creative commons
COMMONS DEED

저작자표시-비영리-동일조건변경허락 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.
- 이차적 저작물을 작성할 수 있습니다.

다음과 같은 조건을 따라야 합니다:

저작자표시. 귀하는 원저작자를 표시하여야 합니다.

비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.

동일조건변경허락. 귀하가 이 저작물을 개작, 변형 또는 가공했을 경우에는, 이 저작물과 동일한 이용허락조건하에서만 배포할 수 있습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer

dCollection

공학석사 학위논문

Social Network Service의 Data 위주 연구를 위한 Platform 설계

2012년 8월

서울대학교 대학원

디지털정보융합학과

황 용 태

Social Network Service의 Data 위주 연구를 위한 Platform 설계

지도교수 이 교 구

이 논문을 공학석사 학위논문으로 제출함

2012년 8월

서울대학교 대학원

디지털정보융합학과

황 용 태

황용태의 석사학위논문을 인준함

2012년 7월

위 원 장 강 남 준 (인)

부 위 원 장 이 교 구 (인)

위 원 이 준 환 (인)

국문초록

SNS(Social Network Service) 분야의 발전은 'Web 2.0' 시대와 Mobile 기기의 발달로부터 유발되었다. SNS의 수억 명에 달하는 사용자들이 실시간으로 만들어 내는 데이터를 학계에서 주목하여, 최근까지 다수의 연구가 진행되어 왔다. 이 데이터는 서비스 제공자가 공개한 API(Application Programming Interface)를 통해 접근할 수 있으며, 대부분의 연구에서는 데이터의 방대한 양 때문에 프로그램(Crawler)을 통한 데이터 수집이 필요하다.

SNS에 대한 연구는 기본적으로 융합적인 특성을 가지고 있다. 하지만 연구의 초기에는 소규모의 연구자가 연구를 진행하며, 해당 인원이 가능한 범위에서 데이터 수집과 분석이 진행된다. 따라서 연구 문제 설정 단계에서 수작업으로 데이터 처리를 진행하는 경우가 많으며, 양질의 대량 데이터를 수집할 수 없게 된다. 이는 연구 과정에서의 시행착오와 재 작업으로 나타난다.

본 연구에서는 이러한 문제점을 해결하기 위하여, 사용이 용이하고 데이터 위주의 연구를 도와 줄 수 있는 SNS 자료 수집-분석 Platform을 설계, 작성하였다. 먼저 기존에 수행되었던 연구들에서 어떤 데이터를 사용했고 수집 방법은 어떠하였는지 분석하였다. 다음으로 SNS에서 제공하는 API 동작과 데이터 형식을 분석하고 모델링 하였으며, SNS를 대상으로 하는 데이터 위주 연구를 수행하면서 발생할 수 있는 문제점과 극복 방안을 살펴보았다.

앞 단계의 내용을 토대로 여러 서비스에 대응 가능하며 확장성을 가진 데이터 수집-분석 Platform이 작성되었다. 이를 통해 SNS 대상 연구 과정에서 중복되는 공통 절차를 간소화 할 수 있으며, 앞에서 제시한 연구상의 문제점도 어느 정도 해결

하였다. 작성된 Platform으로 실제 연구 Case에 적용 가능한 과제를 구성하여, 의도한 데이터가 수집되는지 분석하였다.

주요어 : 소셜 네트워크 서비스, 응용 프로그램 프로그래밍 인터페이스, 크롤러, 플랫폼

학번 : 2010-22686

목차

표 목차

그림 목차

제 1 장　서론

이른바 'Web 2.0' 시대는 기존의 찾고-읽는 인터넷을 상호작용하는 환경으로 바꿔 놓았다 [1]. 또한 모바일 기기의 발달은 인터넷 사용의 시공간적 제약을 많이 줄여 주었다. 이러한 변화의 수혜를 받은 서비스의 종류에는 여러 가지가 있겠으나, 그 중에 Social Network Service(이하, SNS)는 가장 큰 발전을 이룬 분야라고 할 수 있겠다. Facebook의 경우 전세계 사용자는 2012년 현재 약 9억명 정도이며 [2], Twitter의 경우　5억명 수준이다 [3]. 이는 중복을 고려하더라도 약 20억명에 이르는 세계 인터넷 사용자 [4] 중 50%에 달하는 비율로 SNS를 이용하는 것이다.

SNS를 이용하는 수많은 사용자들은 활동하면서 많은 데이터를 실시간으로 만들어 내고 있다. 이러한 데이터는 처리되지 않은 (Raw) 것으로, 수량은 많으나 그 자체로 의미를 가지기는 힘들다. 학계에서는 이 대량 데이터에 주목하여, 여러 분석을 통해 유의미한 결론을 내려는 연구를 계속 해 왔다.

이러한 연구가 가능한 이유는, 서비스 제공자가 공개된 API(Application Programming Interface)를 통해 정보를 읽어올 수 있도록 하였기 때문이다. 물론 API의 형식으로 데이터가 공개되지 않았더라도 기존의 웹 탐색 방법처럼, 사용자에게 보여지는 내용을 분석하여 데이터를 수집할 수 있다. 하지만 제공되는 API를 이용하면 안정적이고 서비스 구조에 맞는 데이터를 얻어올 수 있다.

연구자는 API를 사용하여 데이터를 수집하고, 수집된 데이터로부터 통계적인 처리와 분석을 통해 결론을 내게 된다. 이 중 데이터를 수집하는 과정은 보통 수집기 프로그램(Crawler라고 부르는)에 의해 자동적으로 수행하며, 처리와 분석 과정에서는 연구자의 개입이 필요하다.

SNS를 대상으로 진행되는 연구의 다수는 현상을 인문-사회학적으로 분석하고 있으며, 주 연구자는 인문-사회학적 배경을 가지는 경우가 많다. 하지만 대량의 데이터를 다루기 위해서는 API와 데이터 형식에 대해 이해한 다음, 프로그램을 작성하고 실행해야 하므로 해당 연구자 그룹 내에서 모든 요구 사항을 해결하기 어렵다. 따라서 프로그래밍과 데이터 분석에 능력을 가진 타 분야의 연구자와 협업하는 것이 필수적이다.

원활한 연구를 위해서는 각 분야 연구자가 담당 업무만 진행하는 것이 아니라, 충분한 이해와 긴밀한 협조를 해야 한다는 점에서, SNS를 대상으로 하는 연구는 융합적인 특성을 띤다고 할 수 있다. 인문-사회학 배경의 연구자는 서비스로부터 사용 가능한 동작과 데이터에 대해 알고 있어야 연구 주제를 설정할 수 있으며, 데이터 수집과 분석을 담당한 연구자는 연구의 주제와 목적, 이론적인 배경이 무엇인지 알아야 한다.

하지만 연구자 상호간의 충분한 이해는 쉽지 않으며, 이런 차이는 연구 진행 도중 여러 문제를 발생시킨다. 실제로 본인과 동료가 2010년부터 Twitter를 비롯한 여러 SNS 연구 [5], [6]를 수행하면서, 연구 주제에 대한 많은 의문 사항과 API에 대한 문의, 데이터 수집과 분석 과정에서의 반복적인 작업을 경험했다. 또한 서로 다른 서비스에 대한 데이터 수집이라 하더라도 공통된 부분이 존재함을 알 수 있었다. 본 논문은 어떻게 하면 SNS 연구 과정에서의 시행착오와 반복 과정을 줄일 수 있을까를 고민하면서 시작되었다.

본 논문에서는 먼저 SNS를 대상으로 하는 데이터 위주의 연구를 수행하면서 발생할 수 있는 문제점과 극복 방안을 살펴보도록 한다. 기존에 수행되었던 연구에서 어떤 데이터를 사용했고 수집 방법은 어떠하였는지, 실제 서비스에서는 어떤 동작과 데이터 모델을 사용하는지 분석한다. 이 내용을 토대로, 연구자를 주 사용층으로

하는 SNS 데이터 수집-분석 도구를 설계, 제작한다. 도구의 설계 과정에서는 앞에서 분석한 SNS 데이터 처리 방법 상의 이슈들이 반영되며, 사용의 편의성을 위해 적절한 사용자 인터페이스를 제공하도록 한다.

제 2 장　관련 연구

제 1 절　기술적 요소

1. API와 OpenAPI

API (Application Programming Interface)란 프로그램이 특정 서비스나 자원에 접근하기 위해 따라야 하는 규약이다. 컴퓨터 프로그래밍에서 API는 기존에 작성된 라이브러리나 시스템 기능에 접근하기 위해 필수적인 것으로, 오래 전부터 있어왔던 개념이다. [7]

웹 환경에서 API라는 용어는 특히 웹 서비스와 같은 개념으로 사용된다. 일반적으로 사용자와 서버 사이에는 HTTP(Hypertext Transfer Protocol) 방식을 사용해서 메시지를 주고받으며, 메시지의 내용은 구조상으로 XML(Extensible Markup Language)이나 JSON(JavaScript Object Notification), 형식적으로는 SOAP(Simple Object Access Protocol)를 따르는 경우가 많다 [8]. 이 역시 표현은 다소 변화하고 있으나 기본적인 기준은 이미 90년대 후반에 규정된 것이다.

새롭고 주목할만한 주제는 이른바 'Open API'라고 불리는 것이다. 'Open'이라 함은 비교적 제한 없이 웹 서비스의 기능과 데이터를 다른 프로그램 또는 서비스에서 이용할 수 있다는 것을 의미한다. 이러한 경향은 Web 2.0 개념의 유행과 함께 퍼져나갔다. Open API의 장점은 두 개 이상의 서비스를 합치거나 기존 서비스에 다른 기능을 추가하여 새로운 서비스로 만들 수 있다는 것이다. 대표적인 예로 Google Maps의 API와 Craigslist의 부동산 정보를 합쳐 지도에서 부동산 정보를 보여주도록 한 HousingMaps.com 이 있으며, 이 외에도 수없이 많은 'Mashup' 서비스가

만들어졌다 [9]. 기술적으로는 REST(Representational State Transfer)라고 하여, 한 가지 데이터를 필요에 따라 여러 가지 형태의 문서로 나타낼 수 있는 것을 특징으로 하는 형식의 서비스 구조를 따르는 경우가 많다 [8].

2. (Web) Crawler

(Web) Crawler는 프로그램의 일종으로, 인터넷에서 정보를 수집하고 처리/분류하는 프로그램을 총칭한다. 보통 자동적인 방법으로 데이터를 수집하며, 대표적인 것으로 검색엔진의 데이터를 수집하기 위해 웹을 방문하는 '검색 로봇'이 있다. 인터넷의 경우 하이퍼링크가 존재하여 한 페이지에서 다른 여러 페이지를 가리키기 때문에, 처음 'crawling'을 시작한 페이지로부터 수많은 페이지로 검색 대상이 늘어난다. [10] 이러한 특성을 이용하여 네트워크의 전체적인 구조를 파악할 수 있으나, 탐색에 많은 시간을 소요하므로 대상이나 전략, 제약 조건을 주의 깊게 설정할 필요가 있다. Crawler를 이용한 연구의 예로는 링크 구조에 따른 인터넷 문서의 연결 관계 네트워크 분석이 있다.

Web Crawler의 경우 일반적인 인터넷을 대상으로 하며 웹 문서에 특정한 구조가 없는 경우가 많으므로, Crawler의 목적에 따라 내용의 처리 구현이 각각 달라진다. 그러나 Web Service를 이용한 Crawler의 경우 서비스에서 데이터 형식을 규정하기 때문에 이 형식에 맞는 처리와 탐색을 수행하게 된다.

3. Social Network Service의 특징

Social Network Service(SNS)는 사람들 사이의 사회적인 관계를 형성하고 상호작용할 수 있도록 만들어진 서비스를 의미한다. 컴퓨터 환경에서 이러한 서비스의

역사는 오래 되었으며, 그 범위는 전자 메일이나 Instant Messenger로부터 게시판(BBS; Bulletin Board Service), 대화방까지 다양하다.

최근에 사용되는 의미로서의 SNS는 몇 가지 공통적인 특성을 가지고 있다. 사용자는 서비스에 가입하여 개인 프로파일을 생성하고 몇 가지 개인 정보와 소개를 입력한다. 서비스에는 여러 종류의 컨텐츠를 올릴 수 있으며 주 컨텐츠는 서비스에 따라 글, 사진, 동영상 등 다양하다. 사용자는 기존에 오프라인에서 아는 사람이나 온라인 상의 흥미로운 사람을 대상으로 '친구' 관계를 형성하며, 친구들에 대한 목록을 설정하고 관리할 수 있다. 컴퓨터를 매개로 한 커뮤니케이션의 특성 상 상호관계에서 시공간적인 제약을 축소시키며, 이러한 특징은 Mobile 기술의 발달에 따라 더욱 가속화 되었다.

근래에 두드러지고 있는 SNS의 특징은 실시간성과 위치 기반이다. Mobile 플랫폼과 SNS Application, GPS 내장과 같은 기술이 이러한 특징을 구현 가능하게 하였다. 사용자는 자신이 현재 느끼는 감정이나 주변 상황, 사건에 대해 바로 업로드 할 수 있다. 이런 특징이 단적으로 나타난 예는 2009년 미국 항공기 허드슨강 불시착 사건으로, Twitter의 소식 전달 속도가 언론사보다 빨랐다 [11]. 또한 위치를 기반으로 하는 서비스는 사용자가 현재 어디 있는지를 나타낼 뿐만 아니라, 상점의 맞춤 마케팅이나 주변 다른 사용자와의 상호작용을 가능하게 하였다. 이러한 'Check-in' SNS의 대표적인 예로 Foursquare가 있다. [12]

제 2 절 기존 연구 사례

오늘날 일반적으로 생각할 수 있는 형태의 SNS가 제공된 지는 역사가 그리 길지 않다. 하지만 현재 서비스 중인 SNS의 종류만 해도 수십 가지가 되며, 이 중 사용자의 이용도가 높으며 전세계적인 이슈를 만들어내는 Facebook, Twitter 등의 서

비스에 대해서는 많은 연구가 이루어졌다. 여기서는 이들 중 Twitter에 대해 이루어진 연구를 살펴보도록 한다.

1. Twitter의 특징

Twitter의 가장 큰 특징은 주 컨텐츠인 글의 길이가 140자로 제한된다는 것이다. 이러한 제한은 단문 문자 메시지(SMS) 시스템 때문에 발생되었으며, 시스템적인 제약이 Twitter 컨텐츠의 고유한 특성(짧은 문장 안에 중요한 정보를 포함, 시의적절성)을 만들어내기도 하였다 [13].

사용자는 관심 있는 다른 사용자를 'follow'할 수 있으며, 이렇게 하면 그 사용자가 작성한 글이 자신의 Timeline(기본 컨텐츠 보기 화면)에 나타난다. 본인을 다른 사용자가 follow 한 경우 그 사용자는 'follower'라고 부른다. Follow와 following 관계는 대칭적일 필요가 없으며, 이 또한 Twitter 만의 특징을 만든다. 대표적인 예로 follower 2천7백만에 달하는 Lady Gaga와 같이 주목 받는 사용자가 있는가 하면, 평균적인 사용자의 follower 수는 300 남짓이다 [14].

사용자가 작성한 컨텐츠는 Tweet이라고 불리며, 기본적으로[1] 누구에게나 공개되어 있다. Tweet의 고유한 특징 3가지는 다음과 같다.

Mention: 다른 사용자를 언급하는 기능. 문장 안에서 '@user_id' 와 같이 표시한다. 언급된 사용자들을 주목시키는 효과가 있다 [15].

Reply: 원래 글에 답장하는 기능. 위의 Mention과 연계되어 사용한다.

Retweet: 다른 사용자의 글을 전달하는 기능. 문장 안에서 'RT @user_id' 와 같이 표시하는 경우가 많으며, 다른 양식도 다수 존재한다 [15]. 원래 글에 본인의

[1] 기본 값은 모두에게 공개이며, Private 설정에 따라 비공개로 전환할 수 있다.

의견을 붙여서 전달할 수 있는 (수동) Retweet 방식과, Twitter에서 제공하는 RT 방식 (원 글의 복제) 두 가지가 존재한다 [16].

앞에서 언급한 Tweet의 3가지 특징은 원래 Twitter에서 제공되는 기능은 아니고 사용자들이 자발적으로 규정한 것들이나, 점차 시스템에 포함되었다.

Twitter의 특질들은 대중적인 인기 (5억에 가까운 가입자), 주목 받는 사용자 (백만 이상의 Follower를 가짐), 실시간적 이슈에 반응 (Michael Jackson 사망 당시 시간 당 10만여 개의 Tweet이 작성됨 [17]) 하는 것과 같은 현상을 나타내고 있다.

2. Data 위주 연구 방법론을 사용한 Twitter 연구

Twitter의 인기에 따라, 이를 주제로 한 연구도 많이 진행되었다. 시간의 흐름과 Twitter 서비스의 발전에 따라 초창기와 이후의 연구 주제는 다소 변화하였다. 초기에는 Twitter 서비스의 특징과 기본적인 특성에 대해 연구하였으며, 이후 수치적인 데이터를 토대로 사용자 분류를 수행하는 연구가 주를 이루었다. 최근에는 사용자들 또는 컨텐츠 사이의 네트워크 구조 분석이나 영향력, 정보 흐름에 대한 연구가 수행되고 있다. 그림 1을 보면 시간이 흐를수록 데이터 수집이 대량화 되는 것을 볼 수 있다.

SNS를 대상으로 데이터 중심의 연구를 수행한 논문들은 상당수 존재한다. 이들 논문에서 연구 방법론에 해당되는 항목을 보면, '어떤' 데이터를 '언제', '얼마나' 모았다는 것에 대한 설명이 나와 있다. 이미 존재하는 프로그램을 사용했거나 직접 작성한 경우 프로그램 이름과 주요한 특징을 설명하고 있으며, 여러 대의 서버를 사용하였다면 그 수와 성능을 기록하였다 [18]. 하지만 데이터 수집과 처리 방법,

실제로 마주칠 수 있는 문제들에 대한 서술은 상대적으로 적기 때문에, 후속 연구를 진행하려고 하는 다른 연구자 입장에서 아쉬움을 느낄 수 있다.

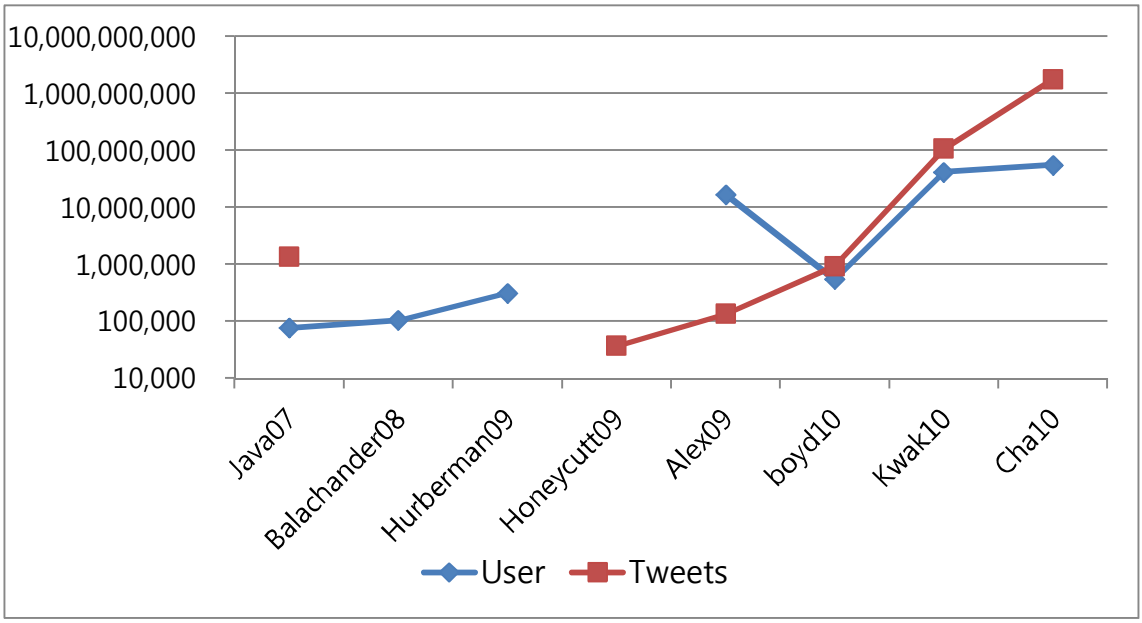


그림 1 연구에서 수집된 사용자 정보와 Tweet 수

Twitter와 관련하여 데이터 수집 방법만을 전문적으로 다룬 논문은 없으나, 본 논문의 목적과 가장 유사한 논문은 Erramilli 외 [19]이다. 이 논문에서는 Twitter 사용자의 쓰기 (글 남기기) 동작에 의미를 두고, 동작의 빈도와 시간대를 측정하였다. 240만 사용자와 1천2백만 Tweet을 수집하였으며, 분석에 사용된 항목은 Follower/Following 관계, 사용자 숫자 ID, 위치, Tweet 내용과 ID, 작성일이다. 수집된 대량의 Tweet 데이터를 공유하는 것이 어렵기 때문에, 이 논문에서는 분석한 데이터 패턴을 통해 사용자들의 글쓰기 동작을 시뮬레이션 할 수 있는 시스템을 개발하여 경향을 예측해 볼 수 있도록 하고 있다.

Java 외[20]에서는 7만 6천 사용자와 130만여 Tweet을 대상으로 Twitter 사용자 특징 분석과 Trend에 대한 네트워크 분석을 수행하였다. 수집 대상은 사용자 숫자 ID, Follower/Following 관계, 위치 정보, Tweet 내용과 ID, 작성일이다.

Krishnamurthy의 조사 [21]에서는 Twitter의 10만 사용자를 대상으로 계층 분류와 행동 분석을 수행하였다. 분석에 근거가 되는 항목은 Follower와 Following, Tweet 총 수, 사용자 숫자 ID, 작성 환경, Tweet 작성 시간이다.

Huberman [22]에서는 31만여 사용자를 대상으로 실제 커뮤니케이션이 일어나는 네트워크에 대한 분석을 수행하였다. 수집 항목은 Follower 수, Following 수, Tweet 내용, Tweet 작성일이다.

Honeycutt 외 [23]에서는 3만 7천여 개의 Tweet을 대상으로 Twitter의 Mention과 대화 기능에 대해 분석하였다. 분석 근거 항목은 개별 Tweet이며, Public Timeline을 지속적으로 수집한 것이다.

Leavitt 외 [24]에서는 12명의 유명 Twitter 사용자와 그의 Following network를 대상으로 영향력 연구를 수행하였다. 분량은 1천5백만 사용자, 13만 Tweet이며, 수집 항목은 사용자의 Follower 목록, Tweet Timeline과 Reply, Retweet, Mention 내용이다.

boyd 외 [15]에서는 44만여 사용자의 Tweet과 20만여 Retweet을 대상으로 Tweet 유형과 Retweet 행동에 대한 분석을 하였다. 수집 대상은 Public Timeline의 Tweet들과 검색 기능으로 수집한 Retweet 내용들이다.

Kwak 외 [25]의 경우 4200만 사용자와 1억여개의 Tweet을 대상으로 사용자 특성과 영향력, 유행하는 Topic에 대한 분석을 수행하였다. 수집 대상은 Follower/Following Network, Tweet 수, 사용자 시간대, Retweet 수, 작성한 Tweet들이다. 시간적으로 변화하는 이슈 추적을 위해 주기적으로 데이터를 수집하였으며, Spam 데이터를 검출하여 제거하였다는 점이 특징적이다. 또한 수집한 대량의 데이터를 일부 공개하고 있다는 점에서 의의가 있다.

Cha 외 [26]에서는 5천 4백만 사용자와 20억개의 follow 연결을 기반으로 사용자 영향력을 측정하였다. 분석 근거 항목은 follower 수, Retweet에 포함된 사용자, Mention에 포함된 사용자, Tweet 내용(키워드)이며, 해당 시점에서 Twitter 연구로는 가장 많은 데이터를 수집, 분석한 것이다 [27].

다음 표는 앞서 언급한 논문들에서 수집, 분석한 데이터 항목들을 정리한 것이다.

논문	User Profile											Tweet 과 Timeline						
	Num. ID	Username	Name	Location	Bio.	Follower, Following #	Relationship	Reg. Date	Time Zone	Total Tweet #	Language	Tweet 내용	ID	작성일	작성 Source	Mention	Reply	Retweet
Java07	●			●			●					●	●	●				
Balachander08	●						●							●	●			
Huberman09						●						●		●				
Honeycutt09												●						
Alex09						●	●					●				●	●	●
boyd10												●						●
Kwak10							●		●	●		●						●
Cha10						●						●				●		●
Erramilli11	●			●			●					●	●	●				

표 1 Twitter 연구 논문에서 사용한 데이터 항목

3. Twitter 데이터 분석 서비스

전문적이고 학술적인 분석 이외에도, 자료 수집 및 분석의 요구는 많다. 그 필요성은 크게는 신상품의 마케팅 및 시장 동향 분석으로부터 사용자 개인의 취미 생활을 위한 것까지 다양하다. 이런 목적을 위해, Twitter의 Open API를 활용하여 제 3

의 서비스 제공자가 만든 웹 서비스가 수백여 종 존재한다 [28]. 아래 표는 분석을 주 목적으로 하는 서비스 중 사용자가 일정 이상이라고 판단되는 곳에 대한 요약 설명이다.

이름	특징	비고
Export.ly	Following 사용자에 대한 통계 제공	
Klout	영향력 분석 (주제별)	
Peerindex	주요 Topic 분석, 사용자 랭킹	
retweetrank	Retweet 빈도 순위	API 제공
Searchtastic	사용자 랭킹, Hashtag 검색	
Trendistic	Issue와 Topic 분석	
TweetEffect	Tweet 작성과 Follower 변동 관계 표시	
Tweetmetrics	사용자 정보 통계	
Tweetmix	최근 이슈 표시	국내 서비스
Tweetrend	사용자 랭킹, 키워드	국내 서비스
TweetStats	시간대별 트윗 작성 정보, Reply 통계	
twInfluence	사용자 영향력 분석	API 제공
Twitaholic	사용자 랭킹	
Twitalyzer	영향력 분석	
TwitGraph	요일별 작성 통계, 주 사용 단어, 상위 Mention 사용자	
Twitter Grader	사용자 랭킹, 기본 정보 통계	
TwitterCounter	시간 변화에 따른 Follower/Following/Tweet 작성 변화	
Twitturly	Tweet에서 많이 언급된 URL 나열	

표 2 Twitter 분석 서비스 요약

앞서 나열한 서비스 중 Export.ly의 경우, 텍스트 정보뿐만 아니라 그림 2와 같은 그래프 통계 정보를 사용자가 직접 파일로 받아볼 수 있다는 점이 특징적이다.

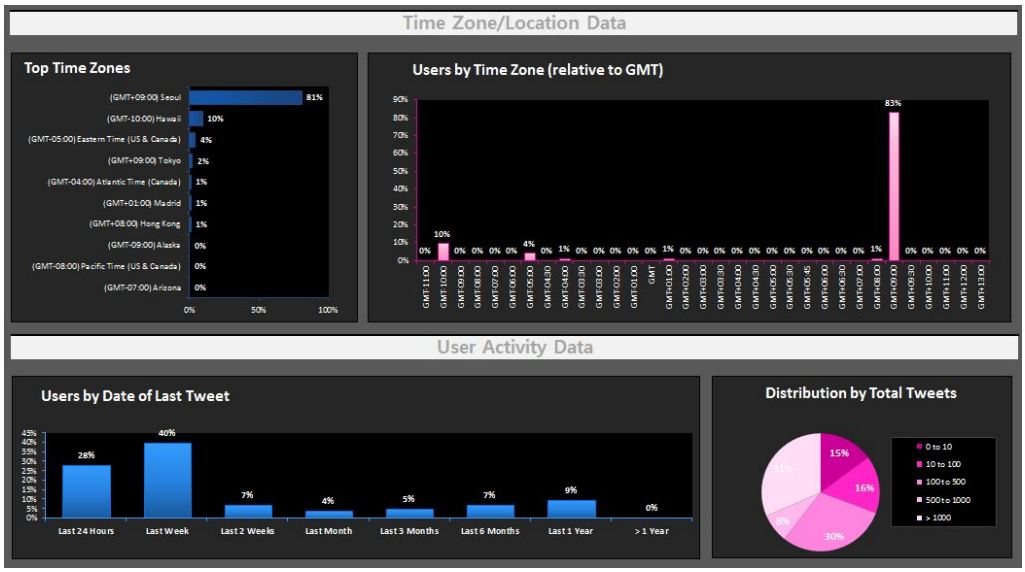


그림 2 Export.ly의 분석 결과 예

여러 종류의 Twitter 분석 서비스는 다음과 같은 공통된 특징을 가지고 있다.

가. 데이터를 수집하고 처리하지만, 그 방식과 기준이 무엇인지 자세히 설명하지 않음

나. API에서 제공되는 기본적인 데이터를 기반으로, 간단한 통계 처리를 수행한 것이 대부분임

다. 원하는 data set을 정할 수 없고 서비스가 사전에 정해놓은 분석 영역만 볼 수 있음

고유의 데이터 수집과 처리 기준이 서비스의 경쟁력과 직결되는 만큼, 가 항과 같이 세부 내역을 공개하지 않는 이유는 어느 정도 납득할 수 있다. 하지만 분석 기준을 검증할 수 없는 문제도 존재한다. 한편, Twitter에서 시간당 API 호출 횟수를 제한하고 있는 만큼, 여러 사람에게 서비스를 제공하기 위해서는 나 항과 같이 최소한의 기본적인 데이터만을 사용하거나, 지속적으로 자체 데이터를 축적한 다음 그것을 토대로 통계를 수행하게 된다. 후자의 방법을 사용할 경우 타 서비스로서

Twitter에서 발생한 모든 Tweet을 수집하는 것은 거의 불가능하며 (시간당 발생되는 Tweet 수와 사용자 수 때문), 주류 사용자 네트워크에서 거리가 먼 사용자의 경우 통계에서 빠지기 쉽다는 단점이 있다.

연구자 입장에서 가장 큰 문제는 다 항과 같이, 본인이 원하는 종류의 데이터를 선택할 수 없다는 점이다. 따라서 앞에서 나열한 수많은 서비스가 있음에도 불구하고, 자신의 연구를 위해서는 직접 데이터와 API를 다룰 수 밖에 없으며, 많은 시간과 노력이 필요하게 된다.

제 3 장　연구 문제 및 방법

제 1 절　　연구 문제

본 연구의 목적은 기술적인 배경이 적은 사용자라도 SNS를 대상으로 하는 연구를 설계, 진행시키는 데 어려움이 없도록 하는 데 있다. 기술적인 배경이 적은 사용자는 크게 둘로 나눌 수 있는데, SNS의 연구에 관심이 있는 사회과학자와 SNS API에 아직 익숙하지 않은 프로그래머이다. 전자의 경우 이론에 대한 사전 지식이 있고 연구 문제를 설정하는 것은 용이한 편이나 실제 데이터 수집과 연구 문제의 검증에 어려움을 겪는다. 후자의 경우 프로그래밍 기술을 가지고 있으나 SNS가 가지고 있는 특성과 데이터 종류, 요청 방법에 대해서는 익숙하지 않다.

이러한 사용자들의 요구 사항을 만족하는 데이터 수집-분석 Platform을 작성하기 위해, 다음과 같은 연구 문제를 설정하였다.

(연구 문제 1) SNS로부터 얻을 수 있는 데이터와 도출할 수 있는 정보는 무엇인가

자료를 수집하거나 분석하기에 앞서 어떤 데이터가 사용 가능한지 파악하는 것이 우선이므로, 서비스로부터 직접 얻을 수 있는 정보와, 이에 계산 등의 절차를 거쳐 도출할 수 있는 정보가 무엇인지 파악한다.

(연구 문제 2) SNS에서 데이터를 수집할 때 주의할 점은 무엇이며 대처 방법이 있는가

서비스의 특성과 자료 수집 프로세스 진행 상의 문제로, 데이터 수집 과정에서 여러 종류의 문제가 발생할 수 있다. 본 연구에서는 이러한 문제점을 미리 살펴보고, 찾아낸 대응 방법은 이후 작성할 데이터 수집/분석 도구에 반영하도록 한다.

(연구 문제 3) SNS를 대상으로 하는 연구에서 사용될 데이터 수집/분석 도구에는 어떤 요소가 포함되어야 하며 어떻게 설계되어야 하는가

다음에는 지금까지 살펴보았던 연구와 서비스의 문제점, 개선책을 토대로 데이터 수집-분석 도구(이하 Platform)를 설계하도록 한다. 연구 초반에 소수의 연구자로서 연구 문제도 설정하고 기술적인 이해도 해내기는 어렵다. 논문에서 작성하고자 하는 Platform은 일차적으로 연구 초-중반의 데이터 경향 파악과 가설 설정 단계에서 도움을 주는 것을 목표로 하며, 이후 연구 발전에 따라 기능을 확장할 수 있도록 한다. 이 Platform을 통해 이해하기 쉬운 방법으로 데이터를 수집하고 처리하는 방법을 제공하도록 한다.

제 2 절 연구 방법

연구 문제 1을 해결하기 위해 각 SNS에서 제공하는 API 문서를 분석하고, 요청할 수 있는 동작과 데이터 항목의 종류를 파악한다. 모든 SNS를 대상으로 하기는 어려우므로, 본 논문에서는 사용자가 많으며 상호 교류가 발생하는 세 개의 서비스(Facebook, Twitter, YouTube)를 대상으로 하였다. 다만 Platform에서 구조 변경 없이 다른 SNS를 지원할 수 있도록 설계한다. 동작과 데이터 항목의 경우에도 제공하고 있는 것은 많으나, 비교적 연구에 사용될 수 있다고 판단되는 것을 우선 선정하도록 한다.

연구 문제 2를 해결하기 위해서는 실제 연구에서 발생했던 문제들을 토대로 원인을 살펴보고, 해결책을 제시한다. 특정 서비스 만을 위한 접근 보다는 SNS 일반적인 연구에 적용할 수 있는 방안을 찾는다. 자료 원(Source), 데이터 수집 방법, 프로그래밍의 기술적인 문제, 데이터 처리 방법 등에 대해 살펴보도록 한다.

연구 문제 3을 위해서는 실제 사용 가능한 응용 프로그램을 설계, 작성한다. 도구는 일반적인 독립실행 프로그램의 형태를 가진다. 앞에서 설명했듯이 데이터 수집과 분석 결과가 복합적일 수 있도록 필요한 데이터를 선택하고 조합할 수 있도록 한다. 도구의 유효성을 검증하기 위해 수작업 자료 수집과 프로그램을 이용한 자료 수집의 결과 및 성과를 비교하고, 실제 연구 Case에 적용할 수 있는 과제를 구현해 보도록 한다.

제 4 장　기능 요구 사항 분석

제 1 절　SNS API 분석

앞에서 언급한 것과 같이, 세 종류(Facebook, Twitter, YouTube)의 SNS에 대해 API 분석을 하도록 한다. 이 중 YouTube의 경우 통상적인 개념의 SNS에 포함되기 보다는 비디오 공유 서비스로 분류되는 경향이 강하나, 매체적인 특성을 제외하면 사용자 간의 상호작용이 타 SNS와 유사하며, 다른 SNS와의 상호작용 (링크로 발생되는) 경우가 많아서 포함되었다. 모든 서비스의 분석 기준은 제공되는 API 문서이며, 부가적으로 관련 프로그래밍 라이브러리[2][3][4]를 참고하였다.

1. Facebook

Facebook에서 제공하는 개발자 문서[5]에 따르면 주요한 자료 형태(이하 '개체'로 표현)를 25종 나열하고 있다. 하지만 실제로 각 개체에 하부 개체가 더 있으므로 전체 개체의 수는 늘어난다. 이 중 연구에 불필요하다고 판단되는 개인 정보(메일, 쪽지 등) 관련 개체를 제외한 것은 다음과 같다. (개체에서 점으로 연결된 이름은 하부 개체임을 의미함)

---

[2] Facebook: RestFB (http://restfb.com/)

[3] Twitter: Twitter4J (http://twitter4j.org/)

[4] YouTube: Google Data Java Client Library (http://code.google.com/p/gdata-java-client/)

[5] https://developers.facebook.com/docs/reference/api/, 사용자 로그인 필요

개체	이름	설명
Album	앨범	사진들이 있는 앨범
CategorizedFacebookType	항목 (+이름, 분류)	분류, 이름, 고유번호 정보가 들어있음
Checkin	체크인	사용자가 방문한 장소에 대한 기록
Checkin.Place	체크인 장소	방문한 장소에 대한 정보
Checkin.Place.Location	체크인 장소 위치정보	방문한 장소의 위치 정보
Comment	덧글	작성된 덧글들
Event	이벤트	이벤트 항목
FacebookType	항목	고유번호 정보가 들어있음
Group	그룹	그룹 항목
Link	링크	링크 게시물 항목
Location	위치	항목과 관련된 위치 정보
NamedFacebookType	항목 (+이름)	항목을 가리키며, 이름, 고유번호 정보가 들어있음
Note	노트	노트 항목
ObjectTag	Object 태그	본문 중 언급된 항목(사용자, 페이지)에 대한 태그 정보
Page	페이지	페이지 항목
PageConnection	페이지 목록	여러 페이지에 대한 목록
Photo	사진	사진 게시물 항목
Photo.Image	이미지	여러 크기의 사진에 대한 정보
Photo.PhotoTag	사진 태그	사진에 지정한 항목 태그 정보
Post	게시물	(모든 종류의) 게시물 항목
Post.Comments	게시물 덧글 목록	덧글들의 목록
Post.Likes	게시물 좋아요 목록	'좋아요'의 목록
Post.Place	게시물 위치	게시물과 관련된 장소 정보
Post.Privacy	게시물 공유 설정	게시물의 공개 대상에 대한 정보
Post.Property	게시물 부가 정보	게시물에 대한 부가 정보
StatusMessage	상태 메시지	상태 메시지 항목
User	사용자	사용자 항목
User.Education	학력	사용자의 학력 정보
User.EducationClass	수업	사용자가 들은 수업에 대한 정보

User.Family	가족	사용자의 가족에 대한 정보
User.Sport	스포츠	사용자와 관련된 스포츠 정보
User.Work	직장	사용자의 직장 정보
Venue	장소	항목과 관련된 장소 정보
Video	비디오	비디오 게시물 항목

표 3 Facebook 개체 목록

이 중 중요한 개체는 User, Post 정도이나 유사한 개체들 중에도 가지고 있는 세부 정보에 다소 차이가 있으므로 (Post와 Checkin, Link, Note, Photo, StatusMessage, Video) 필요에 따라 사용하게 된다.

다른 서비스에서도 항목 간의 연결 관계가 구성되어 있지만, Facebook의 경우에는 'Graph' API라 하여 연결 구조를 서비스에서 중요하게 사용하고 있다. 개체가 고유 번호뿐 아니라 URL을 갖고 있어 이를 통해 항목에 접근할 수 있으며, 서비스 바깥에 존재하는 항목도 URL을 통해 대표된다는 점이 특징적이다.

각 개체 별로 세부 정보 항목을 가지고 있는데, 이 중 User (사용자 정보) 개체가 가지고 있는 정보 항목은 다음과 같다. 각 항목의 값은 단일 값인 경우도 있고 (생일) 값의 목록(학력)일 수도 있다.

항목	이름	설명
bio	내 소개	사용자의 소개
birthday	생일	사용자의 생일. 년도는 공개하지 않았을 수 있음.
connection	관련 정보	사용자에 관련된 종류의 정보
Education	학력	사용자의 학력 정보
Email	메일 주소	사용자의 메일 주소
favoriteAthletes	좋아하는 운동선수	사용자가 좋아하는 운동선수의 목록
favoriteTeams	좋아하는 스포츠팀	사용자가 좋아하는 스포츠 팀의 목록
firstName	이름	사용자의 이름 중 이름 부분
gender	성별	사용자의 성별
hometown	출신지	사용자의 출신지

id	고유번호	항목을 식별하는 고유번호
interestedIn	관심사	사용자의 관심사
languages	언어	사용자가 사용하는 언어의 목록
lastName	성	사용자의 이름 중 성 부분
link	링크	Facebook 페이지로 향하는 링크
locale	로케일	사용자가 서비스를 사용하는 언어 (ISO 코드)
location	거주지	사용자의 거주지
meetingFor	관심사 (성별)	사용자가 만나는 것을 선호하는 상대 성별
middleName	가운데 이름	사용자의 이름 중 가운데 이름 부분
name	이름	항목의 이름
political	정치 성향	사용자의 정치적 성향
quotes	좋아하는 인용구	사용자가 좋아하는 인용구
relationshipStatus	결혼/연애 상태	사용자의 결혼 또는 연애 상태
religion	종교	사용자의 종교
significantOther	결혼/연애 상대	사용자의 결혼 또는 연애 상대자
sports	좋아하는 스포츠	사용자가 좋아하는 스포츠의 목록
timezone	표준 시간대	사용자가 위치한 곳의 표준 시간대
type	형식	항목의 형식
updatedTime	수정 시간	최근 수정된 시간
username	사용자 이름	사용자 고유의 이름(주소 뒷부분)
verified	인증 여부	인증된 사용자인지 여부
website	웹사이트	사용자의 웹사이트 주소
work	직장	사용자의 직장에 대한 정보

표 4 User 개체의 정보 항목

API에서는 주소(URL)를 통해 서비스의 정보 자원에 접근할 수 있도록 하고 있다. 이를 동작을 기준으로 재분류한 목록은 다음과 같다.

이름	설명
목록 정보 가져오기	여러 항목의 정보를 목록으로 가져온다.
항목 정보 가져오기	항목 한 개의 정보를 가져온다.

항목에 대한 상세 정보 가져오기	간략 정보(고유 번호, 이름)만 있는 항목에 대한 상세 정보를 가져온다.
게시물 정보 가져오기	게시물의 정보를 가져온다.
사용자 정보 가져오기	사용자의 정보를 가져온다.
사용자의 연관 정보	사용자에 연관된 정보를 가져온다.
사용자 담벼락	사용자의 담벼락 글을 가져온다.
사용자 친구 목록	사용자의 친구들 목록을 가져온다. 간략 정보만 존재하며, 상세 정보는 별도로 읽어와야 합니다.
사용자 친구의 정보	사용자의 친구들에 대한 사용자 정보를 가져온다.
타임라인	본인의 타임라인을 가져온다.
사용자가 좋아하는 음악	사용자가 좋아하는 음악 페이지 목록을 가져온다.
사용자의 사진	사용자가 작성한 사진 게시물을 가져온다.
사용자가 작성한 게시물	사용자가 작성한 게시물을 가져온다.
사용자의 상태 메시지	사용자가 작성한 상태 메시지를 가져온다.
사용자의 비디오	사용자가 작성한 비디오 게시물을 가져온다.

표 5 Facebook 동작 목록

2. Twitter

Twitter에서 API를 통해 얻을 수 있는 개체는 다음과 같다. 서비스의 특성에 따라 개체의 종류가 상대적으로 적은 것을 볼 수 있다.

개체	이름	설명
GeoLocation	위치	지리상의 지점
HashtagEntity	해시태그 항목	트윗에 표시된 해시태그 항목
MediaEntity	미디어 항목	트윗에 포함된 미디어 항목
Place	장소	국가 또는 지역
Status	트윗	작성된 트윗 항목
Trend	트렌드	트렌드 항목
Trends	트렌드 목록	특정 시간의 트렌드 목록
URLEntity	주소 항목	트윗에 포함된 단축 웹주소 항목

User	사용자	사용자 정보
UserList	사용자 리스트	작성된 사용자 목록
UserMentionEntity	멘션 항목	트윗에 포함된 멘션 사용자 항목

표 6 Twitter의 개체 목록

이 중 가장 중요하다고 판단되는 Tweet 관련 개체 (Status)의 정보 항목은 다음과 같다. Tweet의 상태(새로 작성되었는지, 답글인지, Retweet 되었는지)에 따라 항목 값의 존재 유무에 차이가 있다.

항목	이름	설명
contributorsIDs	공동 작성자	Tweet 공동 작성자의 고유 번호 목록
createdAt	작성 시간	Tweet을 작성한 시간
geoLocation	위치 정보	Tweet과 관련된 위치 정보
hashtagEntities	해시태그	Tweet에 포함된 해시태그 항목의 목록
id	고유 번호	Tweet의 고유 번호
inReplyToScreenName	응답한 사용자	응답 Tweet의 대상 사용자 이름
inReplyToStatusId	응답한 Tweet	응답 Tweet의 원문 Tweet의 고유 번호
inReplyToUserId	응답한 사용자 고유 번호	응답 Tweet의 대상 사용자 고유 번호
isFavorited	관심글 여부	Tweet을 관심글로 지정했는지 여부
isTruncated	넘침 여부	Tweet의 본문이 140자 제한을 초과하였는지 여부
links	링크 목록	Tweet에 포함된 링크의 목록
mediaEntities	미디어	Tweet에 포함된 미디어 항목의 목록
myRetweetedStatus	Retweet 한 Tweet	해당 Tweet에 대해 본인이 Retweet한 정보
place	장소	Tweet과 관련된 장소
retweetCount	Retweet 수	Tweet이 Retweet 된 횟수
retweetedStatus	Retweet 원본	Retweet되었을 경우, Tweet의 원본 Tweet 정보가 들어있음
source	작성 환경	Tweet을 작성한 환경 또는 어플리케이션 이름
text	본문	Tweet 본문
urlEntities	주소	Tweet에 포함된 단축 웹 주소 항목의

		목록
user	사용자	Tweet을 작성한 사용자 정보
userMentionEntities	Mention	Tweet에 포함된 사용자 Mention 항목의 목록
wasRetweetedByMe	직접 Retweet 함	본인이 Tweet을 Retweet 했는지 여부

표 7 Tweet 개체의 정보 항목

특기할 만한 점은, 비교적 최근에 API가 변경되면서 Tweet에 대한 세부 정보들(이름이 'Entities'로 끝나는 항목)이 포함되었다는 것이다. 이 정보가 존재하지 않았을 때는 데이터 분석할 때 텍스트를 처리하여 직접 정보를 추출해야 했으나, 정보가 완전히 얻어내지 못하는 경우가 종종 발생하였다. 서비스 자체에서 미리 처리된 데이터를 제공하므로 이러한 수고를 덜게 되었다.

이름	설명
일간 Trend 목록	하루 동안의 Trend 목록을 가져옴
현재 사용자 고유번호	현재 로그인 한 사용자의 고유 번호를 얻어옴
사용자 고유번호 무작위 생성	무작위의 사용자 고유 번호를 생성하여, 임의의 사용자를 대상으로 할 수 있도록 함
Tweet 특징 추출	Tweet에 다음과 같은 특징이 존재하는지 확인함: Retweet, Mention, Singleton, Hash Tag, 링크.
Tweet 링크 추출	Tweet에 포함된 모든 링크 주소를 추출함
일 평균 Tweet 수	사용자의 일 평균 작성한 Tweet 수를 계산
Tweet 계정 사용일수	사용자가 계정을 등록한 뒤로 지난 날짜 수를 계산
Follower-Following 비율 계산	사용자의 Follower 대 Following 사용자 수 비율을 계산
Follow 사용자 목록	사용자를 Follow하는 다른 사용자의 목록을 가져옴
Following 사용자 목록	사용자가 Following한 다른 사용자의 목록을 가져옴
사용자 정보 가져오기	사용자에 대한 정보를 가져옴
리스트 사용자 목록	리스트에 추가된 사용자의 목록을 가져옴
사용자 리스트 가져오기	사용자가 작성한 리스트를 가져옴
Tweet 가져오기	사용자가 작성한 Tweet 목록을 가져옴

사용자간 대화 분석	사용자와 다른 사용자 사이에 발생한 대화의 횟수를 분석
주간 Trend 목록	주간의 Trend 목록을 가져옴

표 8 Twitter 동작 목록

표 8은 API를 토대로 정리한 동작의 목록이다. 몇 가지 동작(사용자 고유 번호 무작위 생성, 트윗 링크 추출 등)은 연구에 필요하다고 판단되어 추가하였다.

3. YouTube

비디오 중심의 서비스에 맞게 비디오 및 비디오 목록에 관련된 정보, 미디어 메타 정보에 대한 항목이 존재한다. 40개의 개체를 파악하였으며 이들 중 하위 개체를 제외한 목록은 다음 표와 같다.

개체	이름	설명
Category	분류	특정 항목이 속한 분류를 나타냄
CommentEntry	댓글	작성된 댓글 항목
Content	내용	항목의 내용에 대한 정보
FeedLink	목록 링크	항목 목록에 대한 링크
Link	링크	다른 항목에 대한 링크
MediaThumbnail	미리보기 장면	동영상의 특정 시점에 대한 미리보기 장면 정보
Person	인물	간략한 사용자 정보 (작성자 등)
PlaylistEntry	재생 목록 항목	재생 목록의 항목
PlaylistLinkEntry	재생 목록 링크	재생 목록에 대한 링크
PlaylistLinkFeed	재생 목록의 목록	여러 재생 목록에 대한 목록
SubscriptionEntry	구독 항목	사용자가 구독한 항목
SubscriptionFeed	구독 목록	사용자가 구독한 항목들에 대한 목록
TextConstruct	텍스트	텍스트 정보가 들어있습니다.
UserProfileEntry	사용자 정보	사용자에 대한 정보
VideoEntry	비디오	비디오 항목에 대한 정보

VideoFeed	비디오 목록	비디오에 대한 목록
YouTubeMediaGroup	미디어 정보	미디어에 대한 상세 정보

표 9 YouTube의 개체 목록 (일부)

개체들 중 가장 핵심인 비디오 항목 정보는 다음과 같은 세부 정보 항목을 가진다. YouTube 데이터 구조의 특징은 데이터 Category마다 묶어서 개체화 시켰다는 것인데, 이에 따라 하나의 정보를 얻기 위해 여러 단계의 개체를 거쳐 들어가야 한다. (항목명에 음영된 것은 세부 항목이 개체인 것이다)

항목	이름	설명
authors	작성자 목록	항목 작성자의 목록
categories	분류	항목의 분류
commentEntries	댓글 목록	항목에 달린 댓글의 목록
content	내용	항목의 내용
contributors	기여자 목록	항목 기여자의 목록
edited	편집 시간	항목을 편집한 시간
embeddable	공유 가능	외부 사이트에 동영상을 공유할 수 있는지 여부
geoCoordinates	위치	항목과 관련된 위치에 대한 정보
id	고유 번호	항목의 고유 번호
kind	종류	항목의 종류
label	레이블	항목의 레이블
location	장소	비디오와 관련된 장소
mediaGroup	미디어 정보	미디어에 대한 상세 정보
publicationState	공개 상태	비디오 항목의 공개 진행 상태
published	작성 시간	항목을 작성한 시간
racy	제한된 컨텐츠 포함	비디오에 제한된 컨텐츠가 포함되어 있는지 여부
rating	평점	항목의 평점에 대한 정보
recorded	녹화 시간	항목을 녹화한 시간
relatedVideos	관련 비디오	관련 비디오의 목록
responseVideos	응답 비디오	응답 비디오의 목록

rights	권한	항목과 관련된 권한 정보
statistics	비디오 통계	비디오에 대한 통계 정보
summary	요약	항목의 요약
title	제목	항목의 제목
updated	최근 수정 시간	항목이 최근에 수정된 시간
ytIncomplete	데이터 완결성	비디오 메타 정보에 빠진 항목이 있는지 여부
ytRating	YouTube 평점	항목의 평점에 대한 정보

표 10 비디오 개체의 정보 항목

마지막으로 서비스에서 제공하는 동작의 목록이다. 한 개체에 속한 정보의 양이 많은 반면에, 사용할 수 있는 개별 동작의 수는 많지 않은 것을 볼 수 있다.

이름	설명
사용 가능한 비디오 자막 확인	사용 가능한 비디오 자막 언어 목록을 가져옴
비디오 목록 가져오기	비디오의 목록을 가져옴
비디오 정보 가져오기	비디오의 정보를 가져옴
비디오 자막 가져오기	비디오의 자막을 가져옴
사용자 정보 가져오기	사용자들의 세부 정보를 가져옴
재생 목록 가져오기	사용자의 재생 목록들을 가져옴
구독 목록 가져오기	사용자의 구독 목록들을 가져옴
즐겨찾기 가져오기	사용자가 즐겨찾기 한 비디오의 목록을 가져옴
올린 비디오 목록	사용자가 올린 비디오의 목록을 가져옴
검색어로 찾기	지정한 검색어로 비디오를 찾습니다.
키워드로 찾기	지정한 키워드로 비디오를 찾습니다.

표 11 YouTube 동작 목록

제 2 절 SNS 연구 상의 문제점

이번에는 SNS를 대상으로 하는 연구를 진행하면서 관련 연구에 경험이 적은 연구자(사회과학자나 프로그래머 포함)가 자주 맞닥뜨릴 수 있는 문제점들에 대해 살펴보도록 한다.

1. 어떠한 데이터가 있는지 파악하기 어려움

앞 절에서 살펴보았듯이 각 서비스 별로 방대한 종류의 개체, 세부 항목이 존재하며 각각의 의미를 판단하기 어렵다. 특히 각 세부 항목이 단순한 값(숫자나 텍스트)으로 구성된 것이 아니라 또 다른 개체일 경우 그 구조를 이해하는 것은 더욱 어려워진다. 이와 같이 자료 구조가 복잡한 원인은, 자료 교환 형식이 계층화를 촉진하거나(XML 구조일 경우), 한 번의 자료 전달에 중복 없이 많은 정보를 담기 위함일 수 있다. API 문서의 경우 이러한 자료 구조를 있는 그대로 보여주지 못하고 뭉뚱그려 표 형식으로 설명하여, 이해에 어려움을 주고 있다.

이러한 자료 구조를 좀 더 쉽게 이해하기 위해서는 원래의 계층 구조, 세부 항목에 대한 개체 형식을 있는 그대로 보여주어야 할 필요가 있다. 또한 각 자료 별로 어떠한 형식이고 어느 경우에 사용한다는 설명을 같이 조회할 수 있어야 한다.

2. 사람이 하기 어려운 작업들

SNS API의 데이터 형식은 거의 XML 또는 JSON 형식이며, 이는 일반적인 웹 서비스의 경향에 따른 것이다. XML의 경우 (이진 데이터 형식에 비해) 사람이 읽을 수 있는 형식임을 표방하고 있으나 반론도 적지 않다. 실제로 SNS에서 API를 통해 읽어 온 원본 XML 데이터를 직접 확인한다 하더라도 어디에 어떤 데이터가 있는 것인지 파악하기 어렵다.

<?xml version='1.0' encoding='UTF-8'?><entry xmlns='http://www.w3.org/2005/Atom' xmlns:media='http://search.yahoo.com/mrss/'
xmlns:gd='http://schemas.google.com/g/2005'
xmlns:yt='http://gdata.youtube.com/schemas/2007'><id>http://gdata.youtube.com/feeds/api/videos/3DNaj8R4HJg</id><published>2012
-04-16T07:35:06.000Z</published><updated>2012-07-09T08:02:46.000Z</updated><category
scheme='http://schemas.google.com/g/2005#kind' term='http://gdata.youtube.com/schemas/2007#video'/><category
scheme='http://gdata.youtube.com/schemas/2007/categories.cat' term='Film' label='Film & Animation'/><category
scheme='http://gdata.youtube.com/schemas/2007/keywords.cat' term='1루수가 누구야?'/><category
scheme='http://gdata.youtube.com/schemas/2007/keywords.cat' term='who's on first'/><category
scheme='http://gdata.youtube.com/schemas/2007/keywords.cat' term='홍해라흥 픽쳐스'/><category
scheme='http://gdata.youtube.com/schemas/2007/keywords.cat' term='김만중'/><category
scheme='http://gdata.youtube.com/schemas/2007/keywords.cat' term='몽음보따리'/><title type='text'>1루수가
누구야?</title><content type='text'>몽음보따리 시리즈#2 who's on first? 의 한국판 애니메이션('who's on first?' korean
version)홍해라흥 픽쳐스 제공 (HHH PICTURES presents)제작 : 김만중 (Directed by Manjoong Kim)1루수는 누구야 인형
http://www.drjoy.co.kr/goods/view.php?seq=274&main=true&mainType=2 english channel (hhh pictures)</content><link
rel='alternate' type='text/html' href='http://www.youtube.com/watch?v=3DNaj8R4HJg&feature=youtube_gdata'/><link
rel='http://gdata.youtube.com/schemas/2007#video.responses' type='application/atom+xml'
href='http://gdata.youtube.com/feeds/api/videos/3DNaj8R4HJg/responses'/><link
rel='http://gdata.youtube.com/schemas/2007#video.related' type='application/atom+xml'
href='http://gdata.youtube.com/feeds/api/videos/3DNaj8R4HJg/related'/><link rel='http://gdata.youtube.com/schemas/2007#mobile'
type='text/html' href='http://m.youtube.com/details?v=3DNaj8R4HJg'/><link rel='self' type='application/atom+xml'
href='http://gdata.youtube.com/feeds/api/videos/3DNaj8R4HJg'/><author><name>manjoongkim</name><uri>http://gdata.youtube.com/fee
ds/api/users/manjoongkim</uri></author><gd:comments><gd:feedLink rel='http://gdata.youtube.com/schemas/2007#comments'
href='http://gdata.youtube.com/feeds/api/videos/3DNaj8R4HJg/comments'
countHint='3229'/></gd:comments><yt:hd/><media:group><media:category label='Film & Animation'
scheme='http://gdata.youtube.com/schemas/2007/categories.cat'>Film</media:category><media:content
url='http://www.youtube.com/v/3DNaj8R4HJg?version=3&f=videos&app=youtube_gdata' type='application/x-shockwave-flash'
medium='video' isDefault='true' expression='full' duration='285' yt:format='5'/><media:content
url='rtsp://v1.cache6.c.youtube.com/CiILENy73wIaGQmYHHjEjloz3BMYDSANFEgGUgZ2aWRIb3MM/0/0/0/video.3gp' type='video/3gpp'
medium='video' expression='full' duration='285' yt:format='1'/><media:content
url='rtsp://v7.cache8.c.youtube.com/CiILENy73wIaGQmYHHjEjloz3BMYESARFEgGUgZ2aWRIb3MM/0/0/0/video.3gp' type='video/3gpp'
medium='video' expression='full' duration='285' yt:format='6'/><media:description type='plain'>몽음보따리 시리즈#2 who's on
first? 의 한국판 애니메이션('who's on first?' korean version)홍해라흥 픽쳐스 제공 (HHH PICTURES presents)제작 : 김만중
(Directed by Manjoong Kim)1루수는 누구야 인형 http://www.drjoy.co.kr/goods/view.php?seq=274&main=true&mainType=2
english channel (hhh pictures)</media:description><media:keywords>1루수가 누구야?, who's on first, 홍해라흥 픽쳐스, 김만중, 몽
음보따리</media:keywords><media:player url='http://www.youtube.com/watch?
v=3DNaj8R4HJg&feature=youtube_gdata_player'/><media:thumbnail url='http://i.ytimg.com/vi/3DNaj8R4HJg/0.jpg' height='360'
width='480' time='00:02:22.500'/><media:thumbnail url='http://i.ytimg.com/vi/3DNaj8R4HJg/1.jpg' height='90' width='120'
time='00:01:11.250'/><media:thumbnail url='http://i.ytimg.com/vi/3DNaj8R4HJg/2.jpg' height='90' width='120'
time='00:02:22.500'/><media:thumbnail url='http://i.ytimg.com/vi/3DNaj8R4HJg/3.jpg' height='90' width='120'
time='00:03:33.750'/><media:title type='plain'>1루수가 누구야?</media:title><yt:duration
seconds='285'/></media:group><gd:rating average='4.6344395' max='5' min='1' numRaters='4913'
rel='http://schemas.google.com/g/2005#overall'/><yt:statistics favoriteCount='5653' viewCount='14940147'/><yt:threed
source='converted'/></entry>

그림 3 XML 결과 데이터의 예

연구 초기에 적은 양에 대한 데이터 분석이라 하더라도 이러한 데이터를 사람이 직접 처리하기는 힘들다. 따라서 원본 데이터를 사용하기 보다는 웹 상의 사용자 인터페이스에서 데이터를 찾는 것을 선호하게 된다. 하지만 사용자 인터페이스는 화면상의 제약이나 디자인적인 요소에 따라, 모든 정보가 한번에 보여지지 않으므로 많은 동작을 필요로 한다.

XML이나 JSON 형식의 데이터 모두 사람보다는 프로그램이 처리하기 쉬우므로, 프로그램으로 분석하여 추출된 데이터를 사용자가 이용할 수 있도록 한다.

또한 각기 다른 대상에 대해 같은 작업을 수행하는 반복 작업의 경우에도, 프로그램으로 처리하는 것이 더 유리하다. 이 역시 대상을 먼저 선정하고 그 대상에 대해 작업을 반복하도록 설정한다면 해결할 수 있는 문제이다.

3. 연구 발전 과정에서 data set의 변경

연구 문제 설정 과정을 거치면서 필요한 data set을 선정하고, 자료를 모으게 된다. 한번 선정한 set이 변동되지 않으면 좋겠으나, 분석을 하다 보면 특정 항목이 추가로 필요한 경우가 발생한다. 빠진 정보를 다시 수집하는 데는 생각보다 많은 시간이 걸리는데, 대개의 경우 하나의 세부 항목만 수집한다 하더라도 항목에 딸린 모든 정보를 요청해야 하기 때문이다. 따라서 거의 원래 데이터 수집만큼의 시간이 소요된다.

이 문제를 해결하기 위해서는 당장 필요하지 않은 항목의 정보라도 일단 저장해 두었다가, 필요한 항목만 선택해서 사용하는 방법이 있다. 단, 다음에 설명할 API 호출 횟수 제한이나 데이터 저장 공간의 문제가 있기 때문에, 저장하는 정보의 범위는 한 번 요청에 가져올 수 있는 것으로 한정하는 것이 적당하다.

4. API 사용상의 제한

각 SNS가 Open API를 제공하고 있으나, 우선은 일반 사용자들에게 원활한 서비스를 제공하는 것이 우선이다. 따라서 API를 통한 대량의 데이터 요청을 제한하고 있다. Twitter의 경우 한 IP에서 비 로그인 시 시간 당 150회, 로그인 시 350회의 서비스 호출이 가능하다[6]. 자사의 Whitelist에 등록할 경우 시간당 2만회까지 서비스 호출 횟수를 늘릴 수 있으나, 일반 사용자를 대상으로 한 제품(프로그램, 서비스)을 위한 것이 아닐 경우 허용해 주지 않는 경우가 많다.

[6] Twitter의 Traffic 현황에 따라 호출 제한 값은 그 이하가 될 수도 있다. [30]

YouTube의 경우 시간 당 제한은 없으나, 짧은 시간 내에 여러 번 자료 요청을 할 경우 속도를 늦춰줄 것을 요청하는 메시지를 전송한다. 개발자 문서[7]에 따르면 이런 경우 10분 동안 서비스에 대한 요청을 정지할 것을 권장하고 있다. Facebook의 경우 정해진 요청 횟수 제한과 같은 것은 없지만, 서비스 상태에 따라 데이터 대신 오류 메시지가 반환되는 경우가 있다.

이러한 제약에 따라 데이터 수집의 효율성이 저하될 수 있다. Twitter에 대한 실제 데이터 수집에서, 1시간 중 10분 만에 주어진 호출 횟수를 모두 사용하고 나머지 시간 동안 대기하는 상황이 발생하였다. 비효율적인 서비스 호출을 개선(예: 사용자 정보를 일괄적으로 요청하여 호출 횟수를 감소시킴)하면 이러한 문제를 완화할 수 있다. API 호출도 제한된 자원인 만큼, 상황 내에서 최대한 효율적으로 데이터를 요청하고 접근해야 한다.

또한 서비스 상태에 따라 돌발적으로 발생하는 오류를 처리해야 한다. 정상적으로 반환되던 데이터라도 서비스 사용자가 많을 경우 반환되지 않을 수 있으며, 이렇게 빠진 데이터는 자료에 왜곡을 가할 수 있다. 서비스 오류에 맞추어 재시도 하거나 적당한 오류 처리를 하는 것으로 대처할 수 있다.

[7] https://developers.google.com/youtube/faq#limits

제 5 장 Platform 설계

앞 단계의 요구 사항 분석을 토대로 Platform을 설계하였다. Platform의 이름은 Social Network Inspector라고 붙여, 프로그램을 통해 Social Network의 구조를 들여다 볼 수 있음을 나타내었다.

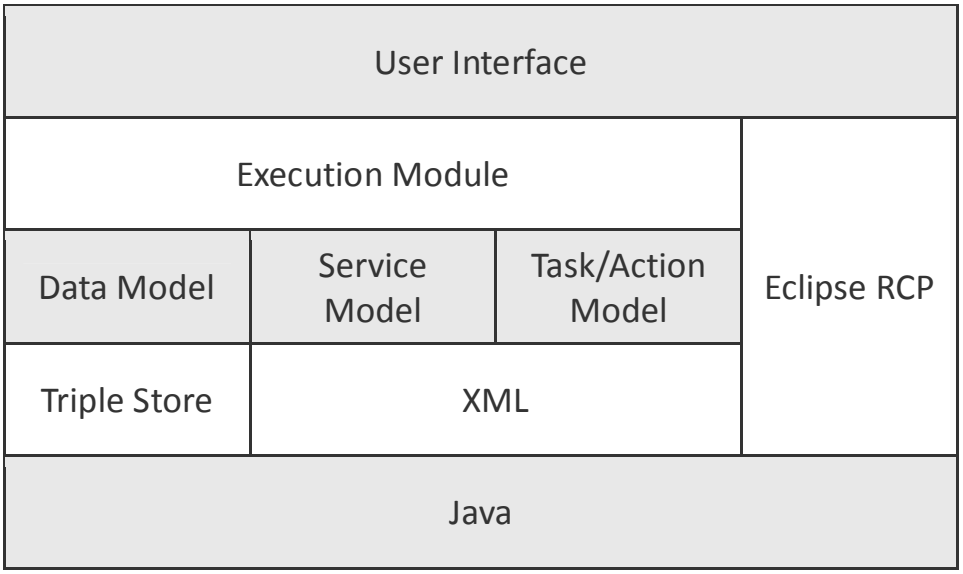


그림 4 Platform 구조도

제 1 절 데이터 모델링

프로그램에서 데이터를 저장하고 사용하기 위해 먼저 데이터 모델링이 필요하다. Platform에서 사용되는 데이터는 크게 서비스에 대한 정보와 실제 내용을 담은 것, 작업(Task)에 관련된 데이터 3가지로 나뉜다.

1. Service Model

서비스와 개체, 자료 항목의 구조를 나타내기 위한 데이터 모델이다. 다음과 같은 요소로 구성되어 있다.

가. Service: 각 SNS의 서비스에 대한 정보를 담고 있다. 기본 정보 (이름, 설명) 외에 Action과 Type의 세부 항목을 갖는다.

나. Action: 서비스에 대해 수행할 수 있는 동작들이다. 기본 정보 외 매개 변수(Parameter)의 세부 항목을 갖는다.

다. Parameter: 동작을 수행하는 데 필요한 설정 값이다.

라. Type: 서비스와 관련된 개체 자료형에 대한 정보이다. 기본 정보 외에 Field(세부 항목) 값을 갖는다.

마. Field: 세부 정보 항목에 대한 내용을 담고 있다.

이것은 곧 SNS에 대한 메타 정보라고 할 수 있으며, 각 서비스 별로 내용을 서술한 XML을 작성하여 여기에 정보를 저장하였다. 또한 이 과정에서 추후 구조 변경 없이 다른 서비스를 추가, 변경할 수 있도록 하였다.

2. Data Model

SNS 상에 존재하는 실제 데이터를 담고 있는 모델이며, 서비스로부터 전달받은 값을 해석하여 저장하는 일을 맡는다. 앞서 Service Model에서 설명한 '자료형'에 해당된다.

기본적으로 SNS 제공자의 데이터 파일 형식(XML, JSON) 또는 프로그래밍 라이브러리가 정의한 모델에 대응되도록 구성하였으나, 편의를 위해 추가 또는 제거한 항목이 존재한다.

수집 절차를 진행하면서 자료를 저장해야 하는데, 이를 위해 Database가 사용되었다. 일반적인 경우 관계형 Database에 Table을 설계하고 데이터를 저장하게 되지만, SNS의 자료 구조는 계층 구조를 가지고 있으며 정보 항목의 수도 매우 가변

적이다. 이러한 문제를 해결하기 위해 Ontology 정보를 저장하기 위해 사용되는 Triple Store를 도입하였다. 이에 따라 부가적으로 SNS의 정보를 Ontology linked data로 나타낼 수 있는 기능이 가능해졌다.

3. Task Model

컴퓨터에 작업을 시키기 위해서 프로그램을 작성해야 하는 것처럼, 자료 수집 절차를 정의하기 위해 간단한 구조의 명령 Set을 정의하였다. 여기서 전체적인 명령 구성을 Task('작업')이라고 부르고, 각 세부 명령 동작을 Action('동작')이라고 호칭한다. Action은 지정된 동작을 수행하고 그 결과인 Data set을 가지고 있다.

정의한 명령 Set는 다음과 같다.

명령	이름	설명
Combine	병합	여러 동작의 실행 결과를 한데 합친다.
Crawl	수집	서비스로부터 데이터를 수집한다.
Input	자료 입력	원본 데이터를 불러온다.
Jump	순서 이동	이전 단계의 실행 결과가 참이면, 지정한 위치 순서로 이동한다.
Output	자료 출력	파일이나 화면으로 데이터를 출력한다.
Script	스크립트	기능 확장을 위해 작성한, 스크립트를 실행한다.
Select	선택	데이터를 선택하거나 묶는다.
Store	저장소 설정	저장소에 대한 설정을 수행한다.

표 12 Task Model의 명령 Set 목록

명령 중 Input은 데이터 수집에 기준이 되는 자료를 불러서 사용할 일이 많은 특성에 따라 추가되었다. Jump의 경우 설정에 따라 이전 단계로 돌아갈 수 있는데, 이렇게 하면 이미 수집된 데이터를 원본으로 다른 데이터를 추가 수집하는 동작을 수행할 수 있다.

이러한 Task는 정보를 담고 있는 XML 파일로 저장하거나 읽어 들일 수 있다. Task 실행 Module은 Task XML 파일을 읽어서 절차에 따라 실행하며, 수집된 데이터를 Triple Store에 저장한다.

제 2 절 사용자 인터페이스 구성

Task를 설계하고 작업 수행의 절차를 표시하기 위해 사용자 인터페이스를 작성하였다. 다양한 운영체제를 사용하는 경우가 많아진 만큼, Platform은 Java 언어를 기반으로 하며 Eclipse RCP (Rich Client Platform) 위에 만들어졌다.

1. 어플리케이션 화면

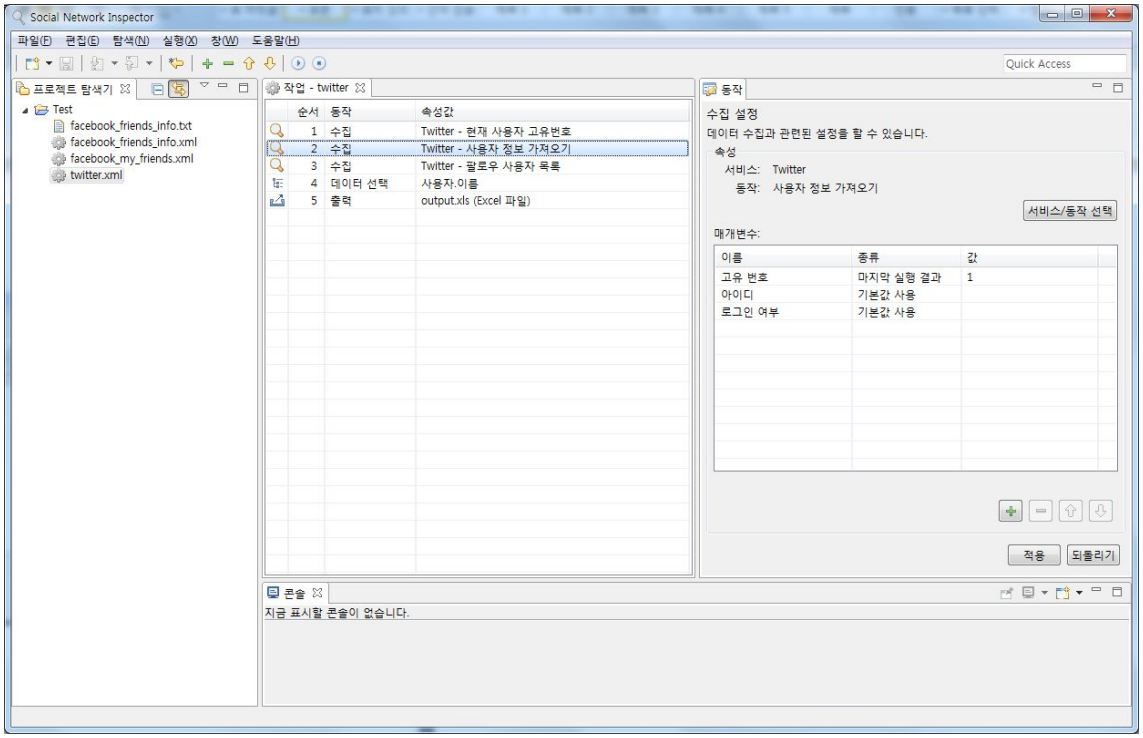


그림 5 어플리케이션 구조

사용자 인터페이스의 전체적인 화면 구성은 다음과 같다.

가. 왼쪽 - Task XML 및 관련 파일을 볼 수 있는 프로젝트 탐색기

나. 가운데 - 작업 설계 화면 (편집기)

다. 오른쪽 - 동작에 대한 세부 설정 화면

라. 아래 - Task의 실행 상태를 표시하는 결과 창

2. 작업 설계 화면

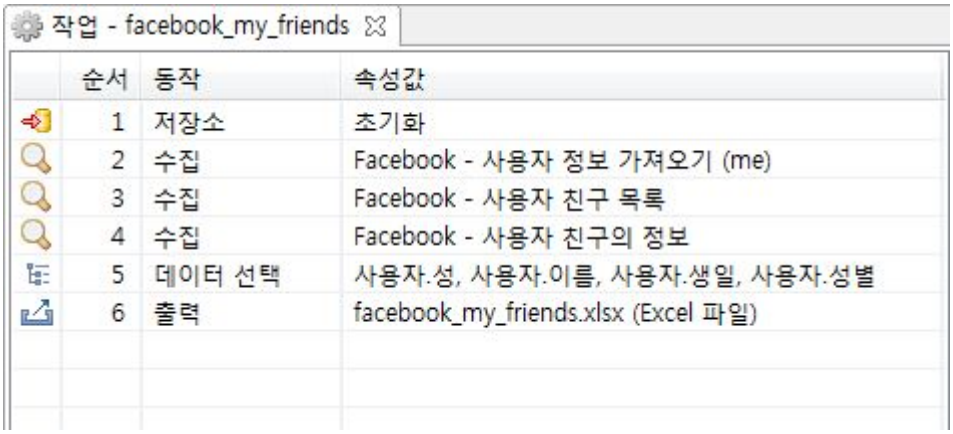


그림 6 작업 설계 화면

작업 설계 화면에서는 동작(Action)을 추가, 삭제, 순서 변경할 수 있다. 화면에는 동작을 나타내는 아이콘과 실행 순서, 동작 이름, 동작의 세부 속성값을 요약해서 표시해 준다.

동작을 추가할 경우 다음과 같은 동작 추가 대화 상자가 표시되며, 추가할 동작을 선택할 수 있다.

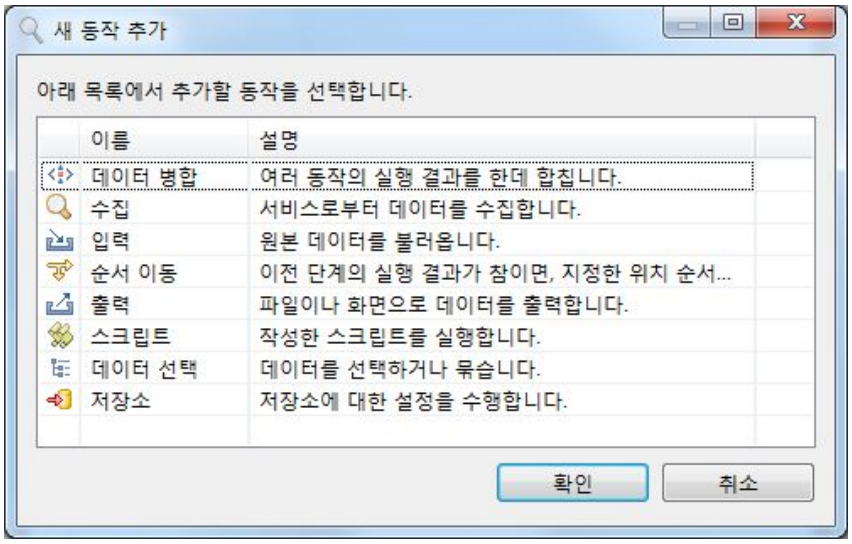


그림 7 동작 추가 대화 상자

목록에서 특정 동작을 선택할 경우 다음의 동작 세부 설정 화면이 표시된다.

3. 동작 설정 화면

동작 설정 화면에서는 각 동작과 관련된 세부 설정 값을 조절할 수 있다.

입력 설정의 경우 파일의 종류, 인코딩, 헤더의 유무, 파일 경로, 내용을 지정할 수 있다. 이 중 파일 경로와 내용은 상호 배타적인 항목으로, 파일로부터 내용을 불러오거나 입력 창에 내용을 간단하게 직접 입력할 수 있도록 구성하였다. 지원되는 파일 종류는 처리하기 용이한 텍스트 파일이나, 많이 사용되는 Excel 형식이 포함된다.

출력 설정의 경우 입력 설정과 비슷하나 덧붙이기(파일 뒤에 계속 덧붙여서 내용을 출력할 것인가) 값이 있고 직접 입력은 할 수 없으며, 헤더(머릿글) 항목을 지정할 수 있다.

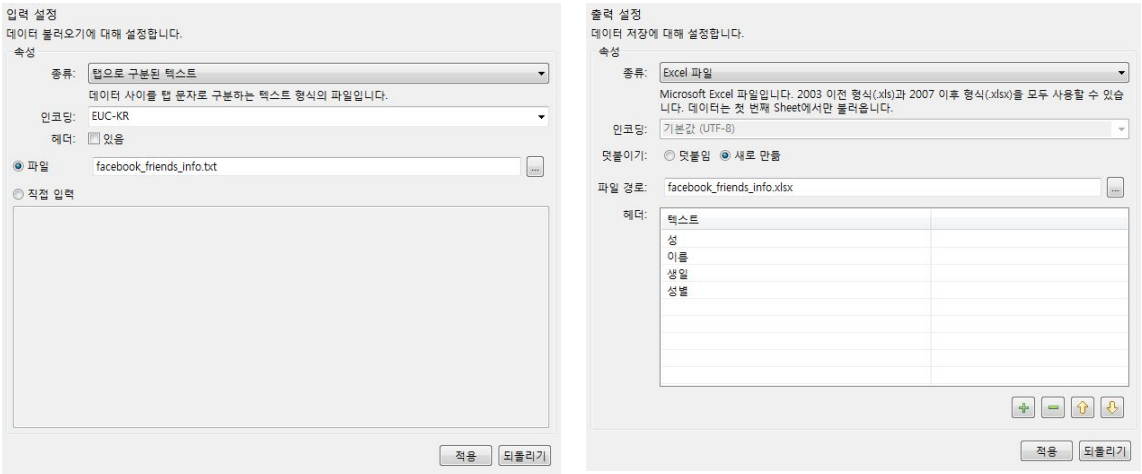


그림 8 입력 및 출력 설정 화면

수집 설정 화면에서는 서비스 및 동작을 선택할 수 있으며, 동작과 연관된 매개변수의 값을 입력할 수 있다.

동작을 선택하려면 동작 선택 대화상자에서 원하는 동작을 선택한다. 어느 동작이 어떤 서비스에 속하는지를 보여주기 위해 계층 구조로 표시하며, 동작이 어떤 일을 하는지 설명을 표시하였다.

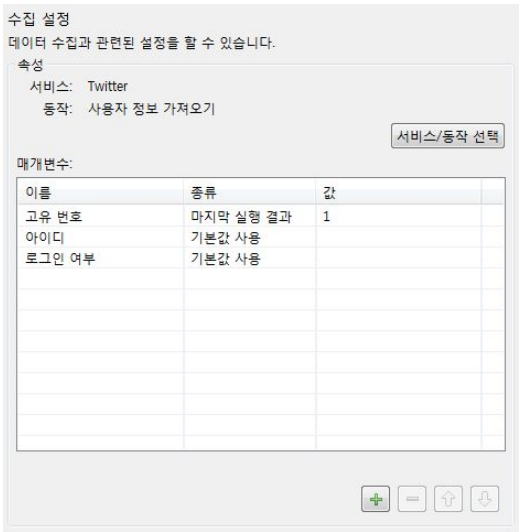


그림 9 수집 설정 화면

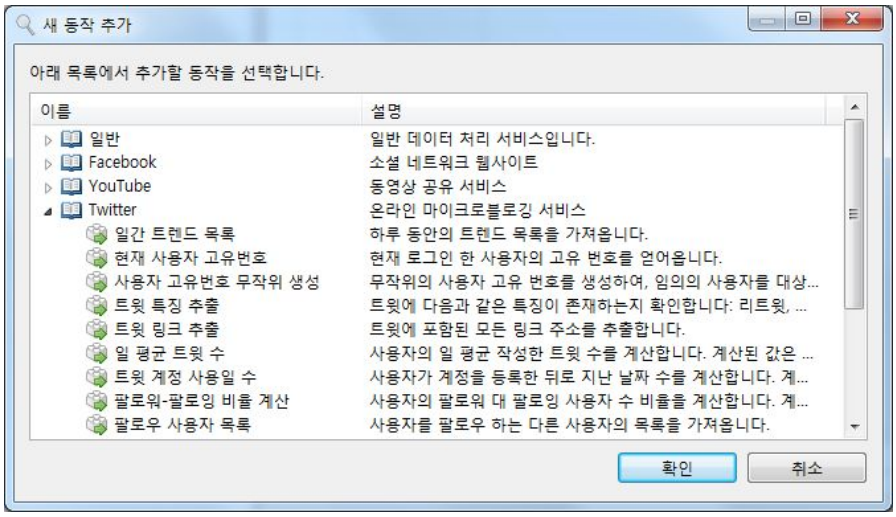


그림 10 동작 선택 대화상자

데이터 선택 설정 화면에서는 수집된 data set에서 어떤 종류의 정보를 추출할지를 결정한다. 한 번에 여러 종류의 정보를 선택할 수 있으며, 기본적인 묶기 기능(개수, 총합, 평균, 최대, 최소)을 지원한다. 값 이름을 선택할 경우 자료 항목 선택 대화상자가 표시된다.

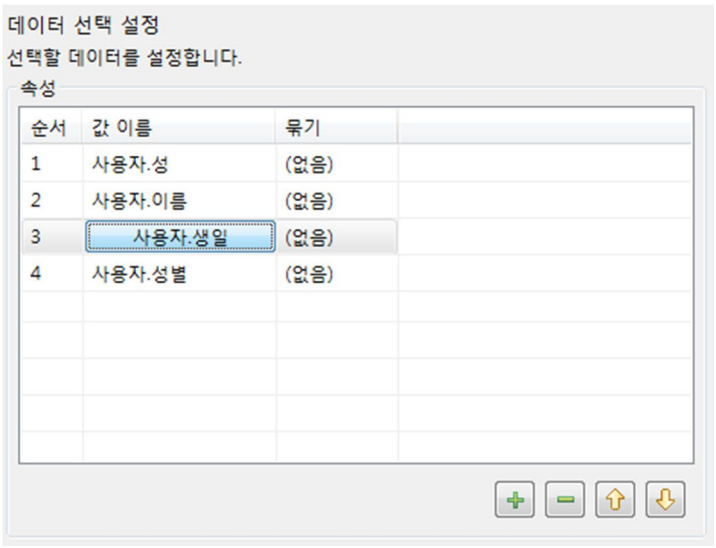


그림 11 데이터 선택 설정 화면

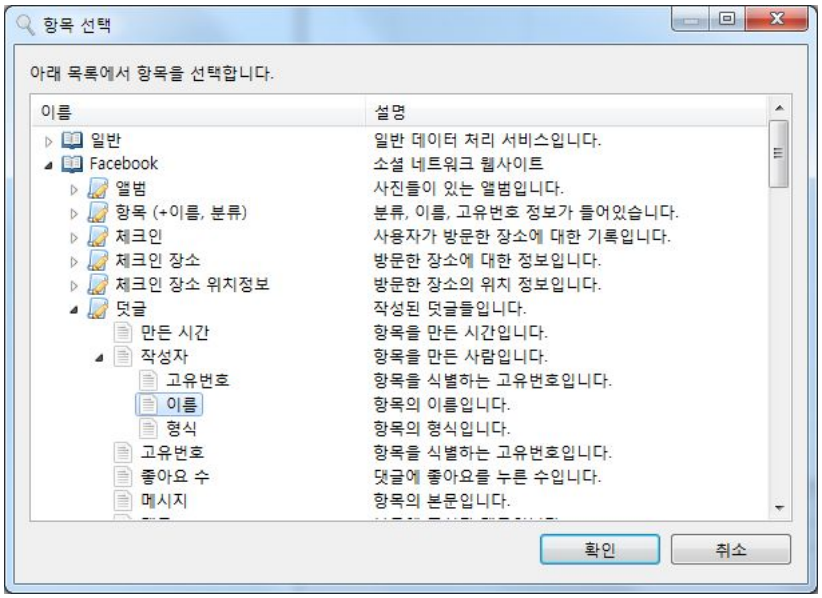


그림 12 자료 항목 선택 대화 상자

자료 항목 선택 대화 상자는 동작 선택 때와 마찬가지로 서비스와 개체, 세부 항목 구조를 위계적으로 표현하였으며, 각 항목에 대한 설명을 표시한다. 세부 항목이 단순 값이 아니라 개체일 경우 그 개체의 세부 항목을 다시 선택할 수 있도록 하였다.

이외 Combine 명령과 관련된 데이터 병합 설정 화면, Jump 명령과 관련된 순서 이동 설정 화면, Script 명령과 관련된 스크립트 설정 화면, Store 명령과 관련된 저장소 설정 화면이 있다.

제 3 절 기능의 확장

1. 서비스의 추가

Platform과 관련된 데이터 모델은 이를 서술하는 XML 파일과 실제 동작 코드로 구성되어 있다. 이러한 모델을 Platform으로부터 분리하여 플러그인 구조로 만들어, 추후 기능의 추가나 확장이 가능하다. 또한 이미 존재하는 기능을 변경하는 경우에도 들어가는 노력이 적다.

플러그인 방식으로 기능을 확장하기 위해서는 앞에서 설명한 것과 같이 두 가지 요소(서비스 서술 XML, 동작 코드)가 필요하다. 이러한 파일들은 net.theyt.sni.service.(서비스이름) 패키지 이하에 위치한다.

서비스 서술 XML의 경우 동작이나 자료형에 대한 정보를 실제 코드에 맞게 채워 주면 되며, 파일명은 service.xml로 고정되어 있다. 서비스와 구체적인 동작을 처리하는 Class는 ServiceHandler라는 이름을 가지며, 미리 지정한 구조를 따른다.

Data Model의 경우 net.theyt.sni.service.(서비스이름).model 패키지에 위치하며, 구성에 특별한 제약은 없다. 서비스나 사이트에서 제공하는 데이터에 맞게 모델을 작성하면 된다.

확장 기능의 구체적인 구현은 이미 구현된 3가지 서비스의 내용을 참조하여 구성할 수 있다.

2. Script 기능

기본적으로 제공되는 Task 동작 Set은 적은 편이다. 이는 Task 실행 Engine의 구현 용이성과 서비스 추가 가능성 때문이다. 하지만 실제 상황에서는 많은 추가기

능이나 데이터에 대한 전/후 처리가 필요하며, 이러한 요구 사항에 대한 대응을 위해 Script 기능이 구현되었다.

Script 기능은 사용자가 작성한 별도의 스크립트를 실행할 수 있는 기능이다. 현재 Platform에서 지원하는 스크립트 언어의 종류는 Beanshell, Java, Javascript, Python (JPython)의 네 종류이며 향후 추가 가능하다. 프로그래밍에 지식이 있는 사용자의 경우 Platform에서 수집, 처리한 데이터를 받아서 스크립트 내에서 가공하고, 다시 반환할 수 있다.

스크립트 내에서는 이전 단계의 결과 데이터를 'input'이라는 이름의 변수로 받을 수 있으며, 처리된 결과를 'output'이라는 이름의 변수로 돌려줄 수 있다. 또한 복잡한 데이터 처리를 위해 Ontology Triple Store에 직접 접근할 수도 있다.

스크립트를 사용한 기능 확장의 예는 다음 장에서 볼 수 있다.

제 6 장　분석 및 기능 평가

제 1 절　　연구 상의 문제점 해결

앞에서 SNS 연구 상의 문제점 네 가지를 살펴보았다. Platform의 작성에 의해 어떤 문제점이 어떻게 해결되었는지 설명한다.

1. 어떠한 데이터가 있는지 파악하기 어려움

사용자 인터페이스 상에서 동작 및 자료 선택 대화상자를 제공하여, 서비스에서 어떤 동작을 사용할 수 있고 어떤 자료를 선택할 수 있는지 표현하였다. 데이터 구조의 특징에 맞게 계층 구조로 보여주며, 각기 알기 쉬운 이름과 설명을 붙여 이해를 도왔다. 정보 항목의 계층 구조는 경로 형태(점으로 구분된)로 나타내었다.

2. 사람이 하기 어려운 작업들

원본 데이터 (XML, JSON)을 사용자에게 숨기고 자료 항목을 선택하여 그 값으로 다룰 수 있도록 하였다. 또한 Jump 동작과 같은 반복 기능을 추가하였으며, 다수의 서비스 동작에 목록 값을 입력으로 할 수 있게 구성하여 같은 동작-다른 입력에 대한 반복을 용이하게 하였다.

3. 연구 발전 과정에서 data set의 변경

서비스에 한 번 요청하여 받을 수 있는 모든 정보가 Ontology Store에 저장되므로, 이 범위에 들어가는 정보 항목은 언제든지 추가 수집 없이 확인할 수 있다. 데

이터의 확장 수집의 경우에도 이미 수집한 데이터를 기반으로 진행할 수 있도록 구성하였다.

4. API 사용상의 제한

사용자 입장에서 API 사용의 제한이나 오류를 직접 처리하는 일이 없도록 사전에 대처하였다. 시간당 호출 횟수가 정해져 있는 경우(Twitter)에는 Platform 수준에서 대응할 수 있는 방안이 없으나, 요청 간의 간격을 조정하여 해결할 수 있는 경우 최대한 효율적으로 서비스 요청을 하도록 자동으로 시간 간격을 조정하도록 하였다.

또한 임의의 서비스 오류에 의해서 원하는 데이터를 수집하지 못하였을 경우, 제한된 횟수 내에서 재시도 하도록 하여 데이터의 왜곡 가능성을 줄였다.

제 2 절 기능 적용 예

작성된 Platform의 동작을 평가하기 위해 몇 개의 예제 Task를 만들고, 실행 결과를 살펴보았다.

1. Facebook 사용자의 친구 정보 수집

Facebook에 로그인 한 현재 사용자의 친구 목록을 얻고, 그 친구들의 개인 정보(성, 이름, 생일, 성별)를 수집하는 Task이다. 일반 사용자가 이 작업을 수행할 경우 흐름은 다음과 같다.

1. 서비스에 로그인
2. 친구 목록 페이지로 이동

3. 각 친구 페이지로 이동

4. 친구의 정보 페이지에서 필요한 정보를 찾아 복사해 놓음

5. 다른 사용자에 대해 반복 (3단계로)

Platform에서 작업을 수행할 경우 흐름은 다음과 같다.

1. 저장소 초기화 및 서비스 로그인 (필요할 때 자동으로)

2. 수집: Facebook 사용자 정보 가져오기 (로그인 사용자)

3. 수집: Facebook 사용자 친구 목록

4. 수집: Facebook 사용자 친구의 정보

5. 데이터 선택: 사용자.성, 사용자.이름, 사용자.생일, 사용자.성별

6. 출력: Excel 파일

작업 흐름을 비교해 볼 때 다소간의 차이가 있는 것을 볼 수 있다. 이는 일반 사용자의 웹 인터페이스와 API의 구성 차이, 또는 사람과 프로그램의 정보 저장 방식 차이에서 기인한 것이다.

Social Network Inspector Platform은 Facebook API를 이용하고 있으며, 특정 동작을 하기 위해서는 로그인을 하거나 필요한 권한을 사용자에게 확인 받아야 한다. 이럴 경우 다음과 같은 인증 창이 표시되어 단계를 진행할 수 있다.

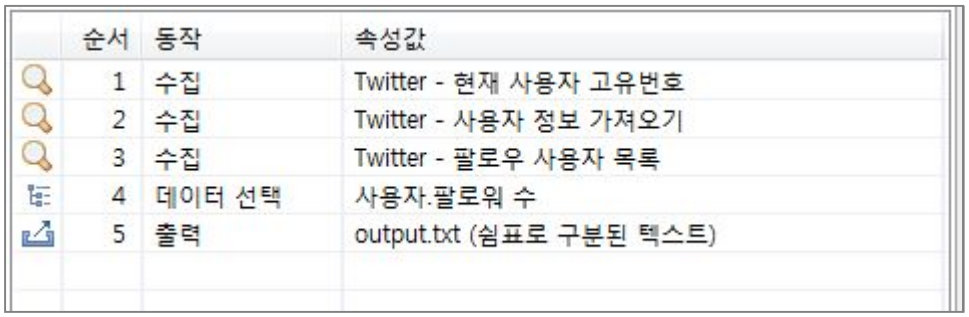
그림 13 Facebook 사용자 인증 화면

작업의 수행 결과 총 4분 20초에 걸쳐 89명(본인 + 친구 88명)의 정보를 정상적으로 읽어와 저장하였다.

2. Twitter Follow 사용자 중 Follower가 가장 많은 수 찾기

Twitter 사용자가 Follow 한 사용자 중, 가장 Follower 수가 많은 경우 몇 명인지를 알아보는 Task이다. 작업 흐름은 다음과 같다.

1. 수집: Twitter 현재 사용자 고유 번호
2. 수집: Twitter 사용자 정보 가져오기
3. 수집: Twitter Follow 사용자 목록
4. 데이터 선택: 사용자.팔로워 수 (최대값으로 묶음)
5. 출력: 텍스트 파일



	순서	동작	속성값
	1	수집	Twitter - 현재 사용자 고유번호
	2	수집	Twitter - 사용자 정보 가져오기
	3	수집	Twitter - 팔로우 사용자 목록
	4	데이터 선택	사용자.팔로워 수
	5	출력	output.txt (쉼표로 구분된 텍스트)

그림 14 구성한 Task 구조

작업의 수행 결과 총 25초 동안 수행되었으며 (대상 Follower 사용자 수 54명), 결과 값을 정상적으로 출력하였다.

3. YouTube에서 한국 사용자가 가장 많이 시청한 비디오 정보 얻기

이번에는 YouTube에서 한국 사용자가 가장 많이 시청한 비디오의 목록을 얻어오는 Task이다. 작업의 흐름은 다음과 같다.

1. 저장소: 초기화
2. 수집: YouTube 비디오 목록 가져오기 (한국 사용자의 가장 많이 시청한 비디오 목록 주소[8])
3. 출력: Ontology 데이터

수집된 모든 데이터의 구조를 볼 수 있도록 Ontology 형식으로 데이터를 출력하였다. 작업의 수행 결과 19초에 걸쳐 98개 비디오의 정보를 수집, 저장하였다.

수집된 Ontology 데이터를 자체 작성한 분석 엔진에 따라 화면에 표현한 모습은 다음과 같다. Linked data로 각 개체가 저장되어 있기 때문에, 다른 항목으로 Hyperlink를 통해 이동하면 해당 항목의 정보를 확인할 수 있다.

[8] https://gdata.youtube.com/feeds/api/standardfeeds/kr/most_viewed

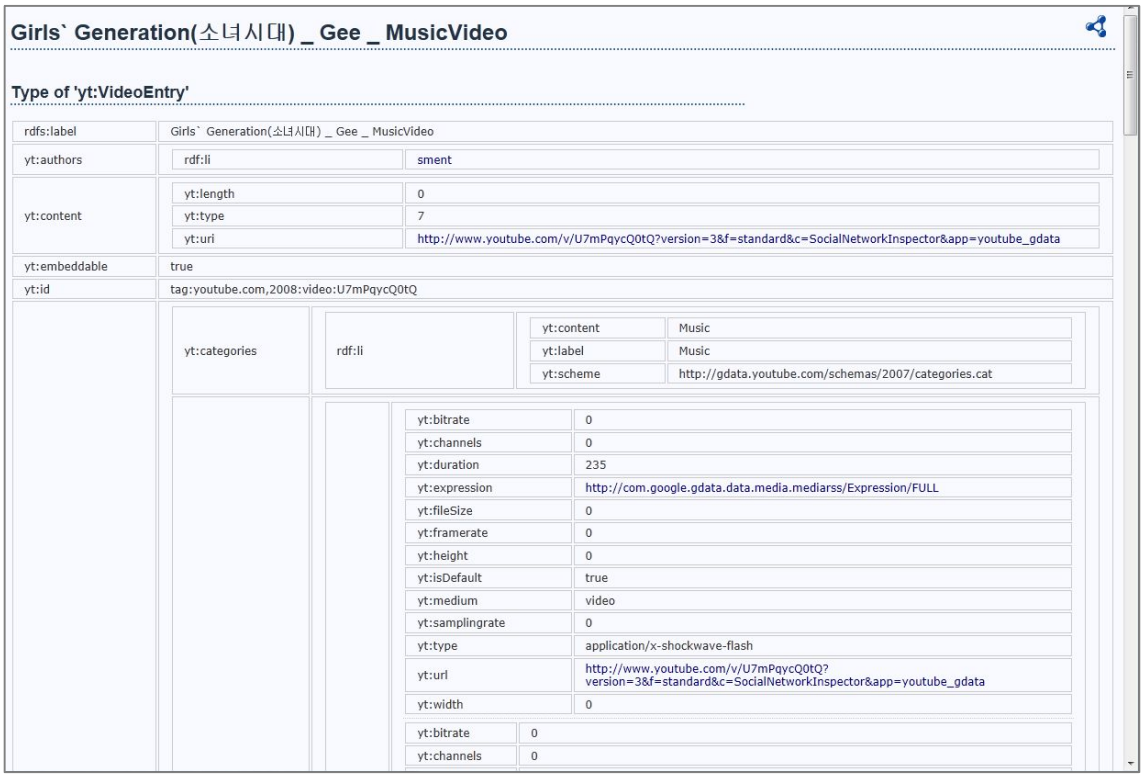


그림 15 YouTube 동영상 정보 Ontology 구조

4. 게시물 작성 시간에 따른 사용자 유사도 계산

Script 기능 확장을 이용한 복합적인 데이터 분석 예제이다. SNS에 대한 연구는 이제 단순 데이터 수집과 통계를 넘어 [29]에서와 같이 Homophily나 네트워크 분석 (Hub, Authority, Influence 등)과 같은 복합적인 데이터 분석을 필요로 하고 있다. 이 예제에서는 그 중 일부로, Twitter에서 사용자가 Tweet을 작성한 시간의 분포를 기준으로 사용자 간의 유사도를 계산하도록 한다. 계산 방법은 다음과 같다.

1. 여러 사용자에 대해 Tweet을 수집한다.
2. 각 사용자의 Tweet을 시간 별로 (24시간) 분류하여 수를 센다.
3. 이 수를 각 사용자의 총 Tweet 수로 나누어 정규화 시킨다. 이를 Tweet 작성

시간 별 분류 Vector라고 부른다.

4. 기준 사용자와 다른 사용자의 Tweet 작성 시간 별 분류 Vector에 대해 Cosine 유사도를 계산한다.

$$\frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}}$$

식은 앞과 같으며 A와 B는 각 사용자의 Tweet 작성 시간 별 분류 Vector이다.

다음은 본인을 기준으로 (본인 포함) 7명의 사용자에 대해 Cosine 유사도를 계산한 것이다.

사용자	유사도
기준 (본인)	1.0000
사용자 1	0.8317
사용자 2	0.7276
사용자 3	0.8536
사용자 4	0.8032
사용자 5	0.7610
사용자 6	0.7677

표 13 사용자 유사도 계산 결과

본인 자신의 경우에는 데이터가 동일하므로 유사도 값이 1이며, 사용자에 따라 유사도 값에 차이가 있는 것을 볼 수 있다. 다만 사용자 전원이 한국 시간대의 사용자이기 때문에 유사도의 편차가 심하지는 않다. 이러한 분석을 토대로 정보 흐름이나 영향력 평가를 추가로 수행할 수 있다.

5. 게시물 본문의 한국어 색인어 분석

이번에는 외부 프로그램을 이용하여, 사용자가 작성한 게시물로부터 색인어를 추출하도록 한다. 색인어 추출에는 한국어 형태소 분석기[9]를 사용하였다.

트위터 도우미 계정 (@dowoomi)의 최근 Tweet 200개를 대상으로 본문을 수집하고, 이를 형태소 분석기를 통해 색인어를 추출하였다. 분석기에서 Tweet 글을 한번에 분석하도록 할 수도 있으나 Tweet의 특성상 연속된 글이라고 보기 어려우므로 각 Tweet을 별도로 하나씩 처리하였다. 각 트윗의 색인어 중 색인어 점수가 가장 높은 것을 더하여 빈도를 세고 정렬한 결과는 다음과 같다. (빈도수 3 이상)

단어	빈도
트위터	69
t.co/va7dHVRm	13
현재	13
양해	11
비밀번호	8
운영원칙	6
불편	5
응용	5
계정	4
사용자	4
이미지	4
핸드폰	4
도움말	3
문제	3
비공개	3
저희	3
타임라인	3

[9] Korean Language Technology (KLT): http://nlp.kookmin.ac.kr/HAM/kor/index.html

화면	3

표 14 Tweet 색인어 분석 결과

원 계정의 특성(트위터 사용자 지원 계정)에 맞는 단어들이 상위에 올라와 있는 것을 볼 수 있다. 결과 중 웹 주소는 사용자 지원과 관련된 설명 페이지이다.

이러한 분석을 통해 기계적인 방법으로 텍스트 분석을 수행하여, 사용자 분류를 수행할 수 있을 것으로 판단된다.

제 7 장 결론

제 1 절 요약

본 논문에서는 SNS의 데이터를 수집하고 분석하는 도구인 Social Network Inspector Platform을 설계, 작성하였다. 이를 위해 먼저 각 SNS의 특징, 주요 동작 및 자료 구조를 분석하여 관련 Model을 작성하였다. 또한 기존의 연구로부터 SNS를 대상으로 자료 수집을 하는 과정에서 발생할 수 있는 문제점을 발견하고 대책을 찾아 보았다.

Platform은 크게 Task 실행 Module과, 사용자 인터페이스로 구성되었다. Task 실행 Module은 구성한 Task를 실행하고 수집된 결과를 저장하는 역할을 하며, 사용자 인터페이스는 Task를 설계하고 실행 결과를 지켜볼 수 있도록 만들어졌다. 이 두 부분이 동작하기 위하여 그 기반에는 Service Model, Data Model, Task 및 Action Model이 각각의 정보를 유지하고 있다.

Platform의 작성 결과 실제로 동작 가능한 Task를 설계하고 실행할 수 있었으며, 데이터도 정상적으로 수집되었다.

제 2 절 연구의 시사점

(Web) Crawler로부터 보았을 때, 지금까지 여러 종류의 자동화 된 데이터 수집 프로그램이 존재했다. 하지만 SNS를 대상으로 특화 하였으면서 Task에 대해서 범용적인 Platform에 대해서는 아직까지 이와 관련된 진행 성과가 보고된 바 없다.

이 Platform을 이용하여 SNS에 대한 학생이나 연구자의 연구 진척에 도움이 될 것이라고 기대한다. 특히 처음에 목표로 하였던 연구 문제 설정 단계에서의 활용에

는 문제가 없으며, 다소의 기능을 보완한다면 본격적인 데이터 수집과 분석에도 이용될 수 있을 것이라고 판단된다.

제 3 절 연구의 한계 및 제언

여러 사례 수집과 고민에도 불구하고 Platform이나 설계 상에 미흡한 부분이 존재하며, 앞으로 개선해 나가야 할 바라고 생각된다.

1. 동작 및 자료 구조 표현의 개선

서비스와 관련된 동작 및 자료 구조의 이해를 돕기 위해 계층적인 형태로 화면에 표현하였다. 하지만 자료 구조의 경우 그 종류가 많으며, 세부 항목까지 합하면 그 수는 서비스당 수백 개가 넘는다. SNS에 대한 이해가 적을 경우 원하는 항목이 어디 있는지 찾는 것도 쉬운 과제가 아닐 수 있다.

이러한 문제를 해결하기 위해서는 '자주 사용하는 자료 항목'을 선정하여 목록의 위편에 표시하거나 잘 보이도록 강조하는 방안이 있다.

2. 병렬 데이터 수집

SNS와 관련된 연구의 유행이 대량 데이터 수집으로 가고 있으며, Big data 분석은 학계에서 현재 각광받는 분야 중 하나이다. 앞에서 서술했다시피 하나의 컴퓨터에서 자료를 요청하는 데 제한을 거는 경우가 있어, 대량 데이터를 지속적으로 수집할 경우 실제로 데이터를 수집하는 시간 효율은 낮아지고는 한다.

이를 해결하려면 여러 대의 컴퓨터를 이용해서 병렬로 데이터를 수집할 수 있다. Platform의 경우 수집된 데이터를 중앙 Database에 모으도록 설정[10]하고 여러 컴퓨터에서 실행한다면 병렬 데이터 수집을 할 수 있다. 하지만 이럴 경우 주의해야 할 점은 원본 data set이나 수집된 data가 겹치지 않도록 구획을 잘 분할하는 것이다.

개선 사항이 남아있지만 처음의 목적을 달성한 만큼, 시험을 거쳐 연구자들에게 널리 사용될 수 있다면 관련 분야 연구에 큰 도움이 될 것이라 기대한다.

[10] Ontology Store의 대상 경로를 네트워크 상의 다른 컴퓨터 경로로 지정할 수 있다.

참고 문헌

[1] Tim O'Reilly, "What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software," *International Journal of Digital Economics*, no. 65, pp. 17-37, 2007.

[2] PCMag.com. (2012, Apr.) Facebook Now Totals 901 Million Users, Profits Slip. [Online]. http://www.pcmag.com/article2/0,2817,2403410,00.asp

[3] mediabistro. (2012, Feb.) Twitter To Surpass 500 Million Registered Users On Wednesday. [Online]. http://www.mediabistro.com/alltwitter/500-million-registered-users_b18842

[4] Miniwatts Marketing Group. (2011) Internet World Stats. [Online]. http://www.internetworldstats.com/stats.htm

[5] Boram Park, Changjin Han, and Namjun Kang, "Network Topology of Social Media and Information Behavior Pattern," in *The 1st Conference on Pioneering Convergence Technologies*, Jeju, Korea, 2011, pp. 56-59.

[6] Eunbin Kim and Yong-tae Hwang, "Communication Networks on Twitter: How Many of Twitter Followers Does One Actually Communicate with?," in *The 1st Conference on Pioneering Convergence Technologies*, Jeju, Korea, 2011, pp. 52-55.

[7] (1996) PC Magazine. [Online]. http://www.pcmag.com/encyclopedia_term/0,2542,t=application+ program

ming+ interface&i=37856,00.asp

[8] Djamal Benslimane, Schahram Dustdar, and Amit Sheth, "Services Mashups: The New Generation of Web Applications," *IEEE Internet Computing*, vol. 12, no. 5, pp. 13-15, Sep. 2008.

[9] Sunilkumar Peenikal, "Mashups and the Enterprise," Mphasis, New York, USA, White Paper 2009.

[10] Mei Kobayashi and Koichi Takeda, "Information retrieval on the web," *ACM Computing Surveys*, vol. 2, no. 32, pp. 144-173, June 2000.

[11] Helena Deards. (2009) Twitter first off the mark with Hudson plane crash coverage. [Online]. http://www.editorsweblog.org/multimedia/2009/01/twitter_first_off_the_mark_with_hudson_p.php

[12] Jason Kincaid. (2009, Mar.) Foursquare Scores Despite Its Flaws. [Online]. http://www.washingtonpost.com/wp-dyn/content/article/2009/03/18/AR2009031802819.html

[13] D. Zhao and B. M. Rosson, "How and Why People Twitter: The Role that Micro-blogging Plays in Informal Communication at Work," in *Proceedings of the ACM 2009 International Conference on Supporting Group Work*, New York, USA, 2009, pp. 243-252.

[14] Boris Veldhuijzen van Zanten. (2010, Sep.) The Next Web - Twitter Statistics: 82% of Twitter users have less than 350 followers. [Online]. http://thenextweb.com/socialmedia/2010/09/30/twitter-statistics-82-of-

twitter-users-have-less-than-350-followers/

[15] d. boyd, S. Golder, and G. Lotan, "Tweet, Tweet, Retweet: Conversational Aspects of Retweeting on Twitter," in *Proceedings of the 43rd Hawaii International Converence on System Sciences*, Kauai, Hawaii, USA, 2010.

[16] Twitter, Inc. (2010, May) Twitter API Documentation. [Online]. http://dev.twitter.com/doc/post/statuses/retweet/:id

[17] Maggie Shiels. (2009, June) BBC News: Web slows after Jackson's death. [Online]. http://news.bbc.co.uk/2/hi/technology/8120324.stm

[18] Christo Wilson, Bryce Boe, Alessandra Sala, Krishna P. N. Puttaswamy, and Ben Y. Zhao, "User Interactions in Social Networks and their Implications," in *Procedings of EuroSys 2009*, Nuremberg, Germany, 2009.

[19] Vijay Erramilli, Xiaoyuan Yang, and Pablo Rodriguez, "Explore what-if scenarios with SONG: Social Network Write Generator," Spain, 2011.

[20] A. Java, X. Song, T. Finin, and B. Tseng, "Why We Twitter: Understanding Microblogging Usage and Communities," in *Proceeedings of the Joint 9th WEBKDD and 1st SNA-KDD Workshop*, San Jose, Califonia, USA, 2007.

[21] Balachander Krishnamurthy, Phillipa Gill, and Martin Arlitt, "A Few Chirps About Twitter," in *Proceedings of the First Workshop on Online Social Networks*, New York, USA, 2008, pp. 19-24.

[22] B. A. Huberman, D. Romero, and F. Wo, "Social Networks that Matter: Twitter Under the Microscope," *First Monday*, no. 14, pp. 1-9, Jan. 2009.

[23] Courtenay Honeycutt and Susan C. Herring, "Beyond Microblogging: Conversation and Collaboration via Twitter," in *Proceedings of 42nd Hawaii International Conference on System Sciences*, USA, 2009, pp. 1-10.

[24] Alex Leavitt, Evan Burchard, David Fisher, and Sam Gilbert, "The Influentials: New Approaches for Analyzing Influence on Twitter," Web Ecology Project, 2009.

[25] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a social network or a news media?," in *Proceedings of the 19th International World Wide Web Conference*, USA, 2010.

[26] M. Cha, H. Haddadi, F. Benevenuto, and K. Gummadi, "Measuring user influence in twitter: The million follower fallacy," in *Proceedings of the 4th International Conference on Weblogs and Social Media*, USA, 2010.

[27] Jeff Young. (2010, May) The Chronicle: Researchers Find 'Million-Follower Fallacy' in Twitter. [Online]. http://chronicle.com/blogs/wiredcampus/researchers-find-million-follower-fallacy-in-twitter/24290

[28] (2011, Jan.) Twitter Fan Wiki - Apps. [Online]. http://twitter.pbworks.com/w/page/1779726/Apps

[29] De Munmun Choudhury, Hari Sundaram, Ajita John, Doree Duncan Seligmann, and Aisling Kelliher, ""Birds of a Feather": Does User Homophily Impact Information Diffusion in Social Media?," in

arXiv:1006.1702, 2010.

[30] Benny Evangelista. (2010, Sep.) San Francisco Chronicle: Twitter now has 145 million users after growth spurt. [Online]. http://www.sfgate.com/cgi-bin/blogs/techchron/detail?entry_id=71579

[31] Twitter, Inc. Rate Limiting FAQ. [Online]. http://dev.twitter.com/pages/rate_limiting_faq

[32] Infochimps, Inc. (2011, Mar.) Twitter Census. [Online]. http://www.infochimps.com/collections/twitter-census

Abstract

Social Network Inspector: A Platform for Data-driven Research of Social Network Services

Hwang, Yong-tae

Department of Digital Contents Convergence

The Graduate School

Seoul National University

The development of the Social Network Services was induced by the era of 'Web 2.0' and increasing use of the mobile devices. Enormous data is being created and researchers are paying attention to them. These data can be accessed by Application Programming Interface (API) which is publicly opened by service providers. Most of data-driven research requires program called 'crawler' due to large amount of data.

Researches on SNS require interdisciplinary approach which typically requires more people than single subject research. However, there are many cases which require achievement by smaller group of members in early stage of the study. Also, they most likely rely on manual data collecting, it may cause unwanted errors.

In this paper, data collecting and analysis platform for researching Social Network Services has proposed. First, previous studies were reviewed on

"types of data", and "collecting methods". Next, API actions and data types provided by services were strategically modeled. Also, problems and solutions which could be occurred while conducting SNS researches were addressed.

Based on the proceeding research, extensible and easy-to-use data crawling platform was made. By using this platform, common procedure for SNS researches can be simplified, and most of prior mentioned problems were resolved. Finally, several case study for real-world research tasks, performance test and data validation were achieved.

Keywords: Social Network Services (SNS), Application Programming Interface (API), Crawler, Platform

Student Number: 2010-22686