5] S. Altschul, "A Protein Alignment Scoring System Sensitive at all Evolutionary Distances," Journal of Molecular Evolution, v.36, pp.290-300, 1993.

[6] T. smith and M. Waterman, "Identification of Common Molecular Subsequences," Journal of Molecular Biology, v.147, pp.195-197, 1981.

[7] K. Church and J. Helfman, "Dotplot: a Program for Exploring Self-Similarity in Millions of Lines of Text and Code," Journal of Computational and Graphical Statistics, v.2, n.2, pp.153-174, 1993.

[8] S. Needleman and C. Wunsch, "A General Method Applicable to the Search for Similarities in the Amino Acid Sequences of Two Proteins," Journal of Molecular Biology, v.48, pp.443-453, 1970.