P2P 환경에 유효한 허위 데이터 감축 알고리즘

이원주°, 김승연*, 전창호**
두원공과대학 인터넷프로그래밍과°, 듀폰 포토마스크*, 한양대학교 컴퓨터공학과**,
wonjoo@doowon.ac.kr°, sykim@photomask.com*, chjeon@cse.hanyang.ac.kr**

An Efficient False Data Reduction Algorithm in P2P Environment

Wonjoo Lee°, Seungyun Kim*, Changho Jeon**
Department of Internet Programming, Doowon Technical College°,
DuPont Photomasks, Inc.*,
Department of Computer Science & Engineering, Hanyang University**

요약

현재 P2P(Peer-to-Peer) 환경에서는 많은 허위 데이터가 생성되면서 불필요한 네트워크 트래픽이 증가하는 문제점이 발생하고 있다. 본 논문에서는 이러한 문제점을 줄일 수 있는 FDR(False Data Reduction) 알고리즘을 제안한다. 이 알고리즘은 멀티미디어 콘텐츠 파일의 헤더 정보를 이용하여 데이터 파일에 고유 ID를 부여함으로써 허위 데이터 생성을 방지한다. 또한 허위 데이터로 확인된 데이터는 각 peer에 저장된 데이터 전송경로를 이용하여 허위 데이터를 삭제함으로써 허위 데이터 증가에 따른 네트워크 트래픽을 감소시킨다.

1. 서론

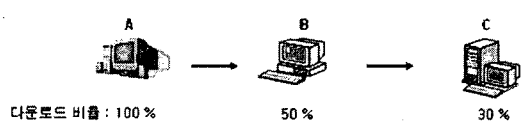
인터넷의 발전에 따라 급속히 성장한 P2P(Peer-to-Peer) 환경은 다량의 미디어 콘텐츠를 생성하고 활용하는데 기여하고 있다[1][2]. 하지만 P2P 환경에서는 콘텐츠의 진위 여부를 검증하는 것이 쉽지 않기 때문에 기존의 클라이언트/서버 환경에 비해 콘텐츠 관리에 어려운 점이 있다[3][4]. 특히 악의적인 사용자들은 P2P 응용 프로그램이 가지고 있는 허위 데이터에 대한 취약점을 이용하여 허위 데이터의 이름이나 설명을 변경하여 인기 있는 콘텐츠에 대한 허위 데이터들을 생성한다. 또한 콘텐츠 저작권을 가진 업체에서 저작권을 보호하기 위한 하나의 방법으로 허위 데이터를 생성하는 정책을 사용하기도 한다. 이러한 점은 유용한 콘텐츠를 찾는 P2P 사용자에게 혼란을 주며, 불필요한 네트워크 트래픽 증가의 원인이 된다. 그리고 위조된 콘텐츠는 다운로드를 완료하기 전에 P2P 사용자가 해당 콘텐츠의 진위 여부를 검증 할 수 없으며, 다운로드 받는 동안 다른 사용자에게 전송될 수 있다. 따라서 P2P 환경에서는 위조된 콘텐츠의 확산으로 인해 네트워크 트래픽이 증가하는 문제점이 있다.

본 논문에서는 P2P에서의 허위 데이터 확산에 따른 네트워크 트래픽 증가를 최소화 할 수 있는 알고리즘을 제안한다. 이 알고리즘의 특징은 새로운 데이터를 생성할 때마다 고유(unique) ID를 부여하여 허위 데이터 생성을 막는다. 또한 각 peer들이 다운로드 받은 콘텐츠에 대한 경로를 관리함으로써 해당 콘텐츠가 허위 데이터로 판정되면 그 경로를 이용하여 허위 데이터를 삭제한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 P2P 응용 프로그램을 소개하고 문제점을 설명한다. 3장에서는 제안한 알고리즘에 대하여 자세히 설명한다. 그리고 4장에서 성능평가에 대하여 설명하고, 결론을 맺는다.

2. 기존의 P2P 응응 프로그램의 문제점

일반적으로 P2P 응용 프로그램은 <그림 1>과 같이 데이터를 전송 할 때 다운로드를 종료하지 않아도 다른 peer에 전송할 수 있도록 설계되어 있다.



<그림 1> P2P 환경의 데이터 전송 모델

즉, peer B는 peer A로부터 데이터를 50% 다운로드 받은 상태에서 받은 데이터의 진위 여부를 판정하지 않고 peer C로 재전송한

다. peer B에서 데이터를 100% 다운로드한 후에 해당 데이터가 허위로 판정되면 서버에 보고한다. 기존의 P2P 응용 프로그램에서는 peer C로 전송중인 데이터를 중단할 수 없다. 이러한 P2P 환경의 데이터 전송 모델의 문제로 인해 허위 데이터의 확산은 증가한다. 또한 불필요한 네트워크 트래픽이 증가한다.

KaZzA, eDonkey, eMule[5], Overnet, Pruna[6] 등과 같은 기존의 P2P 응용 프로그램은 허위 데이터에 대한 대응책이 부족하다. 특히 Pruna는 허위 데이터에 대한 대응전략으로 허위 데이터 신고 기능을 사용하고 있다. 즉, P2P 사용자가 다운로드 받은 데이터가 허위 데이터로 판정되면 서버에 허위 데이터임을 신고하고, 다른 사용자들로 하여금 해당 데이터에 대한 허위 데이터 신고 수를 참조하게 함으로써 허위 데이터의 확산을 방지한다. 하지만 허위 데이터에 대한 정보가 사용자에게 실시간으로 제공되지 않기 때문에 사용자가 데이터를 검색한 이후에 허위 데이터로 판정되는 경우에는 허위 데이터 확산을 방지 할 수 없다는 단점이 있다. 그리고 허위 데이터로 확인되어도 그 데이터들은 P2P 시스템에 존재함에 따라 허위 데이터의 확산은 계속된다.

3. 제안하는 허위 데이터 감축 알고리즘

본 논문에서는 P2P 환경에서 허위 데이터의 확산으로 네트워크 트래픽의 증가를 최소화하는 FDR(False Data Reduction) 알고리즘을 제안한다. 이 알고리즘은 먼저 P2P 환경에서 생성되는 데이터에 고유 ID를 부여하여 허위 데이터의 생성을 방지한다. 또한 각 peer들이 다운로드 받은 콘텐츠에 대한 경로를 저장함으로써 해당 콘텐츠가 허위 데이터로 판정되면 그 경로를 이용하여 허위 데이터를 삭제한다.

3.1 고유(Unique) ID 부여

기존의 파일명 중심의 검색 방법을 사용하는 P2P 시스템에서는 사용자가 파일명을 변경하는 것은 매우 쉽기 때문에 허위 데이터를 생성하기가 용이하다. 따라서 기존의 파일명 중심의 검색 방법보다는 미디어 콘텐츠의 파일 헤더에 저장되어 있는 제목과 저자, 파일 크기 등의 정보를 이용하여 새로운 ID를 부여하고 그 ID를 검색하는 방법을 택한다. 이때 악의적인 사용자가 허위 데이터로 판명된 파일에 대해 다른 이름으로 다시 공유하는 것을 방지하기 위해 같은 파일에 대하여 동일한 ID를 부여하여 사용한다.

3.2 FDR(False Data Reduction) 알고리즘

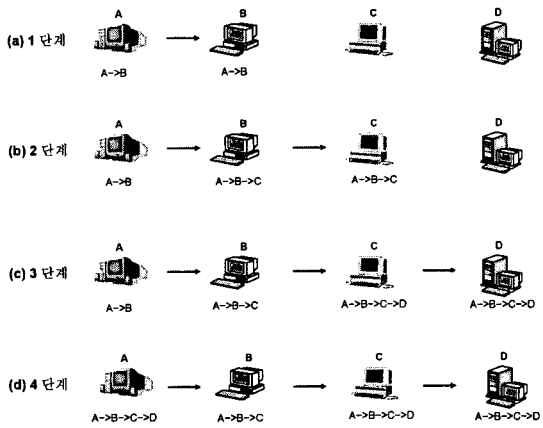
FDR 알고리즘에서는 여러 단계로 전송 중인 데이터에 대해서 허위로 확인된 데이터를 삭제하기 위해 데이터의 전송 경로를 유지한다. 데이터 전송경로 유지 알고리즘은 <그림2>와 같다.

```
// 데이터 송신경로
void file_sending (receiver_nodeID,contentID)
{
    receiving packet; //패킷 송신
    // 전송경로(data path) 설정
    aData.set_path(receiver_nodeID,contentID);
}

// 데이터 수신경로
void file_receiving (sender_nodeID,contentID)
{
    receiving packet();  //패킷 수신
    //패킷을 보내고 있는 peer의 전송경로(data path) 생성
    receiving_sender_data_path(sender_nodeID,contentID);
    //현재 수신중인 전송경로(data path) 설정
    aData.set_path(sender_nodeID,contentID);
    if (last_node)  //마지막 노드
    {
        //자신의 전송경로(data path)를 최초의 peer에 전송
        send_data_path_to_source
    }
}
```

<그림 2> 데이터 전송경로 유지 알고리즘

<그림 2>의 데이터 전송경로 유지 알고리즘에 대한 설명은 <그림 3>을 예로 들어 설명한다.



<그림 3> 데이터 전송경로 유지 예제

<그림 3>에서 4 개의 peer A, B, C, D가 존재한다고 가정하고 데이터는 A->B->C->D로 전송된다고 가정한다. (a)1단계에서 A->B로 데이터 전송이 시작되면 전송중인 데이터에 대해 고유 ID와 함께 전송 경로를 A와 B에 저장한다. 그리고 (b)2단계에서 C가 B에게 데이터 전송요청을 하면 송신하는 B와 수신하는 C의 경로를 서로 수정한다. (c)3단계에서도 (b)2단계와 같이 전송경로를 수정하고 다시 D가 C에게 전송을 요청하는 경우 각 C와 D의 전송경로를 수정한다. (d)4단계에서는 노드 실패에 대응할 수 있게 마지막 Peer D가 가지고 있는 전체 데이터 전송경로를 초기 Peer A에 복사함으로써 데이터 전송경로는 완성된다.

이 방법은 데이터 전송경로를 각 Peer에 저장 관리함으로써 전송중인 데이터가 허위 데이터로 확인되면 확인된 위치에서 데이터 전송경로의 역방향에 따라 각 Peer에서 해당 데이터를 삭제할 수 있다. 허위 데이터 삭제 알고리즘은 <그림 4>와 같다.

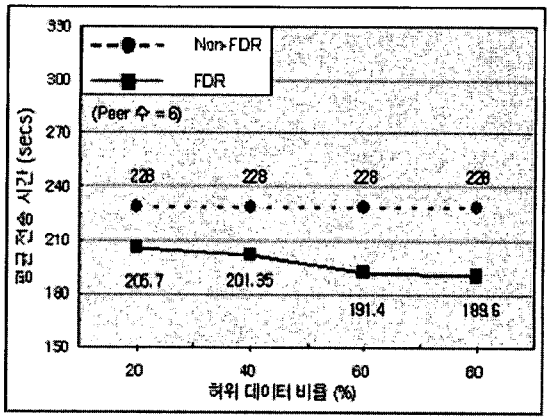
```
void delete_fake_file(contentID)
{
    aData.get_data_path();    //데이터 전송경로 생성.
    delete_file;              //파일 삭제
    if(no_exist_next_node)    //노드 존재 여부 판정
    {
        if(is_last_node){
            • 데이터 삭제 완료인 경우 모든 경로 삭제 및 종료;
        } else {
            if(no_first_node){ //첫번째 노드가 실패인 경우
                //데이터 경로를 따라 순방향으로 삭제
                delete_forward(aData.get_data_path());
            } else {
                //노드 실패일 경우 실패한 노드를 건너 뜀
                skip_node(aData.get_data_path());
            }
        }
    } else {  //정상적인 허위 데이터 삭제의 경우
        //저장된 전송경로(data path)의 역방향으로 데이터 삭제
        send_deleting_signal_to_previous_node(aData->
        get_data_path());
    }
}
```

<그림 4> 허위 데이터 삭제 알고리즘

노드 이탈(departure)과 노드 실패(failure)는 peer가 데이터 전송 중에 P2P 환경을 빠져 나가는 경우이다. 노드 이탈의 경우에는 peer가 정상적으로 P2P 시스템에 종료 사실을 알리고, 자신의 데이터 경로를 주변 peer에 복사한 후 P2P 응용 프로그램을 종료한다. 하지만 노드 실패의 경우에는 peer가 데이터 전송 경로와 종료 사실을 P2P 시스템에 전달하지 못하고 시스템에서 삭제된다. 이때 실패하기 전의 peer에 저장된 데이터 전송경로에 따라 허위 데이터를 삭제한다. 만약 첫 번째 peer가 실패하면 데이터 전송경로의 순방향으로 추적하여 허위 데이터를 삭제한다. <그림 3>을 예로 들어 허위 데이터 삭제 과정을 설명한다. (d)4 단계의 peer C에서 다운로드 받은 데이터가 허위 데이터로 확인되면 검색 서버에 신고하고 허위 데이터를 삭제한다. 그리고 데이터 전송경로에 따라 peer C로 데이터를 전송한 peer B에 허위 데이터임을 알린다. 따라서 peer B는 자신의 허위 데이터를 삭제한 후 데이터 전송경로에 따라 peer A에게 알린다. 이러한 과정으로 데이터 전송경로상에 나타난 모든 peer에 존재하는 허위 데이터를 삭제한다.

4. 성능 평가 및 결론

본 논문에서는 Network Simulator version 2(NS-2)를 사용하여 제안한 FDR 알고리즘의 성능을 평가하였다. 성능 평가 척도는 평균전송시간을 사용한다. 평균전송시간은 유효 데이터를 전송하는데 소요되는 평균시간이다. 허위 데이터의 비율에 따른 평균전송시간을 측정한 결과는 <그림 5>와 같다.



<그림 5> 유효 데이터 비율에 따른 평균전송시간

<그림 5>에서 FDR은 FDR 알고리즘을 적용한 결과이고Non-FDR은 FDR 알고리즘을 적용하지 않은 결과이다. 실험 결과를 살펴보면 허위 데이터 비율이 20인 경우 FDR의 평균전송시간은 Non-FDR에 비해 9.78% 감소하였다. 또한 허위 데이터 비율이 80인 경우 FDR의 평균전송시간은 Non-FDR에 비해 16.84% 감소했음을 알 수 있다. 따라서 허위 데이터 비율이 증가할수록 평균전송시간 면에서 FDR 알고리즘의 성능이 향상됨을 알 수 있다.

그러므로 본 논문에서 제안한 FDR 알고리즘은 미디어 콘텐츠 파일의 헤더 정보를 이용하여 데이터 파일에 고유 ID를 부여하여 허위 데이터 생성을 방지하고, 각 peer에 저장된 데이터 전송경로를 이용하여 허위 데이터를 삭제함으로써 P2P 환경에서 허위 데이터 증가에 따른 네트워크 트래픽을 감소시키는 효과가 있음을 알 수 있다.

5. 참고 문헌

[1] Ripeanu, M., "Peer-to-peer architecture case study: Gnutella network", Peer-to-Peer Computing, pp.99 – 100, 2001.

[2] Stoica, I., Morris, R., Karger, D., Kaasho, F. and Balakrishnan, H., "Chord: A scalable P2P lookup service for Internet applications", in ACM SIGCOMM, San Diego, CA, Assoc. Comp. Mach Press, New York, 2001.

[3] Ratnasamy, S. Francis, P., Handley, M., Karp, R., and Sheker, S., "A scalable content addressable network", in ACM SIGCOMM. Assoc. Comp. Mach. Press, New York, 2001.

[4] Clarke, I., Sandberg, O., Wiley, B., and Hong, T.W., "Freenet: A distributed anonymous information storage and retrieval system", in International Workshop on Designing Privacy Enhancing Technologies, Berkeley, CA. Springer-verlag, Heidelberg, 2000.

[5] http://www.emule-project.net/

[6] http://www.pruna.com/