

개인 맞춤형 댓글 필터링 Chrome 확장 프로그램 시스템 개발

유대겸, 고태희

(지도교수 : 이병국)

Development of a Personalized Comment Filtering System Using a Chrome Extension

Hee-tae Ko, Gum-dae You

요약

본 시스템은 사용자가 직접 정의한 금칙어를 기반으로 온라인 플랫폼의 댓글을 실시간 필터링하는 개인 맞춤형 크롬 확장 프로그램을 설계하고 구현하는 것을 목표로 한다. 최근 온라인 뉴스 및 동영상 플랫폼에서 악성 댓글, 혐오 표현, 정치적 편향성 발언이 빈번히 발생하고 있으며, 이는 사용자에게 심리적 피로감을 유발한다. 기존의 플랫폼 단위 필터링은 일관적인 정책 기반으로 운영되어, 개별 사용자의 민감도나 선호를 반영하기 어렵다. 이를 해결하기 위해 본 연구에서는 사용자가 Google OAuth로 로그인하여 개인 금칙어를 관리하고, FastAPI와 MySQL 기반 백엔드에서 이를 저장, 연동하는 시스템을 구축하였다. 프론트엔드는 Chrome 확장 프로그램 형태로 제작되었으며, MutationObserver를 이용하여 웹페이지 내 댓글의 변화를 실시간 감지하고, 금칙어가 포함된 댓글을 블러 처리하여 사용자가 직접 노출 여부를 제어할 수 있도록 하였다. 본 연구는 단순한 키워드 필터링을 넘어, 사용자 중심의 맞춤형 콘텐츠 관리라는 관점에서 온라인 댓글 환경의 건전성 제고에 기여한다.

Keyword: MutationObserver, Google OAuth, FastAPI, MySQL

I. 서 론

오늘날 온라인 플랫폼은 단순한 정보 공유 수단을 넘어, 사람들의 여론 형성과 사회적 담론이 이루어지는 주요한 공간으로 자리잡았다. 유튜브, 네이버 뉴스, 온라인 커뮤니티, SNS등의 댓글 영역은 사용자 간 의견 교류의 장이자 사회적 의사 표현의 통로로서 기능하고 있다.

그러나 이러한 익명성 기반의 댓글 환경은 건전한 의사소통의 장이라기 보다, 혐오 표현, 정치적 편향 발언, 근거 없는 비방과 공격이 빈번히 발생하는 공간으로 변질되고 있다. 한국리서치 “우리 국민의 온라인 댓글 인식 조사(2025)”에 따르면, 응답자의 70% 이상이 콘텐츠를 이용할 때 댓글을 함께 읽는다고 답했으며,

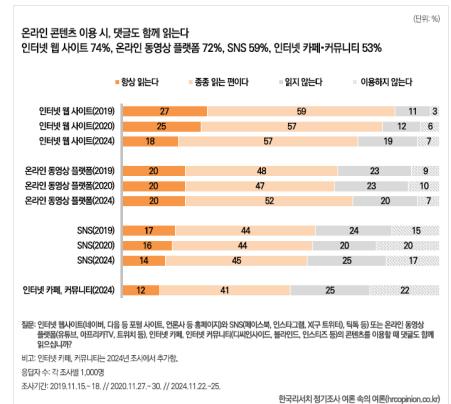


그림 1. 온라인 콘텐츠 이용 시 댓글 읽는 비율
그 중 약 60%가 “악성 댓글 문제는 심각하다”고 인식하고 있다.

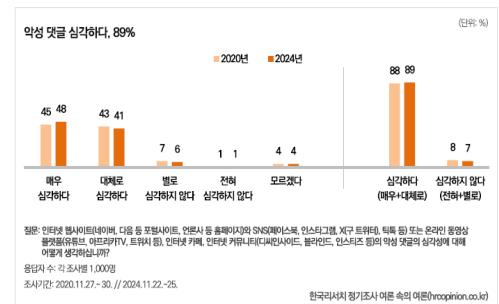


그림 2. 악성 댓글에 대한 인식

특히 정치, 연애, 사회 분야 기사에서는 부정적 감정이 과도하게 표출되는 현상이 두드러진다. 이러한 문제는 단순히 온라인 공간의 불쾌감에 그치지 않는다. 악성 댓글은 타인에게 정신적 피해를 주고, 사회 전반의 여론 형성을 왜곡시키며, 특정 집단에 대한 혐오를 조장한다. 또한 장기간 노출 시 사용자 스스로의 정서적 안정감을 해치고 플랫폼의 피로를 가중시킨다. 결국 온라인 이용자에게는 댓글을 보고 싶지 않지만 차단할 수 없는 딜레마가 발생하게 된다.

II. 기존 시스템의 한계

현재 대부분의 플랫폼(유튜브, 네이버 등)은 커뮤니티 가이드라인에 따라 자체 필터링이나 신고 시스템을 운영하고 있다. 또한 Google의 Perspective API와 같은 AI 기반 자동 필터링 기술도 존재한다. 하지만 이러한 시스템에는 몇 가지 공통적인 한계가 있다.

첫째, 플랫폼 중심의 일괄 필터링 구조다. 욕설이나 비방 표현은 어느정도 걸러지지만, 플랫폼이 정한 기준에 따라 일관적으로 작동하기 때문에 사용자 개개인의 감정적 기준이나 민감도 차이를 반영하지 못한다. 예를 들어, 정치 기사에 대한 특정 키워드나 인물 이름에 피로감을 느끼는 사용자가 있더라도 그 단어를 회피할 방법이 없다.

둘째, 사용자 제어권의 부재다.

기존 시스템에서는 사용자가 직접 “왜 특정 댓글이 보이지 않는지”, 또는 “어떤 단

어 때문에 필터링되었는지”를 명확히 알 수 없다. 결과적으로 사용자는 플랫폼이 정한 불투명한 기준에 따라 콘텐츠를 수동적으로 소비하게 된다.

이처럼 구조적 한계로 인해, 사용자가 자신만의 기준으로 댓글 노출을 조정할 수 있는 개인 맞춤형 필터링 시스템의 필요성이 커지고 있다.

III. 프로젝트의 목적 및 필요성

이 프로젝트는 이러한 문제를 해결하기 위해 사용자 주도형 댓글 필터링 시스템을 설계하고 구현하는 것을 목적으로 한다.

기존 일관된 기업들의 정책 필터링에서 벗어나, 개별 사용자가 자신의 판단에 따라 필터링 기준을 정의하고 즉시 반영할 수 있는 구조를 구축하는 것이다.

1. 사용자의 자율성의 보장

사용자가 자신의 온라인 경험을 스스로 통제할 수 있도록 함으로써, 플랫폼이 결정하는 필터링에서 사용자가 직접 결정하는 필터링으로 하여금 사용자 경험을 향상시킨다.

2. 정신적 피로 감소와 정보 선택권 강화
불쾌하거나 피로감을 유발하는 단어가 포함된 댓글을 보지 않음으로써 사용자는 필요가 없는 정보로 부터 스스로를 보호하고, 동시에 다른 의견이나 건전한 댓글에는 접근성을 유지 할 수 있다.

3. 기술적 접근의 현실성

Chrome 확장 프로그램을 기반으로 구현함으로써, 사용자는 크롬 웹스토어에 간단하게 설치하여 손쉽게 댓글 환경을 관리할 수 있다. 또한 백엔드 서버와 데이터베이

스를 통해 사용자별 금칙어를 저장하고 즉시 반영함으로써 실시간 개인화 환경을 구현할 수 있다.

IV. 본론

이 장에서는 본 프로젝트에서 설계한 개인 맞춤형 댓글 필터링 시스템의 구체적인 구성 요소와 각 파일들의 역할, 그리고 전체적인 동작 과정을 기술한다.

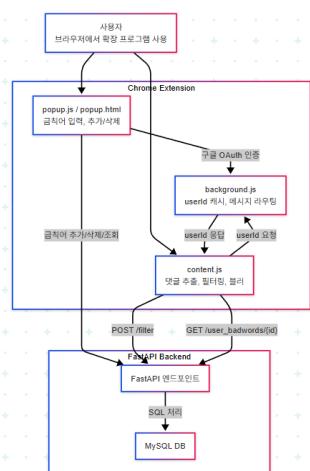


그림 3. 시스템 아키텍처

1. 전체 시스템 구조

시스템의 전반적인 구조는 사용자의 브라우저 환경에서 동작하는 Chrome 확장 프로그램과 이를 지원하는 백엔드 서버, 그리고 정보를 저장하는 데이터베이스로 이루어져 있다.

Chrome 확장 프로그램은 사용자가 직접 금칙어를 등록하고 관리할 수 있는 인터페이스를 제공하며, 백엔드 서버와 실시간으로 통신하여 해당 정보를 데이터베이스에 반영한다. 사용자가 웹사이트에 접속하면 확장 프로그램은 댓글 영역을 탐지하고, 등록된 금칙어가 포함된 댓글을 자동으로 블러 처리하여 표시한다.

2. 백엔드 서버 구현

백엔드 서버는 Python FastAPI 프레임워크를 기반으로 구축되었다. FastAPI는 비동기 요청을 효율적으로 처리할 수 있는 경량 웹 프레임워크로, RESTful API 구조를 통해 확장 프로그램과 안정적으로 통신 한다.

서버의 핵심 파일인 main.py는 전체 API의 진입점으로, 사용자 금칙어 관리와 댓글 필터링 기능을 담당한다. 주요 기능으로는 사용자 금칙어 목록을 불러오는 /user_badwords/{user_id} 경로, 금칙어를 추가하는 /user_badwords 경로, 삭제 기능을 제공하는 /user_badwords/{user_id}/{word} 경로 그리고 댓글 목록 중 금칙어가 포함된 항목만 판별하는 /filter 경로가 있다.

이 모든 API는 SQLAlchemy를 통해 MySQL 데이터베이스와 연결되며 Pydantic 모델을 사용하여 요청 데이터 유효성을 검증한다. 또한, CORS(Cross-Origin-Resource Sharing) 설정을 적용하여 Chrome 확장 프로그램이 외부 API로 안전하게 요청을 보낼 수 있도록 하였다. 백엔드 서버는 FastAPI의 비동기 구조를 통해 클라이언트 요청에 빠르게 응답하며, 사용자가 금칙어를 추가하거나 삭제할 때 UI에 실시간으로 반영될 수 있도록 동작한다. 서버는 단순히 데이터 전달만 수행하는 것이 아니라, 사용자별 Google 계정 ID를 기준으로 데이터를 분리하여 관리함으로써 보안성과 사용자 맞춤성을 동시에 확보하였다.

3. 데이터베이스 설계

데이터베이스는 MySQL을 기반으로 구축

되었으며, 초기 설정은 init.sql파일을 통해 수행된다. 데이터베이스 이름은 commentfilter로 지정되었으며, 주요 테이블은 user_bad_words 하나로 구성된다. 이 테이블에는 사용자의 고유 ID(user_id), 금칙어(word), 분류(category), 생성일(created_at) 필드가 포함되어 있다. 특히 동일한 사용자가 같은 단어를 중복 등록하지 못하도록 (user_id, word)조합에 UNIQUE제약조건을 적용하였다. 이를 통해 데이터 중복을 방지하고 사용자별 데이터 무결성을 보장한다.

데이터베이스는 FastAPI서버와 직접 연동되어 있으며 금칙어 추가, 삭제, 조회 요청 시마다 실시간으로 데이터를 주고받는다. 또한 SQLAlchemy를 통해 쿼리 실행을 안전하게 수행하도록 설계되어, SQL 인젝션과 같은 보안 취약점에 대한 위험도 최소화 하였다.

4. 프론트엔드 Chrome 확장프로그램 구현
이 프로젝트의 인터페이스는 Chrome 확장 프로그램으로 구현되었다. 확장 프로그램은 브라우저 내부에서 직접 실행되며, 사용자의 로그인 인증, 금칙어 등록, 실시간 댓글 감시 및 블러 처리 기능을 담당한다. 확장 프로그램은 다음 네 가지 주요 파일로 구성되어 있다.

(1) manifest.json

확장 프로그램의 기본 설정 파일로, 이름, 버전, 권한, 실행 스크립트, OAuth 2.0 클라이언트 ID등의 정보를 정의한다. 이 파일은 Chrome 환경에서 프로그램의 실행 구조를 결정하는 핵심 메타데이터 역할을 한다.

다. 이 시스템에서는 Google 로그인 인증을 위해 identity 및 identity.email 권한을 명시하였다.

(2) background.js

확장 프로그램의 백그라운드 프로세스로서, 전체 서비스의 중앙 관리자 역할을 수행한다. Google OAuth 인증을 통해 로그인한 사용자의 ID를 가져와 저장하고, popup.js 와 content.js 간의 메시지를 중계한다. 즉, 사용자의 로그인 상태를 유지하면서 비동기적으로 정보를 전달하는 허브 기능을 수행한다.

(3) popup.html/popup.js

이 두 파일은 UI부분을 담당한다. popup.html은 사용자가 금칙어를 입력하고 확인할 수 있는 화면을 구성하며, popup.js는 사용자의 입력 이벤트를 처리하여 FastAPI서버로 데이터를 전송한다. 사용자가 금칙어를 추가하면 서버의 /user_badwords API로 전달되어 데이터베이스에 저장되며, 삭제 시에는 /user_badwords/{user_id}/{word} 경로를 호출한다. 응답 결과는 즉시 화면의 목록에 반영되어 실시간으로 갱신된다.

(4) content.js

content.js는 이 프로젝트의 핵심 로직이 포함된 파일이다. 사용자가 접속한 웹페이지 내에서 댓글 영역을 감시하고, 등록된 금칙어가 포함된 댓글을 블러 처리한다. 이를 위해 Chrome의 MutationObserver API를 사용하여 DOM(Document Object Model)의 변화를 실시간으로 탐지한다. 새로운 댓글이 로드되거나 수정되면, content.j

s는 벡엔드에서 금칙어 목록을 받아와 해당 단어가 포함된 댓글에 시작적 블러 효과를 적용한다. 이 로직은 완전히 클라이언트 측에서 수행되므로 네트워크 지연 없이 즉각적인 필터링 결과를 제공한다.

5. 시스템 동작 과정

본 시스템의 전체 동작 흐름은 다음과 같다.

- (1). 사용자가 Chrome 확장 프로그램을 설치하고 실행한다.
- (2). 프로그램이 Google OAuth 2.0을 통해 로그인된 Chrome 계정 정보를 가져온다.
- (3). 사용자는 팝업 화면을 통해 금칙어를 등록하거나 삭제한다.
- (4). FastAPI 서버는 해당 금칙어를 MySQL 데이터베이스에 저장한다.
- (5). 사용자가 유튜브 또는 네이버 뉴스 폐이지를 열면 content.js가 댓글 영역의 변화를 실시간으로 감지한다.
- (6). 데이터베이스에 저장된 금칙어 목록을 불러와, 해당 단어가 포함된 댓글을 블러 처리한다.
- (7). 사용자는 불쾌한 댓글을 보지 않고, 원할 경우 블러된 부분을 해제하여 확인할 수 있다.

이와 같은 흐름을 통해 사용자는 별도 프로그램 없이 브라우저 내부에서 자신의 온라인 경험을 자유롭게 제어할 수 있다. 또한, 시스템은 사용자별 데이터를 분리 저장하기 때문에 다수의 사용자가 동시에 이용하더라도 서로의 금칙어 설정이 충돌하지 않는다.

6. 설계상의 특징 및 장점

본 프로젝트에서 구현한 시스템은 다음과 같은 기술적 특징을 가진다.

실시간성: MutationObserver를 통한 DOM 감시로 댓글이 추가되자마자 필터링이 수행된다.

개인화: Google ID를 이용하여 사용자별 필터링 규칙이 유지된다.

경량성: FastAPI와 Chrome 확장 프로그램 구조를 통해 높은 처리 속도와 낮은 메모리 사용을 실현하였다.

보안성: OAuth 인증과 SQLAlchemy의 파라미터 바인딩으로 보안 취약점을 최소화 하였다.

확장성: 유튜브, 네이버 뉴스 외에도 동일한 구조를 다양한 플랫폼 확장이 가능하다.

V 결론

본 프로젝트에서는 온라인 상에서 빈번하게 발생하는 악성 댓글, 혐오 표현, 정치적 편향 발언 등으로 인해 사용자들이 겪는 정신적 피로와 부정적 감정 노출 문제를 완화하기 위해 Chrome 확장 프로그램 기반의 개인 맞춤형 댓글 필터링 시스템을 설계하고 구현하였다. 기존의 댓글 관리 시스템은 플랫폼 사업자가 일방적으로 정한 규칙에 의존하고, AI 기반 필터링은 정지적 인물과 관련된 단어, 지역적 혐오발언 등 기업의 사정상, 규정상 처리하지 못하는 한계가 존재한다. 이 프로젝트는 이러한 한계를 해결하고자 사용자가 스스로 필터링 기준을 정의하고, 자신의 계정 정보를 기반으로 맞춤형 필터링을 수행할 수

있는 시스템을 제안하였다.

시스템은 Chrome 확장 프로그램, FastAPI 서버, MySQL 데이터베이스로 구성되어 있으며, Google OAuth 2.0 을 통해 사용자별 고유 식별 정보를 생성하고 이를 기반으로 금칙어를 개별 관리할 수 있도록 구현하였다. 사용자가 팝업 화면을 통해 금칙어를 등록하면 FastAPI서버가 해당 데이터를 MySQL데이터베이스에 저장하고, 확장 프로그램 내에서는 content.js 파일을 통해 웹페이지의 댓글 영역을 실시간으로 감시한다. 이후 MutationObserver API를 이용하여 댓글의 DOM 구조 변화를 탐지하고, 등록된 금칙어가 포함된 댓글에 대해 블러 효과를 적용함으로써 사용자가 불쾌한 내용을 직접 마주하지 않도록 한다. 이러한 전 과정은 비동기 통신으로 이루어져 있으며, 사용자 입력이 실시간으로 UI와 데이터베이스에 반영되어 즉각적인 반응성을 보인다.

테스트 결과, 유튜브와 네이버 뉴스 플랫폼에서의 테스트에서 평균 필터링 속도는 0.15초 내외로 측정 되었고 댓글 100개 중 금칙어 포함 댓글을 정확히 감지하여 블러 처리하는 등 높은 정확도를 보였다. 또한 100개 이상의 금칙어가 데이터베이스에 저장된 상태에서도 시스템의 속도 저하나 오류가 발생하지 않았고 Google OAuth로 그인 세션이 안정적으로 유지되었다. 이를 통해 본 시스템은 실시간성, 안정성, 그리고 사용자 맞춤성 측면에서 모두 높은 완성도를 입증하였다.

이 프로젝트에서 개발한 시스템은 단순히

댓글을 가리는 기능에 그치지 않고, 사용자가 자신의 온라인 경험을 직접 설계할 수 있도록 한다는 점에서 의의가 있다. 사용자는 자신이 불쾌함을 느끼는 특정 단어나 주제를 스스로 차단함으로써 정신적 피로를 줄일 수 있고, 동시에 보고 싶은 정보는 유지할 수 있는 선택적 회피 환경을 구현하였다. 이는 댓글 전체를 비활성화하는 기존 방식보다 정보 접근성과 사용자 경험을 동시에 보장한다는 점에서 차별화된다. 또한, 청소년이나 일반 사용자들이 혐오 표현이나 자극적인 콘텐츠로부터 자신을 보호할 수 있도록 돋는 사회적 가치도 지닌다.

기술적 측면에서도 Google OAuth 인증, 비동기 FastAPI서버, MutationObserver 기반 실시간 필터링, SQLAlchemy를 활용한 안전한 데이터 연동 등 웹 기술을 통합적으로 활용하였다.

향후 프로젝트에서는 다양한 웹사이트 추가와 댓글뿐만 아니라 혐오성 광고나 불쾌한 이미지 콘텐츠까지 사용자가 선택적으로 차단할 수 있는 기능으로 확장함으로써, 사용자의 디지털 피로를 보다 폭넓게 완화 할 수 있을 것이라고 기대된다. 이를 위해 향후에는 Docker 환경을 활용한 서버 배포와 클라우드 기반 서비스 운영을 통해 시스템의 확장성과 접근성을 더욱 향상시킬 예정이다.

결론적으로 이 프로젝트는 기존의 플랫폼 중심적 댓글 관리 구조를 벗어나, 사용자가 직접 자신만의 필터링 기준을 정의하고 제어할 수 있는 개인 맞춤형 시스템을 구

축합으로써 온라인 댓글 환경의 자율성과 다양성을 확대하였다. 이 시스템은 기술적 완성도뿐만 아니라 사회적 활용 가능성 면에서도 높은 실용적 가치를 지니며 보다 건강한 온라인 소통 문화를 조성하는 데 기여할 것으로 기대된다.

References:

[1] 한국리서치, 「우리 국민의 온라인 댓글에 대한 인식은?」, 2025년 2월 12일

출처:

<https://hrcopinion.co.kr/archives/32142>

[2] Quickz, "YouTube Comment Filter (Chrome Extension) GitHub Repository," GitHub, 2025년

출처:

<https://github.com/Quickz/youtube-comment-filter>