

멀티미디어 스트리밍을 위한 프록시 서버에서의 사용자 선호도 기반 캐쉬 교환 알고리즘

이준표⁰ 박성한
 한양대학교 컴퓨터공학과
 (jplee, shpark)@cse.hanyang.ac.kr

User Popularity based Cache Replacement Algorithm for Multimedia Streaming in Proxy Server

Jun Pyo Lee⁰ Sung Han Park
 Dept. of Computer Science & Engineering, Hanyang University

요 약

대용량의 멀티미디어 데이터들을 다수의 사용자가 동시에 요구할 경우 접속이 끊기거나 데이터의 손실과 같은 접속 성능이 현저하게 저하되는 문제가 발생할 수 있다. 이는 대부분의 사용자들의 요청을 원거리의 Central Server가 단독으로 처리하기 때문에 발생하는 문제이다. 따라서 Central Server의 Load 감소와 초기 지연시간 및 비디오 패킷의 손실의 문제점을 해결하기 위해 프록시 서버를 사용자의 근 거리에 위치시키는 방법을 이용한다. 대용량의 멀티미디어 데이터들을 복잡한 인터넷의 중간경로를 거치지 않고 다수의 사용자들에게 직접 전달함으로써 데이터의 손실을 방지함과 동시에 보다 안정적이고 빠른 속도로 제공하는 것이 가능하다. 특징적으로 프록시 서버는 Central Server에 비해 비교적 제한된 용량을 가지고 있다. 따라서 본 논문에서는 사용자의 선호도를 기반으로 하여 앞으로 요구할 것이라고 예측되는 비디오 데이터들만을 선별적으로 저장하도록 하는 알고리즘을 제안한다. 시뮬레이션을 통해 제안하는 캐쉬 교환 알고리즘을 기존의 알고리즘과 비교, 관찰한 결과 상대적으로 높은 Hit rate가 나타남을 확인하였다.

1. 서 론

현재 인터넷을 사용하는 기업과 사용자는 계속하여 폭발적으로 증가하고 있을 뿐만 아니라 관련 기술도 급성장하고 있다. 뿐만 아니라 Broadband Network의 급속한 성장에 따라 인터넷 상에서의 비디오 스트리밍 기술 역시 함께 발전했다. 비디오 스트리밍 기술을 사용하여 사용자는 실시간으로 원하는 비디오에 접근하여 사용하는 것이 가능하다. 그러나 인터넷에서 제공하는 멀티미디어 데이터 서비스의 품질과 속도에 대한 만족도는 크게 개선되지 않고 있다. 현재 서비스의 만족도를 향상시키기 위해 원거리에 위치하는 Central Server의 I/O Bandwidth를 증가시키는 방법을 사용하거나 또는 접근할 수 있는 사용자의 수를 제한하는 방법 등을 사용하고 있으나 이와 같은 방법은 대단히 제한적이며 QoS를 보장하지 못한다. 또한 최근에는 대용량의 멀티미디어 스트리밍 서비스를 위한 프록시 서버의 필요성을 인식하여 활용하기 시작했으나 Central Server에 비해 상대적으로 제한된 저장 용량을 비효율적으로 활용함으로써 많은 문제점들이 발생하고 있다. 따라서 프록시 서버에서 선별적으로 비디오 데이터를 저장하기 위한 효과적인 캐쉬 교환 알고리즘에 대한 연구는 중요한 과제로 인식되고 있다. 최근의 연구에서 이와 같은 문제점들을 해결하기 위해 가장 오래전에 접근한 비디오 데이터를 교체하고 최근의 비디오만을 유지하는 LRU(Least Recently Used) 알고리즘과 가장 적게 사용된 비디오 데이터를 교체 대상으로 선정하는 LFU(Least Frequently Used) 알고리즘 그리고 일정 시간동안에 발생한 사용자 접근 빈도수를 고려한 Time Popularity 알고리즘[1] 등 다양한 알고리즘이 제안되었다. 하지만 이와 같은 교환 알고리즘은 프록시 서버와 관련된 비디오 데이터에 대한 접근 패턴 정보를 전혀 고려하지 않고 있다.

본 논문에서는 제한된 용량을 가지는 Proxy Server의 효율적 활용을 위하여 비디오 데이터들에 대한 사용자 선호도인 Time Popularity와 프록시 서버에서 지속적으로 유지되는 캐쉬 교환 빈도수를 기반으로 한 새로운 캐쉬 교환 알고리즘을 제안한다. 본 논문은 다음과 같

이 구성된다. 2장에서는 기존의 캐쉬 교환 알고리즘에 대해 알아본다. 3장에서는 제안하는 알고리즘에 대해 설명한다. 4장에서는 성능평가 내용을 기술하고, 5장에서는 결론 및 향후 연구과제에 대해 기술한다.

2. 관련 연구

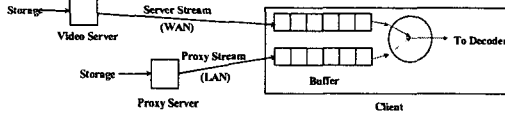
본 알고리즘의 기반이 되는 프록시 서버를 이용한 비디오 전송 모델에 대해서 알아본다. 제안 배경이 되는 기존의 대표적인 캐쉬 교환 알고리즘의 동작과 장단점을 살펴보고 제안하는 알고리즘의 기능을 확장하는데 응용되는 기법에 대해 기술한다.

2.1 프록시 서버를 이용한 비디오 전송 모델

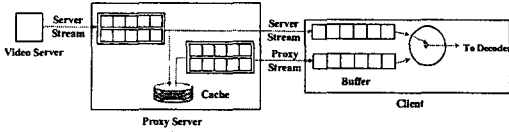
프록시 서버를 이용하여 비디오를 사용자에게 전송하는 방법으로 Client-Synchronization 모델(그림 1-(a))과 Proxy-Synchronization 모델(그림 1-(b))이 존재한다[2]. Client-Synchronization 모델은 비디오의 전반부 부분인 prefix만을 프록시에 저장하고 후반부 부분인 postfix를 원거리에 있는 비디오 서버가 서비스 하도록 구성하게 함으로써 비디오 전송의 초기 지연시간을 효과적으로 감소시키기 위한 목적으로 사용되는 방법이다. 그러나 이와 같은 모델은 프록시 서버와 비디오 서버 간에 전송 동기화에 대한 비용이 증가하게 되며 사용자가 증가할수록 이와 같은 문제는 보다 더 심각해지게 된다. 또한 비디오의 prefix와 postfix의 분할점(cut-off point)을 적절하게 선택하여 저장하지 못하는 경우, 원거리에 있는 비디오 서버의 부담이 급격하게 증가하게 되며, 이에 따라 비디오 서버의 전송 오류로 인한 재 전송 비용이 증가하게 되는 단점을 가지고 있다.

Proxy-Synchronization 모델은 사용자가 주로 요청하는 비디오들만을 선별하여 프록시 서버에 저장하고 사용자에게 서비스 하는 방법이다. 따라서 사용자가 주로 요청하는 비디오는 신속하게 사용자에게 서비스되며 프록시 서버에 미리 저장되어있지 않은 비디오만을 원격지의 비디오 서버에게 요청하여 필요한 경우 프록시 서버에 미리 저장해 두으로써 앞으로의 사용자의 요구에 반영할 수 있도록

하는 방법이다. 이 모델을 이용하여 서비스 하는 경우, 비교적 제한된 저장 공간을 프록시 서버가 효과적으로 운영할 수 있도록 하기 위해 새로운 데이터를 저장하고 기존의 비디오 중에서 가장 접근 가능성이 적은 비디오를 교체할 수 있도록 하는 알고리즘이 필요하다.



a) Client-Synchronization 모델



b) Proxy-Synchronization 모델

그림 1. 프록시 서버를 이용한 비디오 전송 모델

2.2 Time Popularity 알고리즘

제한된 저장 공간을 가지는 프록시 서버에 사용자들이 주로 요청하는 비디오들만을 선별적으로 저장하기 위한 방법 중에 하나인 Time Popularity 알고리즘은 초기에 설정된 시간 동안에 사용자들이 요청한 각 비디오들의 빈도수를 기반으로 하여 이루어진다. 그림 2과 같이 설정해 놓은 t_1 시간 동안에 이루어지는 사용자들의 비디오 패킷에 대한 접근 빈도수를 지속적으로 계산한 후, 캐쉬의 저장 공간 부족 시 t_2 시간동안 이 접근 빈도수를 기반으로 하여 빈도수가 가장 낮은 비디오를 제거 대상 비디오로 선정한다. 이는 설정된 시간동안에 낮은 Popularity 값을 갖는 비디오는 앞으로 요청될 가능성이 낮으며, Popularity가 높은 비디오는 이후에도 지속적으로 요청될 가능성이 높다는 가정을 기반으로 한 알고리즘이다. 그러나 이 알고리즘은 프록시 서버의 캐쉬와 관련된 비디오의 과거 상태정보를 활용하지 않고 있다. 일정 시간에서의 빈도수만을 고려하며 제거대상 비디오를 선정함으로써, 사용자에게 주로 요청되는 비디오일 경우라 하더라도 현재 시간에서 요청된 경우가 발생하지 않았다면 다음 시간 구간에서는 프록시 서버에서 삭제될 가능성이 큰 알고리즘이다. 즉, 제거대상 선정시, 전체 시간에 걸쳐 각 비디오가 사용자에게 의해 어떻게 요청되어져 왔는지에 대한 상태정보를 고려하지 않는다.



그림 2. Time Popularity 알고리즘

2.3 Video Prefix Selection

그림 3은 전체 비디오에 대한 사용자의 접근 성향을 나타낸다. 하나의 비디오는 여러 개의 패킷으로 구성되어진다. 또한 이들 패킷들을 전반부인 prefix와 후반부인 suffix로 구분할 수 있다[3]. 특징적으로 사용자들은 전체 비디오에서 전반부인 prefix만을 주로 요청하는 경향을 가지고 있다. 즉, b_1 에 대한 사용자의 접근 빈도수는 b_2 보다 높으며 b_2 는 b_3 보다 상대적으로 높은 접근 빈도수를 나타내다고 할 수 있다.

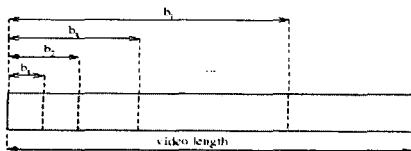


그림 3. Video Prefix에 대한 사용자 선호도

3. 사용자 선호도 기반 캐쉬 교환 알고리즘

2

Time Popularity 알고리즘은 초기에 설정된 Time t 에 의존적인 방법이다. 그러므로 본 논문에서는 이와 같은 문제점을 해결하기 위해 전체 시간에 걸쳐 계산 되어지는 비디오의 상태정보인 VPRN(Video Packet Replacement Number)을 이용한다. 또한 사용자의 비디오 접근 패턴을 고려하여 특정 비디오의 prefix를 구성하는 패킷 삭제 시에 접근 가능성이 상대적으로 낮은 suffix를 동시에 삭제시키고 그와 반대로 suffix를 구성하는 패킷 삭제 시에는 prefix 패킷들을 프록시 서버에 남겨둠으로써 프록시 서버의 저장 공간을 효율적으로 사용하고 자 한다.

제안하는 알고리즘은 제한된 저장 공간을 가지는 프록시 서버가 더 이상의 비디오 데이터들을 저장할 수 없을 때, 사용자에게 의해 앞으로 요청될 가능성이 가장 작은 비디오를 선정하여 제거한 후, 원격지의 Central Video Server로부터 전송되어 온 새로운 비디오 데이터를 저장하도록 하는 방법이다. 사용자가 원하는 비디오가 프록시 서버에 저장되어 있지 않을 경우, 원격지의 Central Server에게 해당 비디오를 요청하게 되며 이 요청된 비디오는 프록시 서버를 거쳐 사용자에게 전달되게 된다. 이때 프록시 서버는 해당 비디오를 캐쉬에 저장함으로써 이후에 동일한 비디오 요청 시에 서비스 하도록 한다. 그러나 프록시 서버는 Central Server에 비해 상대적으로 제한된 저장 공간을 갖기 때문에 여유 공간 부족 시에 앞으로 사용자에게 의해 요청될 가능성이 가장 작은 비디오를 선택하여 제거한 후, 새로운 비디오를 저장해야만 한다. 따라서 본 논문에서는 그림 5과 같이 사용자의 선호도를 기반으로 한 캐쉬 교환 알고리즘을 제안한다. 각 비디오에 대한 Popularity, VPRN, 그리고 Video Prefix Selection은 프록시 서버의 성능에 영향을 주는 중요한 요소이므로 본 논문에서는 이 3가지 요소를 이용한다. 프록시 서버의 저장 공간 부족 시에 제거대상을 선정하는 기준으로 활용하기 위해 프록시 서버에 저장되어있는 비디오 데이터들의 Popularity를 Time t 동안 지속적으로 계산한다. 이렇게 계산된 비디오들 중 Popularity가 가장 낮은 비디오들을 제거 대상 후보로 선정한다. 이들 제거대상 후보들을 대상으로 VPRN을 참조하여 가장 작은 VPRN을 갖는 비디오를 제거대상으로 선정하고 프록시 서버에서 완전히 제거시킨다. VPRN은 사용자가 요청한 비디오가 프록시 서버에 저장되고 삭제되었던 빈도수를 의미하며, 전체 시간을 걸쳐서 특정 비디오가 프록시 서버에 추가와 삭제가 얼마만큼 발생하였는가를 의미한다. 특정 비디오에 대한 VPRN의 빈도수가 낮다는 의미는 앞으로 이 비디오가 프록시 서버에 다시 캐시될 가능성이 적다는 의미이며 그와 반대로 빈도수가 높은 비디오는 초기에 설정해 놓은 적절치 못한 Time t 로 인해 프록시 서버에 지속적으로 저장되지 못하고 있는 상태라고 할 수 있다. 따라서 같은 Popularity 빈도수를 갖는 다 하더라도 높은 VPRN을 갖는 비디오는 사용자에게 의해 보다 더 요구될 가능성이 높다고 할 수 있다. VPRN을 계산하는 전체 알고리즘은 표 1과 같다. 사용자는 특징적으로 전체 비디오에 있어서 주로 전반부인 Prefix만을 집중적으로 접근하는 경향이 있으며 짧은 접근 지속시간을 갖는 것이 특징이다. 프록시 서버의 캐쉬에서는 하나의 비디오가 원활한 전송을 위해서 다수의 패킷 형태로 존재하고 있다. 따라서 특정 비디오의 prefix를 구성하는 패킷 삭제 시에 접근 가능성이 상대적으로 낮은 suffix를 동시에 삭제시키고 그와 반대로 suffix를 구성하는 패킷 삭제 시에는 prefix 패킷들을 프록시 서버에 남겨둠으로써 보다 많은 저장 공간을 확보함과 동시에 사용자에게 요청될 가능성이 높은 비디오를 새롭게 저장할 수 있다.

표1. VPRN 계산 알고리즘

```

Algorithm Compute_VPRN
/* 사용자가 요청한 Video(Vx, Vy, Vz)가 초기 상태시에 프록시 서버에 저장되어있지 않으며 지속적으로 Popularity가 계산되어진다고 가정한다.*/
단계1. 사용자가 새로운 Video를 프록시 서버에게 요청한다.
      프록시 서버의 cache miss가 발생한다.
      Central Server에게 Vx 요청한다.
단계2. Central Server는 Vx를 프록시 서버에게 전송한다.
    
```

프록시 서버는 V_x 를 저장한다.
 이후 사용자가 V_x 요청 시 프록시 서버가 서비스 한다.
 단계3. 사용자가 새로운 Video V_y 를 프록시 서버에게 요청한다.
 프록시 서버의 cache miss가 발생한다.
 Central Server에게 V_y 요청한다.
 Proxy Server의 저장 공간 부족 현상(cache full)이 발생한다고 가정한다.
 Popularity를 기반으로 하여 V_x 가 제거 대상으로 선정되었을 경우, VPRN Log Table에 V_x 를 추가하고 Replacement Number를 증가시킨다.
 프록시 서버에서 V_x 를 완전 제거한다.
 새로운 Video인 V_y 를 프록시 서버에 저장한다
 단계4. 사용자가 새로운 Video V_z 를 프록시 서버에게 요청 한다.
 프록시 서버의 cache miss가 발생한다.
 Central Server에게 V_z 를 요청한다.
 Proxy Server의 저장 공간 부족 현상이 지속된다고 가정한다.
 제거대상 선정 시, Popularity와 함께 VPRN Log Table을 활용한다.

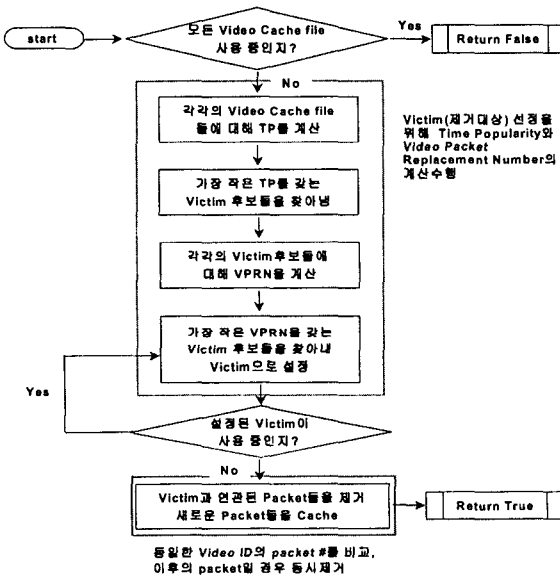


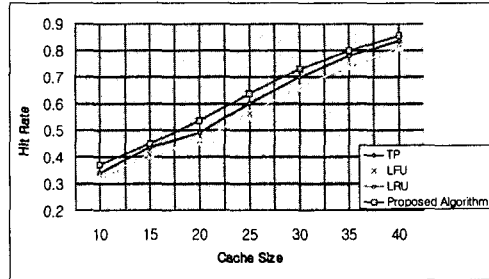
그림 4. 사용자 선호도 기반 캐쉬 교환 알고리즘

4. 성능평가

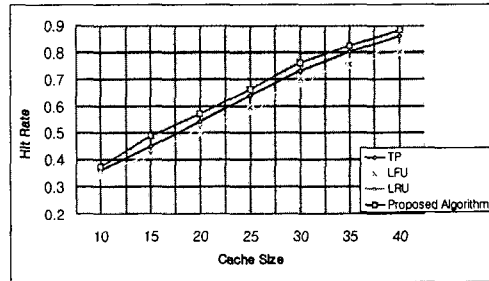
본 논문에서는 제한한 알고리즘의 성능 평가를 위해 Central Server와 열 명의 사용자가 연결된 프록시 서버를 각각 하나씩 두었으며 전송경로를 설정하기 위한 지연시간은 고려하지 않는다. 또한 전송 중에 발생할 수 있는 패킷 손실에 따른 재 전송 역시 고려하지 않는다. 프록시 서버에서 수행되는 교환 알고리즘의 성능을 평가하기 위해서 기존에 알고리즘인 Time Popularity, LFU 그리고 LRU를 대상으로 성능을 비교, 평가하였다. 표 2는 시뮬레이션을 수행하기 위한 파라미터들을 나타내고 있다. 프록시 서버의 캐쉬 크기를 증가시키면서 24시간동안 실시된 시뮬레이션의 결과를 비교해 보면 제안하는 알고리즘이 기존의 알고리즘에 비해 비교적 높은 Hit rate를 보이는 것을 알 수 있다. 높은 Hit rate를 나타낸다는 것은 사용자가 주로 요구하는 비디오가 프록시 서버에 미리 선별되어 저장되어 있어 사용자의 요구에 신속하고 안정적으로 서비스를 제공할 수 있다는 것을 의미한다. 그림 5은 캐쉬 크기 변화에 따른 각 알고리즘 별 Hit rate를 보여주고 있다.

표2. 시뮬레이션 파라미터

Simulation Time	24시간
Request Rate	5 Users per minute
Cache Size(GB)	10, 15, 20, 25, 30, 35, 40
Number of Videos	700 Files
Video Size	Approximately 70MB
Packet Size	1200 Byte
Access Pattern	Random Access Pattern Zipf Distribution($\theta = 0.3$) Pattern



(a) Random Access Pattern



(b) Zipf Distribution Pattern

그림 5. Cache Size 변화에 따른 각 알고리즘 별 Hit Rate

5. 결론

현재의 비디오 전송은 손실과 지연에 상당히 민감하게 반응하여 이를 기반으로 한 여러 분야의 발전에 커다란 문제점으로 지적되고 있다. 이와 같은 문제를 해결하기 위해 본 논문에서는 인터넷 VOD 서비스의 품질을 크게 개선시킬 수 있는 방법을 제공하였다. 본 논문에서는 제한된 저장 공간을 가지는 프록시 서버를 효율적으로 사용하기 위해서 사용자가 주로 요청하거나 또는 요청할 가능성이 있는 비디오만을 선별적으로 저장할 수 있는 캐쉬 교환 알고리즘을 제안하였다. 캐쉬의 크기를 변화시켜가며 수행한 각 알고리즘의 시뮬레이션을 통해 제안된 알고리즘은 기존의 알고리즘에 비해서 높은 Hit rate를 나타냄으로써 제안하는 알고리즘이 효율적임을 확인하였다.

참고 문헌

[1] Kuan-Sheng Hsueh and Sheng-De Wang, "A packet-based caching proxy with loss recovery for video streaming," *Proceedings of the 2003 Pacific Rim International Symposium on Dependable Computing*, pp. 185-190, Dec 2002.
 [2] Ewa Kusnierek, David Du and Yingfei Dong, "Proxy-Assisted periodic broadcast architecture for large-scale video streaming," *University of Minnesota, Tech. Rep.*, 2003.
 [3] S. Sen, J Rexford and Towsley, "Proxy Prefix Caching for Multimedia Streams, in *Proc. IEEE INFOCOM*, vol. 3, March 1999, pp. 1310-1319.