Min-Max Hash 를 활용한 다중 집합 기반의 유사도 측정

윤진욱*, 김병욱*
*동국대학교 경주캠퍼스 컴퓨터공학과
e-mail : jinuknamja@dongguk.ac.kr, bwkim@dogguk.ac.kr

Min-Max Hash for Similarity Measurement based on Multiset

Jin-Uk Yoon*, Byoungwook Kim*
*Dept. of Computer Engineering, Dong-Guk University- Gyeongju

요 약

데이터 마이닝에서 클러스터링은 서로 유사한 특징을 갖는 데이터들을 동일한 클래스로 분류하는 방법이다. 클러스터링에는 다양한 방법이 존재하지만 대표적으로 집합으로 표현된 데이터들의 유사도를 측정하기 위해서는 자카드 유사도(Jaccard Similarity)를 이용한다. 자카드 유사도는 서로 다른 집합 간의 공통된 부분을 상대적으로 평가하여 유사도를 측정하는 방법이다. 그러나 최근에는 데이터를 저장할 수 있는 기술과 매체의 발전으로 표현할 수 있는 데이터의 영역과 범위는 발전되고 있기 때문에 많은 연산과 시간의 비용이 발생하게 된다. 이를 해결하기 위해서 두 데이터의 표본의 유사도를 통해 실제 데이터들의 유사도를 추정할 수 있는 Min-Hash 가 제안되었다. 본 논문에서는 이를 활용하여 집합의 영역을 다중 집합(Multiset)으로 확장하여 중복되는 값을 가질 수 있는 두 데이터 간의 유사도를 효율적으로 추정할 수 있는 Min-Max Hash 를 제안한다.

1. 서론

데이터 마이닝에서 클러스터링은 서로 유사한 특징을 갖는 데이터들을 동일한 클래스로 분류하는 방법이며, 이는 협업 필터링(Collaborative Filtering)의 영역에서 고객들의 데이터를 활용하여 온라인 구매, 컨텐츠 평가 등에서 활용된다. 또한 서로 다른 텍스트들의 비슷한 정도를 측정하여 표절과 같은 문제도 해결할 수 있다[1]. 현재 이와 같은 방법을 활용하는 데이터들은 집합의 형태를 기반으로 하여 표현되고 있다. 일반적으로 집합으로 표현된 데이터들의 유사도를 측정하기 위해서는 자카드 유사도(Jaccard Similarity)를 이용하여 데이터들의 공통된 부분을 상대적으로 평가하여 유사도를 측정한다. 그러나 오늘날 데이터를 저장할 수 있는 기술과 매체의 발전으로 표현할 수 있는 데이터의 영역과 범위는 점점 더 발전되고 있기 때문에 이전과 같은 방법으로 유사도를 측정하고 데이터를 분류하기에는 수 많은 연산과 시간의 비용이 발생한다. 이를 해결하기 위해서 데이터가 가지는 모든 값들을 비교하지 않고 해시 함수를 통하여 일부의 항목만 추출하여 생성된 일정한 크기의 표본들의 유사도를 측정하여 실제 데이터들의 유사도를 측정할 수 있는 Min-Hash 가 제안되었다[2].

본 논문에서는 데이터를 표현하는 집합의 영역을 다중 집합(Multiset)으로 확장하여 중복되는 값을 가질 수 있는 데이터들의 유사도를 비교해보고, [2]에서 제안된 Min-Hash 를 활용하여 다중 집합을 기반으로 하는 데이터들의 유사도를 효율적으로 추정할 수 있는 Min-Max Hash 를 제안한다.

2. 배경 지식

2.1 Jaccard Similarity

자카드 유사도(Jaccard Similarity)는 두 집합의 유사도를 측정하는 방법으로, 두 집합의 합집합의 크기에 대하여 교집합의 상대적은 크기를 계산하여 두 집합의 유사한 정도를 나타낼 수 있는 방법이다.

$$\text{Jaccard}(A,B) = \frac{|A \cap B|}{|A \cup B|}.$$

예를 들어서 두 집합 A={a,b,d}와 B={a,b,c,e}가 존재할 때, 집합 A 와 B 의 유사도는 Fig. 1 과 같다.

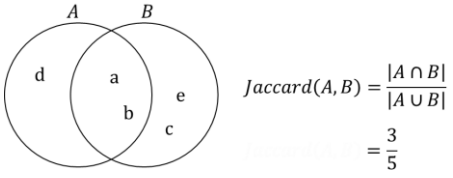


Figure 1. Jaccard Similarity

2.2 Min-Hash

Min-Hash 는 두 집합의 원소들을 특정 해시 함수에 사상했을 때 그 중 가장 작은 결과값을 나타내며 이

를 활용하여 유사도를 근사적으로 추정할 수 있는 방법이다. Min-Hash 를 통해 나온 가장 작은 결과값을 Min-Hash Value 라고 표현하며 집합 A 에 대한 Min-Hash Value 를 $h_{min}(A)$로 표현한다. 비교 대상인 두 집합 A 와 B 의 Min-Hash Value 가 서로 동일할 확률은 두 집합의 자카드 유사도와 동일하다[2].

$$Pr[\ h_{min}(A) = h_{min}(B)\ ] = Jaccard(A, B)$$

m 개의 원소로 이루어진 집합의 Min-Hash 를 구하기 위해서 대표적으로 사용하는 해시 함수 h의 형태는 $ax + b \bmod p$ 이며, a와 b는 임의의 자연수이고 p는 m 보다 크거나 같은 가장 작은 소수이다.

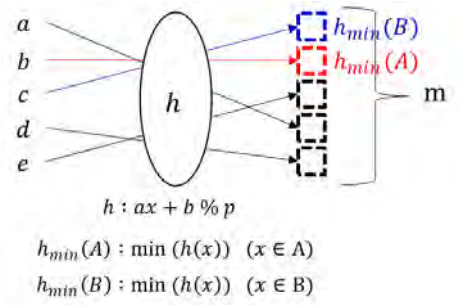


Figure 3. Min-Hash

집합 A 에 대하여 n 개의 Min-Hash 를 통하여 Min-Hash Value 를 n 개를 가질 때 이를 벡터의 형태로 나타낼 수 있으며 Min-Hash Signature 로 표현하고 다음과 같이 Sig_A로 정의한다.

$$Sig_A = [\ h_{min_1}(A), h_{min_2}(A)), \ldots h_{min_n}(A))]$$

집합 A 와 B 의 Min-Hash Signature 에서 k 번째 Min-Hash Value 를 서로 같은 값으로 가질 때 유사도의 가중치는 다음과 같이 정의한다.

$$\widehat{Jaccard}_k(A,B)\begin{cases}1 & h_{min_k}(A) = h_{min_k}(B)\\ 0 & h_{min_k}(A) \neq h_{min_k}(B)\end{cases}$$

따라서, n 개의 해시 함수를 활용하여 생성된 집합 A 와 B 의 Min-Hash Signature 의 유사도를 계산하여 집합 A 와 B 의 유사도를 근사적으로 추정할 수 있다.

$$Jaccard(A,B) \fallingdotseq \frac{1}{k}\sum_{k=1}^{n}\widehat{Jaccard}_k(A,B)$$

2.3 Multiset

다중 집합(Multiset)은 집합을 확장시킨 개념으로써, 집합과는 다르게 동일한 원소의 중복도 허용한다. 원소들의 중복된 정도를 Multiplicity 라고 하며 다음과 같이 다중 집합의 원소들의 Multiplicity 를 나타내는 함수 f와 집합 U 의 원소들을 가지는 다중 집합 A 를 A = <U, f> 로 표현하며 A 에 대하여 다음과 같이 정의한다.[3]

U = {a, b}
A = {a, a, b}

$$f(a) = 2, f(b) = 1$$

위의 다중 집합 A 에는 집합 U 의 원소인 a 와 b 를 가지고 있으며 a 의 Multiplicity 는 2 이고 b 는 1 에 해당된다. 또한 다중 집합 B = <U, g>에 대하여 다음과 같이 정의한다.

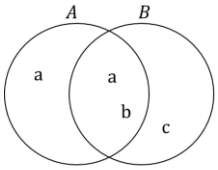
B = {a, b, c}

$$g(a) = 1, g(b) = 1, g(c) = 1$$

두 다중 집합 A 와 B 의 합집합을 A∪B = <U, p> 로 표현하고 A 와 B 의 교집합을 A∩B = <U, q>을 구하기 위해서는 두 집합의 원소들의 Multiplicity 를 활용하여 다음과 같이 정의할 수 있다.

$$x \in U \text{ 일 때, } \begin{cases} p(x) = max(f(x), g(x)) \\ q(x) = min(f(x), g(x)) \end{cases}$$

위의 정의된 식들과 2.1 의 Jaccard Similarity 을 활용하여 두 다중 집합 A 와 B 의 유사도를 다음과 같이 계산할 수 있다.



$$Jaccard(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{\sum_{x \in U} \min(f(x), g(x))}{\sum_{x \in U} max(f(x), g(x))} = \frac{g(a) + f(b) + f(c)}{f(a) + g(b) + g(c)} = \frac{2}{4}$$

Figure 2. Multiset Jaccard Similarity

3. Min-Max Hash

기존의 방식으로 다중 집합에서의 유사도를 구하기 위해서는 모든 원소들의 Multiplicity 에 대한 정보를 모두 저장해야 하거나 순서가 존재하지 않는 집합의 특성을 가진 것을 고려하면 원소 하나 당 다른 다중 집합의 원소 개수만큼 서로 비교를 해야한다.

이러한 연산 횟수를 효과적으로 줄이기 위해서 Min-Hash 의 특성을 활용하여 다중 집합의 표본을 구하고 표본 간의 유사도를 통한 실제 두 다중 집합의 유사도를 근사적으로 추정한다.

U = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11 }
A = { 1, 1, 3, 3, 4, 5, 5, 5, 6, 6, 10, 11, 11 }
B = { 1, 1, 1, 2, 2, 2, 3, 4, 5, 6, 7, 7, 8, 8, 9, 10 }

집합 U 의 원소들에 대하여 A = <U, f>와 B = <U, g>가 존재할 때 두 다중 집합에 대하여 다음과 같이 정의한다.

Definition 1. (Weight) 다중 집합은 중복된 원소에 대하여 해시 함수에 사상 시킬 때 같은 값으로 충돌(collision)이 발생한다. 그래서 다중 집합에 속해있는 모든 원소들에 대하여 해시 함수에 사상했을 때 충돌이 발생한 횟수를 해당 원소의 Weight 로 정의한다.

Definition 2. (Weight Signature) 집합에서는 Minhash Value 들로 집합의 표본을 나타내며 이를 Sig 로 정의하였다. 다중 집합에서는 Min-hash Value 뿐만 아니라 해당 Min-hash Value 의 Weight 를 나타내기 위해 다중 집합 A = <U, f>에 대하여 Weight 를 나타낼 수 있는 함수를 $w_A : x \mapsto f(\chi)$ 로 표현하고 Weight Signature 를 다음과 같이 $Weight_A$ 로 정의한다.

$$Weight_A = [w_A(h_{min_1}(A)), \dots, w_A(h_{min_n}(A))]$$
$$Weight_B = [w_B(h_{min_1}(B)), \dots, w_B(h_{min_n}(B))] \quad (1)$$

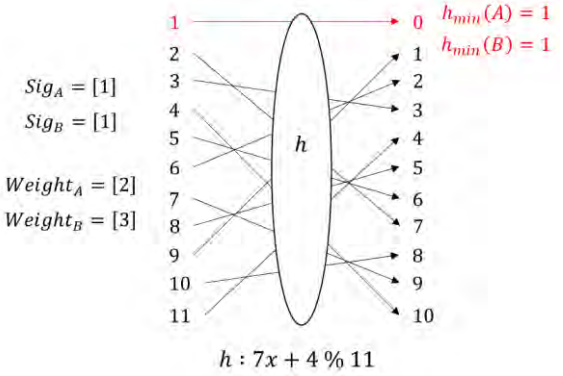


Figure 3. Weight Signature

Definition 3. (Min-Max Hash) Min-Hash 를 통하여 추정된 두 집합 A 와 B 의 유사도는 중복된 원소를 정의하지 않기 때문에 MinHash Value 값이 동일한 사건에 대하여 유사도 가중치를 0 과 1 로만 정의한다. 그러나 다중 집합에서는 중복된 원소를 포함하고 있기 때문에 MinHash Value 가 서로 동일하더라도 두 집합에서 차지하는 비율이 어느 정도인 지 알 수 없다. 그래서 MinHash Value 의 Weight Signature 를 활용하여 다중 집합 A = <U, f> B = <U, g> 유사도 가중치를 다음과 같이 2 가지로 정의한다.

$$\widehat{A \cap B}_k \begin{cases} w_A(h_{min_k}(A)) & h_{min_k}(A) \le h_{min_k}(B) \\ w_B(h_{min_k}(B)) & h_{min_k}(A) > h_{min_k}(B) \end{cases}$$
$$\widehat{A \cup B}_k \begin{cases} w_B(h_{min_k}(B)) & h_{min_k}(A) \le h_{min_k}(B) \\ w_A(h_{min_k}(A)) & h_{min_k}(A) > h_{min_k}(B) \end{cases} \quad (2)$$

Definition 4. (Similarity) Eq. 1 과 Eq. 2 를 바탕으로 n 개의 해시 함수를 활용하여 생성된 다중 집합 A 와 B 의 유사도를 다음과 같이 근사적으로 추정할 수 있다.

$$\text{Jaccard(A,B)} = \frac{|A \cap B|}{|A \cup B|} = \frac{\sum_{x \in U} max(\ f(x), g(x)\ )}{\sum_{x \in U} max(\ f(x), g(x)\ )} \quad (3)$$

4. 알고리즘

본 논문에서 제시하는 알고리즘은 총 2 단계로 이루어져 있다. 1 단계에서 두 개의 다중 집합 A, B 의 Weight Signature 를 구한다. 2 단계에서는 1 단계에서 구해진 Signature 와 Weight Signature 을 바탕으로 유사도를 계산한다.

Algorithm 1 Min-Max Hash
Input: Multiset *A, B* as Dataset, *Value* as Size of Signature
Output: Jaccard(*A*, *B*)

```
1.  N ← Value
2.  for each i ∈ N do
3.      Sig_A [i] ← 0
4.      Sig_B [i] ← 0
5.      Weight_A [i] ← 0
6.      Weight_B [i] ← 0
7.  Min ← 0
8.  Max ← 0
9.
10. WeightSignature(A, Sig_A, Weight_A)
11. WeightSignature(B, Sig_B, Weight_B)
12.
13. for each i ∈ N do
14.     if Sig_A [i] = Sig_B [i] then
15.         Min ← Min + Weight_A [i]
16.         Max ← Max + Weight_B [i]
17.     else if Sig_A [i] > Sig_B [i] then
18.         Min ← Min + Weight_B [i]
19.         Max ← Max + Weight_B [i]
20.     else
21.         Min ← Min + Weight_A [i]
22.         Max ← Max + Weight_B [i]
23.
24. Return Min/Max
```

Figure 5. Min-Max Algorithm

Figure 5 에서는 Weight Signature 함수를 통하여 두 다중 집합 A 와 B 에 대한 Signature 와 Weight 정보를 구하여 유사도를 계산한다.

Algorithm 2 WeightSignautre
Input: Multiset A, Sig_A, $Weight_A$
Output: Sig_A, $Weight_A$

```
1.  for each i ∈ N do
2.      a ← random integer with 1 ≤ a ≤ N
3.      b ← random integer with 1 ≤ b ≤ N
4.      p ← minimum prime number with p ≥ N
5.      h(x) ← ax + b mod p
6.      weight ← 0
7.      for each j ∈ |A| do
8.          minhash ← N
9.          Val ← min(minhash, h(Aj))
10.         if Val = minhash then
11.             weight ← weight + 1
12.         else if Val < minhash then
13.             weight ← 0
14.
15.     Sig_A [i] ← minhash
16.     Weight_A [i] ← weight
```

Figure 4. Weight Signature Algorithm

Figure 6 에서는 다중 집합 A 에 대하여 n 개의 해시 함수를 통하여 Signature 와 Weight Signature 를 구한다. 간단하게 다중 집합 A 의 원소들을 탐색하면서 해시 함수에 사상하는 값들 중 가장 작은 값을 구하고 Weight 는 동일한 원소를 해시 함수에 사상했을 때의 조건을 검사하여 계산한다.

5. 결론

본 논문에서는 Min-Hash 를 활용하여 집합의 영역에서 확장시킨 다중 집합의 데이터들의 유사도를 효과적으로 추정할 수 있는 방안을 제시하였다. 향후 다중 집합의 데이터를 활용할 수 있는 영역에서 기존의 제시된 다중 집합의 유사도 측정 방법들에 대한 상대적인 평가가 필요하며 구체적인 실험을 통하여 본 논문에서 제시한 방법의 효율성을 입증할 계획이다. 이를 통하여 다양한 데이터 영역에서의 유사도 활용 방안을 제시할 수 있을 것이라 본다.

참고문헌

[1] Leskovec, J., Rajaraman, A., Ullman, J : Mining of Massive Datasets. (2014)
[2] Andri Z. B : On the resemblance and containment of documents. Compression and Complexsity of Sequences, IEEE, pp 21-29. (1997)
[3] Apostolos S. : Mathematics of Multisets. WMC 2000: Multiset Processing, Springer, pp 347-358 (2001)