

블룸필터의 오류 확률에 대한 분석*

김성용,[†] 김지홍[‡]
세명대학교

An Analysis on the Error Probability of A Bloom Filter*

SungYong Kim,[†] JiHong Kim[‡]
Semyung University

요 약

최근 정보통신 기술의 발달로 인하여 데이터의 양이 점차 증가하고 있으며, 이에 대한 처리와 관련된 연구가 활발히 진행되고 있다. 주어진 집합 내에 특정 개체의 존재여부를 알기 위해 사용되고 있는 블룸필터는 데이터의 공간 활용에 매우 유용한 구조이다. 본 논문에서는 블룸필터에서 발생할 수 있는 오류 확률을 소개한다. 특히 실험실적 분석방법에 의하여 수정된 긍정오류 확률에 대한 일반식을 유도한다. 마지막으로 지금까지 사용되고 있는 블룸필터에 대한 긍정오류 확률식과 이에 대한 관련논문을 이용하여 비교, 분석한다.

ABSTRACT

As the size of the data is getting larger and larger due to improvement of the telecommunication techniques, it would be main issues to develop and process the database. The bloom filter used to lookup a particular element under the given set is very useful structure because of the space efficiency. In this paper, we introduce the error probabilities in Bloom filter. Especially, we derive the revised false positive rates of the Bloom filter using experimental method. Finally we analyze and compare the original false positive probability of the bloom filter used until now and the false decision probability proposed in this paper.

Keywords: Bloom filter, False Positive Rate, False Negative Rate

1. 서 론

시스템 내에 저장된 캐쉬 메모리, 라우팅 테이블 등의 데이터 증가로 인하여, 데이터의 존재여부를 확인할 수 있는 인덱스로서 블룸필터를 자주 사용한다. 블룸필터는 n 개의 입력요소에 대하여 k 개의 해시함수 결과 값을 사이즈가 m 인 배열의 해당 비트에 '1'로 설정하는 것으로 데이터의 공간 활용에 매우 유용한

구조이다. 하지만 블룸필터의 특성상 존재하지 않는 데이터를 존재한다고 판단하는 긍정오류 확률(false positive rate) 발생할 수 있다. 반면에 부정오류 확률(false negative rate)은 존재하지 않는다는 장점이 있다.

1970년에 블룸필터가 제안[1]된 이래로, 블룸필터에 대한 긍정오류 확률에 대한 다양한 연구와 긍정오류 확률에 대한 최소값 및 하한경계에 대한 분석[2]이 이루어 졌으며, 이후에도 블룸필터의 응용에 관한 연구로서, 일반화된 블룸필터[7], 압축 블룸필터, 스펙트럼 블룸필터, 캐쉬메모리에 적용 가능한 카운팅 블룸필터[6]등에 대한 연구가 이루어졌으나, 모두 기본적으로 논문[1]에서 제시된 긍정오류 확률을 이용하

접수일(2014년 2월 3일), 수정일(1차: 2014년 7월 28일, 2차: 2014년 9월 12일), 게재확정일(2014년 9월 12일)

* 이 논문은 2013학년도 세명대학교 교내학술연구비 지원에 의해 수행된 연구임

[†] 주저자, ksy4765@naver.com

[‡] 교신저자, jhkim@semyung.ac.kr(Corresponding author)

여 이루어 졌다.

Bose(3)는 2형 Stirling 수를 이용하여 수정된 긍정오류확률식을 제시하였으나, 많은 오류를 포함하고 있다. Christensen(4)은 Bose의 논문을 바탕으로 확률론적으로 분석하였다.

본 논문에서는 두 논문에서 제시한 방법과 다른 실험실적 방법에 의해 긍정오류확률을 도출한다. 그리고 Bose(3)와 Christensen(4)에 의해 제시된 수정된 긍정오류확률식과 본 논문의 결과와 비교한다.

본 논문은 1장은 서론, 2장 본문에서는 블룸필터의 개념, 긍정오류확률의 개념과 함께 실험실적 방법에 의해 일반화된 긍정오류확률식을 유도한다. 그리고 기존의 긍정오류 확률식과 이에 대한 문제점을 분석한 논문들과의 결과를 비교, 분석한다. 그리고 마지막으로 3장에서는 결론을 맺는다.

II. 본 론

2.1 블룸필터의 오류 확률

2.1.1 블룸필터 개념

블룸필터는 1970년 Burton Howard Bloom에 의해 고안된 것으로 원소가 집합에 속하는지 여부를 검사하는데 사용되는 확률적 자료 구조이다(2).

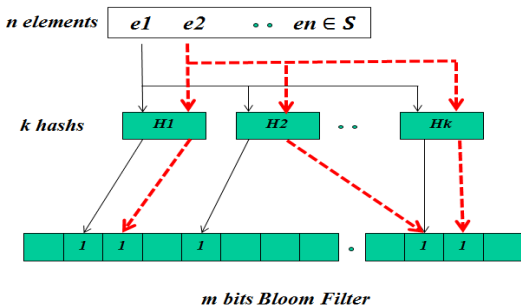


Fig. 1. The element Insertion process

그림 1과 같이 초기에 모두 “0”으로 설정된 m ($=16$) 비트의 블룸필터를 이용하여, 개체에 대한 해쉬함수 결과값에 따라 블룸필터의 내용이 “1”로 세트된다. 예를 들어 개체 e_1 에 대한 세 개의 해쉬함수 결과 값이 각각 2, 5, 14 이라면, 이에 해당되는 비트가 “1”로 설정된다. 마찬가지로 개체 e_2 에 대해서 3, 14, 15 번째 비트가 “1”로 설정된다. 이러한 블룸필터

에는 개체 e_1, e_2 가 존재함을 표시하고 있다. 블룸필터를 이용하여 개체의 존재여부를 확인하기 위한 룩업(look up) 과정에서 그림 2와 같이 정상적으로 입력된 e_1 의 경우에는 실제 존재하는 개체이므로 true positive라 하고, 반면에 입력되지 않은 개체 x 에 대해서는 실제로 존재하지 않지만, 블룸필터 룩업과정을 통하여 존재하는 것으로 표현되기 때문에 이를 긍정오류(false positive)라 한다.

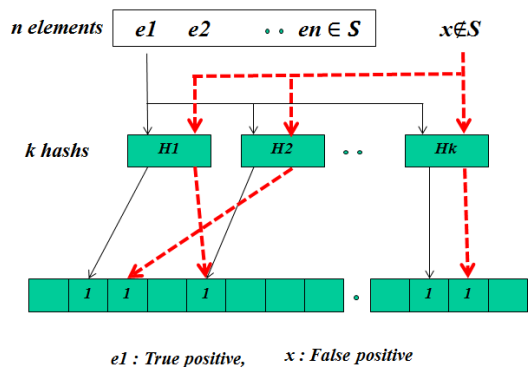


Fig. 2 The Look up process

이와 같이 블룸 필터는 많은 양의 데이터에 대한 존재여부를 빠르고 효율적으로 검색할 수 있다는 장점을 가지고 있다. 즉, 긴 정보를 짧게 줄여 비트 플래그로 바꿔 저장되기 때문에, 해당 정보를 이용하여 데이터의 존재여부를 판단하는 것이다.

2.1.2 블룸필터의 긍정오류 확률

긍정오류란 긍정으로 판정된 것 중에서의 실제로 존재하지 않는 오류를 말한다. 이와 같이 블룸필터는 주어진 공간을 이용하여 요소의 존재여부를 표현하기 때문에, 이러한 긍정오류가 발생할 수 있는 확률을 항상 가지고 있다. 그러나 이러한 긍정오류확률은 m, n, k 을 적당한 값으로 조절하여 제어할 수 있다(2).

1970년 Bloom에 의해 제안된 이래, 지금까지 블룸필터의 긍정오류 확률 $P_{fp.org}$ 는 아래의 식으로 정의되어 지금까지도 많은 논문에서 이를 인용하고 있다.

$$P_{fp.org} = [1 - (1 - \frac{1}{m})^{nk}]^k \approx (1 - e^{-nk/m})^k \quad (1)$$

블룸필터는 초기에 모두 “0”으로 설정된 상태에서

해쉬함수의 결과값에 의해 블룸필터를 설정한 것이다. 즉, n 개의 입력요소에 대하여 존재유무를 확인할 수 있도록 k 개의 해쉬함수 결과값을 m 비트의 블룸필터에 각 비트로 세트시킨 결과를 이용하여 검색과정에서 이를 이용하는 방식이다.

위의 식에서 해쉬함수의 결과값은 항상 균등 분포 (equally distributed)로 가정한다. 이와 더불어 k 개의 해쉬함수 결과값이 각각 독립이라는 판단으로 전개된 결과이다.

위의 긍정오류 확률식의 내부 수식을 P_{nk} 이라하면, $P_{nk} = 1 - (1 - \frac{1}{m})^{nk}$ 은 nk 번 시행에 대하여 임의의 비트가 "1"로 세트될 확률을 의미하며, 검색과정에서 특정 요소에 대한 k 개의 해쉬함수의 결과값이 모두 이들 비트를 지정하였을 때 긍정오류가 발생한다는 것을 의미한다.

2.2 블룸필터의 오류 확률

2.2.1 긍정오류 확률과 부정오류확률

블룸필터에 입력되는 n 개의 요소에 대하여 m 비트의 블룸필터에 적용한 경우, 검색과정에서 블룸필터의 결과값에 따라 다음과 같이 구분될 수 있다.

먼저 표 1과 같이, 블룸필터 검색을 통하여 긍정 (positive)으로 판단되는 경우의 수(N_p)와 부정 (negative)으로 판단되는 경우의 수(N_n)의 합으로 표현할 수 있다. 긍정으로 판단된 N_p 는 블룸필터에는 존재한다고 하지만, 실제 존재하는 경우의 수 N_{tp} (true positive)와 실제 존재하지 않지만 존재한다고 판단한 경우의 수 N_{fp} (false positive)의 합으로 표현할 수 있다.

Table 1. Four Probabilities of the Bloom Filter

	BF=1 (S_p)	BF≠1 (S_n)
Exist ($x \in S$)	N_{tp}	N_{fn}
No Exist ($x \notin S$)	N_{fp}	N_{tn}

블룸필터에서 발생할 수 있는 오류는 긍정오류와 부정오류가 있다.

긍정오류확률은 실제로 집합에는 존재하지 않는 것에 대하여 블룸필터에 의해 존재하는 것으로 판단된

경우를 의미하며, 다음과 같이 표현된다.

$$P_{fp} = \Pr(BF=1|x \notin S)$$

$$P_{fp} = \frac{N_{fp}}{N_{tn} + N_{fp}} \tag{2}$$

반면에 블룸필터에 의해 존재하지 않는 것으로 판단되지만, 실제로 존재할 확률은 부정오류(false negative)라 정의하고, 부정오류확률을 다음과 같이 표현할 수 있다.

$$P_{fn} = \Pr(BF \neq 1|x \in S)$$

$$P_{fn} = \frac{N_{fn}}{N_{tp} + N_{fn}} = 0 \tag{3}$$

블룸필터는 특성상 부정오류는 발생하지 않기 때문에 $P_{fn} = 0$ 이 된다. 이와 같은 특성으로 인하여 블룸필터의 긍정오류확률은 객체에 대한 실제 존재 여부를 확인할 수 있는 도구로 유용하게 사용된다.

표 2의 예는 집합에 속한 한 개의 요소 e1에 대한 해쉬함수 결과를 블룸필터에 적용한 결과를 표현한 것이다. 이때 집합에 속하지 않은 요소 x에 대한 해쉬함수 결과값을 이용한 요소의 존재여부를 확인하기 위한 look up 과정에서 긍정오류발생여부를 보인 표이다.

Table 2. All cases of the $m=4, n=1, k=1$

$e1 \in S, x \notin S$		Bloom filter				FP
e1	x	m0	m1	m2	m3	
0	0	1	0	0	0	1
0	1	1	0	0	0	
0	2	1	0	0	0	
0	3	1	0	0	0	
1	0	0	1	0	0	
1	1	0	1	0	0	1
1	2	0	1	0	0	
1	3	0	1	0	0	
2	0	0	0	1	0	
2	1	0	0	1	0	
2	2	0	0	1	0	1
2	3	0	0	1	0	
3	0	0	0	0	1	
3	1	0	0	0	1	
3	2	0	0	0	1	
3	3	0	0	0	1	1
$N_T=16$						$N_{fp}=4$

첫 번째 라인의 경우, e1에 대하여 해쉬함수 결과

값이 "0"인 경우, 결과적으로 블룸필터의 $m_0=1$ 이 된다. 이때 입력집합에 속하지 않은 요소 x 에 대한 해쉬값이 "0"으로 나타나면 긍정오류가 발생한다.

결과적으로 $m=4, n=1, k=1$ 인 경우에는 집합에 속하지 않은 요소에 대해 존재한다고 판정된 FP(False Positive)가 4개이며, 나머지 12개는 모두 TN(True Negative)로 간주할 수 있으며, 긍정오류확률(P_{fp})은 $4/16$ 이 된다.

표 2의 경우와 같이, $m=4, n=1, k=1$ 인 경우의 긍정오류확률을 구하기 위하여 (1)식에 대입하면 $P_{fp(k=1).org} = 1 - (1 - \frac{1}{4})^1 = \frac{1}{4}$ 이 된다. 이는 실제로는 존재하지 않지만 블룸필터에 의해 긍정으로 판단된 경우를 나타내고 있으며, 표1을 통해서도 실제 긍정오류확률은 $1/4$ 이 된다는 것을 알 수 있다.

2.2.2 일반화된 긍정오류 확률식

전 절에서와 같이 $k=1$ 인 경우에는 쉽게 긍정오류확률을 구할 수 있다. 그러나 $k>1$ 인 경우에 적용될 수 있는 긍정오류확률은 다소 복잡하다.

다음은 $m=4, n=1, k=2$ 인 경우를 살펴본다.

Table 3. All cases of the $m=4, n=1, k=2$

$e1 \in S$			Bloom filter			FP occurred by $x \notin S$ elements	FP
k1	k2	m0	m1	m2	m3		
0	0	1	0	0	0	(0,0)	1
0	1	1	1	0	0	(0,0)(0,1)(1,0)(1,1)	4
0	2	1	0	1	0	(0,0)(0,2)(2,0)(2,2)	4
0	3	1	0	0	1	(0,0)(0,3)(3,0)(3,3)	4
$N_T=256, N_{fp}=4+4 \times 12=52, N_{tn}=204$							

표 3은 표 2의 예와 마찬가지로 방법으로 집합에 속한 한 개의 요소 $e1$ 에 대한 블룸필터 결과 값을 설정한 후, 집합에 속하지 않은 x 요소의 해쉬함수 결과 값에 대한 FP 발생여부를 표시하였다. 전체 64개의 입력 경우 중에 $k1=0$ 인 경우만을 상세히 표시하였다. 나머지 $k1=1,2,3$ 인 경우에도 $k1=0$ 인 경우와 동일한 방법을 사용한다.

첫 번째 라인의 경우, $e1$ 에 대한 두 개의 해쉬함수 결과 값이 "0"인 경우를 의미하며, 결과적으로 블룸필터의 $m_0=1$ 이 된다. 이때 입력집합에 속하지 않은 요소 x 에 대한 두 개의 해쉬값이 모두 "0"으로 나타나면

긍정오류가 발생한다.

마찬가지 방법으로 해쉬함수들의 결과값을 16가지 경우의 수에 모두 적용하면 결과적으로 $m=4, n=1, k=2$ 인 경우에는 집합에 속하지 않은 요소에 대해 존재한다고 판정된 전체 FP는 52개이다. 나머지 204개는 모두 TN로 간주할 수 있다. 그러므로 긍정오류확률(P_{fp})은 $52/256=0.203$ 이 된다. 반면에 $m=4, n=1, k=2$ 를 기존의 긍정오류 확률식(식 1)에 대입하면 다음과 같다.

$$P_{fp(k=2).org} = \left(1 - \left(1 - \frac{1}{4}\right)^2\right)^2 = 0.19$$

이와 같이 $k=1$ 인 경우에는 지금까지 사용되고 있는 긍정오류확률식이 맞지만, $k>1$ 인 경우에는 기존의 긍정오류확률식이 잘못 되었음을 알 수 있다.

그러므로 일반화된 긍정오류확률식을 구하기 위하여 그림과 같이 3단계로 구분하여 유도한다.

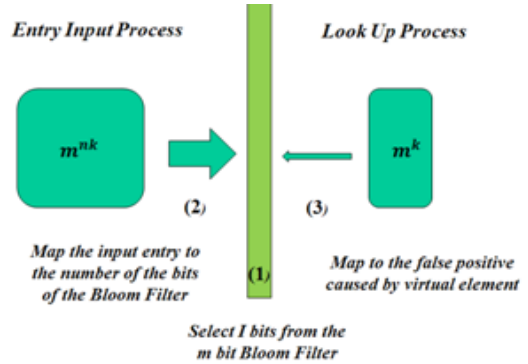


Fig. 3. The calculation process of FP

- 1단계 : m 개의 블룸필터로부터 i 비트 선정
 - 2단계 : 입력개체로부터 블룸필터의 i 비트 맵핑
 - 3단계 : 가상개체 x 와 블룸필터의 i 비트와 맵핑
- 단계1의 경우에는 mC_i 이며, 단계2의 경우에는 다음과 같은 식으로 표현될 수 있다.

$$S_i = i^{nk} - \sum_{j=1}^{i-1} (iC_j \times S_j) \tag{4}$$

여기서 S_i 는 모든 입력의 해쉬함수 결과에 따라 i 개의 블룸필터 비트를 설정한 경우의 수에 해당된다. S_1 은 1개의 비트를 세트한 경우이며, S_2 는 2개의 비

트를 세트한 경우에 해당된다. 그러므로 S_i 는 S_{i-1} 까지의 값을 순환적으로 제외한 값이 된다. 결국 단계1과 단계2를 종합하면 블룸필터의 i 개의 비트까지 설정한 모든 경우의 수가 해당된다.

$$N_i = {}_m C_i \times S_i \tag{5}$$

결국 N_i 는 입력과정을 통하여 블룸필터의 i 개의 비트를 세트할 수 있는 모든 경우의 수를 포함하고 있다.

단계 3에서는 집합에 속하지 않는 개체 x 에 대한 k 개의 해쉬함수 결과값이 N_i 와 매핑되는 개수를 구하면 긍정오류가 발생하는 총 개수를 구할 수 있다.

$$N_{fp} = \sum_{i=1}^{nk} (N_i \times i^k) \tag{6}$$

전체 경우의 수가 $m^{(n+1)k}$ 개이므로, 긍정오류 확률은 다음과 같은 식으로 정의할 수 있다.

$$P_{fp} = \frac{1}{m^{(n+1)k}} \sum_{i=1}^{nk} ({}_m C_i \times S_i \times i^k) \tag{7}$$

Table 4. The values of the S_i, N_i

n	k	m	S_1	N_1	S_2	N_2	S_3	N_3	S_4	N_4
1	2	4	1	4	2	12				
2	2	4	1	4	14	84	36	144	24	24

표 4에서 $m=4, n=1, k=2$ 인 경우에는 $N_{fp} = 4 \times 1^2 + 12 \times 2^2 = 52, P_{fp} = 52/256 = 0.203$ 이며, $m=4, n=2, k=2$ 인 경우는 $N_{fp} = 4 \times 1^2 + 84 \times 2^2 + 144 \times 3^2 + 24 \times 4^2 = 2020, P_{fp} = 2020/4096 = 0.493$ 이 된다.

2.2.3 기존에 제안된 긍정오류확률과의 비교

긍정오류확률의 오류를 최초로 분석한 논문은 Bose[3]에 의해 시행되었다. Bose는 2형 Stirling 수(Stirling Number of the second type)를 이용하여 다음과 같이 유도하였다.

$$N_{fp} = \sum_{i=1}^m i^k ({}_m C_i) \sum_{j=0}^i (-1)^j {}_i C_j j^{kn} \tag{8}$$

Bose[3]의 결과는 두 가지 오류를 범하고 있다. 모두 구간과 관련된 오류이다.

첫 번째 구간은 $i=1$ 에서 m 까지가 아니라, 입력에 의해 블룸필터에 세트될 수 있는 nk 까지 적용되어야 하며, 두 번째 구간은 $j=0$ 에서 시작하고, $(-1)^j$ 을 사용함으로써, 전체적인 FP 수가 음의 값으로 나타나게 된다. 즉, Stirling 수를 적절히 적용하지 못하였다.

Christensen[4]은 기본적으로 Bose[3]의 결과를 이용하였고, 확률론적으로 분석하여 다음과 같이 정의하였다.

$$N_{fp} = \sum_{i=1}^m i^k ({}_m C_i) \sum_{j=1}^i (-1)^{i-j} {}_i C_j j^{kn} \tag{9}$$

Christensen[4]에 의해 제안된 수식에서도 첫 번째 합에 대한 오차가 남아있다. 단계3에서의 집합에 속하지 않는 가상입력에 대한 최종 긍정오류의 개수를 구하기 위해서는 첫 번째 합에 구간이 m 이 아니라 nk 가 되어야 한다. 일반적으로 nk 값은 블룸필터의 크기 m 에 비해 적기 때문이다. 그러나 식 9에서 i 가 nk 보다 큰 경우에는 값이 "0"이 되기 때문에 본 논문에서 제안한 방식과 동일한 결과를 갖는다.

마지막으로 표 5는 네 가지의 긍정오류확률식의 값을 비교한 것으로 크기가 다른 m, n, k 을 각 식에 대입한 값의 결과이다.

Table 5. The comparison between four equations

n	k	m	$P_{fp.org}$	P_{fp}	$P_{fp.bose}$	$P_{fp.christensen}$
1	2	4	0.191	0.203	0.171	0.203
2	2	4	0.467	0.493	-0.141	0.493
2	2	8	0.171	0.177	0.039	0.177
2	3	8	0.167	0.184	-0.015	0.184
2	4	8	0.185	0.218	0.010	0.218

표 5를 보면, 본 논문에서 제안하는 수식과 Christensen이 제안한 수식은 방식이 다르지만 N_{fp} 의 수가 일치함에 따라 긍정오류확률이 같은 값을 나타내고 있으며, 또한 기존의 긍정오류확률에 비해 비교적 큰 값을 나타내고 있는 것을 알 수 있다. 그에 비해 Bose의 수식은 다른 수식에 비해 긍정오류확률값이 현저하게 낮으며, 때때로 음수의 값이 나타나기도 한다.

본 논문에서 제안한 수식은 Christensen의 수식과 비교하여 결과적으로 긍정오류확률값이 같지만 경우의 수를 통해 논리적이고 계산된 값에 대해 검증된 형태를 가지고 있다.

III. 결 론

1970년에 Bloom에 의해 발표된 블룸필터 [1]는 해쉬함수의 결과값이 균등하게 분포되어 있다는 가정하에 지금까지 사용되고 있으나, 본 논문에서는 이러한 기존의 긍정오류 확률식 (식1)의 오류를 분석하였다. $k=1$ 인 경우에는 식1이 성립한다. 그러나 $k > 1$ 이 경우에는 k 개의 해쉬함수 결과값이 각각 독립이라는 가정으로, 각각의 해쉬함수 결과에서 적어도 한번 "1"이 될 확률들에 대한 k 곱으로 표현되었다. 그러나 실제로 본 연구를 통해 나타난 결과는 해쉬함수의 개수가 커짐에 따라 긍정오류 확률은 다소 증가함을 알 수 있으며, 또한 k 가 증가함에 따라 기존의 긍정오류확률에 비해 본 논문에서 제안된 긍정오류확률이 조금씩 폭이 증가함을 알 수 있다.

본 연구에서는 정확한 긍정오류확률의 갯수를 구하기 위하여 3단계로 분석하였다. 1,2 단계에서는 실제 입력에 의해 블룸필터가 설정되는 경우의 수를 분석하였으며, 마지막 3단계에서는 블룸필터에 세트된 비트수에 따라 집합에 속하지 않는 가상입력요소에 의해 발생하는 긍정오류의 수를 계산하여, 이를 모두 합산함으로써 전체 긍정오류의 개수를 구하였다. 이와 더불어 기존의 긍정오류확률식에 대한 문제점을 수학적으로 분석한 논문[3,4]와 비교하여, 이들이 제안했던 수정된 긍정오류확률식과 차이가 있음을 보였다.

본 논문에서는 일반적인 블룸필터에서의 긍정오류 확률을 실험실적 방법을 이용하여 일반적인 식을 유도하였다. 사용된 m, n, k 값은 비록 적은 값을 사용하였지만, 다양한 방법의 검증을 통하여 일반적으로 블룸필터에서 발생하는 긍정오류 확률을 표현할 수 있는 식을 도출하였다. 만일 허용할 수 있는 긍정오류 확률값이 미리 주어진다면, 이에 따라 m, n, k 의 값을 적절하게 사용하여 구현할 수 있을 것으로 사료된다. 또한 추후에는 제안된 긍정오류 확률을 이용하여 적절한 해쉬함수의 개수, 긍정오류 확률을 최소화하기 위한 연구 등을 수행할 예정이다.

References

- [1] B. Bloom, "Space/Time Trade-offs in Hash Coding with Allowable Errors" Communications of the ACM, Vol. 13, No. 7, pp. 422-426, Nov. 1970.
- [2] A. Broder and M. Mitzenmacher, "Network Applications of Bloom Filters : A Survey", Internet Mathematics, Vol. I, No. 4, pp. 485-509, 2004.
- [3] P. Bose, H. Guo, and E. Kranakis, "ON THE FALSE-POSITIVE RATE OF BLOOM FILTERS", School of Computers Science, Carleton University, May. 2004.
- [4] K. Christensen, A. Roginsky, and M. Jimeno, "A new analysis of the false positive rate of a Bloom filter", Information Processing Letters 110, pp. 944-949, 2010.
- [5] O. Rottenstreich and I. Keslassy, "The Bloom Paradox: When not to Use a Bloom Filter?", INFOCOM, pp. 1638-1646, Mar. 2012.
- [6] L. Fan, P. Cao, and J. Almeida, "Summary Cache : A Scalable Wide-Area Web Cache Sharing Protocol.", IEEE/ACM Transactions on Networking, vol. 8, no. 3, pp. 281-293, June. 2000.
- [7] R. P. Laufer, P. B. Velloso, and O. C. M. B. Duarte, "A Generalized Bloom Filter to Secure Distributed Network Applications", Computer Networks, Vol. 55, no. 8, pp. 1804-1819, June. 2011.

〈 저 자 소 개 〉



김 성 용 (SungYong Kim) 학생회원
2013년 2월: 세명대학교 정보통신학부 졸업
2013년 3월~현재: 세명대학교 정보통신학 석사과정
〈관심분야〉 정보보호 응용, 데이터베이스 보안, 의료정보 보안



김 지 홍 (JiHong Kim) 종신회원
1982년 2월: 한양대학교 전자공학과 졸업
1984년 2월: 한양대학교 전자통신공학과 석사 졸업
1996년 3월: 한양대학교 전자통신공학과 박사 졸업
1991년 3월~현재: 세명대학교 정보통신학부 교수
〈관심분야〉 네트워크 및 정보보호, 정보보호 응용, 데이터베이스 보안