지능화 기술 생태계 분석을 위한 데이터 수집 및 가공

ETRI 한국전자통신연구원
Electronics and Telecommunications Research Institute

목 차

핵심 요약

연구의 필요성 및 목적

◈ 오픈소스 소프트웨어의 중요성 및 활용성 증가

- 오픈소스 소프트웨어(이하 오픈소스)는 이미 대부분의 산업의 주요 소프트웨어 개발 및 사용 원천으로 운영되고 있으며, 가트너 조사 결과 90% 이상의 응답자가 응용 프로그램 개발에 오픈소스를 활용하고 있다고 보고됨

- 최신기술습득, 개발기간단축, 개발비용절감, 중복개발예방 등 SW 개발 효율성은 물론 결과물 품질 보증, 개발자 역량 강화, 성과 공유 및 확산 등 SW 개발 효과성도 확대되고 있음

◈ 오픈소스 기술 생태계의 이해 필요

- 오픈소스는 자율적이고 자발적으로 성장해나가는 자원으로, 자원이 체계적으로 자기조직화(self-organizing)되는 분야인지를 사람과 기술 대상의 생태계 관점에서 구성 여부 및 발전체계를 이해하고 확인할 필요가 있음

  - 오픈소스는 결국 개발자가 기술을 개발하는 부분으로, 누가 개발하고 있는가? 무엇이 개발되고 있는가? 어떤 형태로 개발되고 있는가? 시장 및 산업 수준으로 확장되고 있는가? 등 사람과 기술을 대상으로 생태계를 분석하고 이해하는 것이 적합함

◈ 지능화 기술 생태계 구축 전략과 분석 방법의 필요

- 국내 정보화 기술은 국가 주도의 기술 및 시스템 개발로 이루어져왔으며, 과거 통신 기술의 정보화 기반뿐만 아니라 최근의 데이터 개방 및 4차 산업혁명 기술까지 정부의 로드맵 구축과 투자로 개발되어오고 있음

- 대량의 오픈소스가 개발되고 발전되는 기술 생태계를 살펴보기 위해, 오픈소스를 자동으로 수집하고, 주요기술·주요 개발자·특정기술에 따른 기술유형, 개발현황을 분석하는 빅데이터 분석 방법론을 개발함

  - 오픈소스를 관리하는 깃허브를 중심으로, 깃허브에 등록된 상위 인지도 기준의 주요 기술을 분석하고, 빅테크 기업 위주의 주요 개발자의 오픈소스 개발 현황, 그리고 특정 기술 단위에서 오픈소스가 개발되고 있는 현황을 분석함

깃허브 오픈소스 데이터 분석 방법론 개발

분석절차 및 방법

- 깃허브의 저장소 분석을 통한 기술현황을 살펴보기 위해, 데이터 수집 및 전처리, 기술통계 및 유형 분석, 시사점 도출의 순서로 진행함
  - (데이터 수집) 총 세 가지 데이터베이스를 구축을 목표로 하며, (1) 주요 저장소를 분석하기 위해 상위 인지도를 나타내는 스타 수 속성으로 저장소를 수집하고, (2) 기업의 저장소 개발 현황을 살펴보기 위해 기업이 소유자인 저장소를 추출하며, (3) 검색기술의 개발현황을 살펴보기 위해 기술 토픽에 해당하는 저장소를 출력함
  - (분석 방법) 다수의 기업 또는 기술 저장소의 내용을 확인하기 위해 항목(브랜치, 풀 리퀘스트, 스타 수 등)들의 기술통계를 제시하고, 현재 개발되고 있는 오픈소스의 기술내용 유형을 분석하기 위해, 저장소의 토픽들을 사용하여 k-means와 DBSCAN(Density-Based Spatial Clustering of Applications with Noise) 클러스터링를 수행하고 기술유형을 도출함

깃허브 오픈소스 저장소 데이터 분석 결과

깃허브 주요 저장소 분석

- 상위 인지도(스타 수 기준) 저장소를 대상으로, 상위 저장소 개발 현황과 저장소 주요 토픽을 분석하여 웹 기술, 기계학습 위주의 28가지 클러스터를 도출하였음
  - (프로그램 개발자 중심의 생태계) 깃허브의 저장소들은 대부분 프로그램 개발자 또는 사용자가 많으며, Front-end, Back-end 중심의 범용적 어플리케이션, 플랫폼 개발이 주를 이루고 있음
  - (Front-end 및 Back-end 오픈소스 조합을 통한 웹 서비스 개발) Front-end, Back-end 기술과 관련한 오픈소스들이 깃허브에 공개됨에 따라, 관련 오픈소스 패키지 및 프로그램을 이용하고 조합하여, 개발상황에 적합한 웹 서비스를 개발하고 있음을 확인함
  - (소프트웨어 개발, 활용, 교육 등 협력 및 공유의 공간) 개발·활용·교육 측면에서 검증과 개발을 반복하고, 새로운 프로그래머를 양성하는 등 소프트웨어 R&D 모델인 나선형(spiral) 구조의 발전 생태계를 자체적으로 형성하고 있음을 발견함

빅테크 기업 저장소 분석

- 국제적 빅테크 기업인 구글, MS, 인텔, 페이스북, 애플, 아마존, 바이두, 알리바바, 텐센트, 네이버, 카카오, 삼성전자의 저장소 현황 및 오픈소스 기술현황을 분석함

- 기업 활동 특성에 따른 오픈소스 활동과 자체 플랫폼 주도의 오픈소스 활동이 이루어지고 있음을 확인하였으나, 대부분 기반 기술 위주의 오픈소스만 공개되어 실제 기업의 시장이나 사업 지식을 얻기에는 한계점이 있음

미래기술 저장소 분석

- 특정 기술토픽을 지닌 오픈소스 저장소를 추출하여 개발 기술내용 및 기업현황을 탐색하고, 기술 및 시스템 구조도 상의 오픈소스 적용 분야 도출 및 기술 로드맵 개발을 통해 기술 생태계를 민간과 국가가 협력하여 기획할 수 있는 방안을 제시함
  - 자율주행차(autonomous vehicle), 메타버스(metaverse) 토픽을 지닌 저장소를 분석하여, 요소기술의 오픈소스 개발 현황을 탐색하고 주요 기술용어를 도출함
  - 미래 기술 대상에 대한 기반 기술을 확인하였으며, 오픈소스 개발 수 또는 수준에 따른 기술평가 가능성을 제시하고, 국가와 민간이 주도해야할 기술개발 영역을 도출하고 기획하는데 활용가능한 기술로드맵 작성 방식을 개발함

시사점 및 개선방향

오픈소스 저장소 데이터 분석의 시사점

- 깃허브에서 제공하는 오픈소스의 다양한 속성 및 기술 정보를 이용한 빅데이터 분석 가용성을 확인하였으며, 오픈소스 주요 기술 및 기업을 탐색하고 민간 기술 생태계로부터의 기술기획 가능성을 제시하여, 오픈소스 기술 생태계 발전 및 조성 방식을 탐색함

오픈소스 저장소 분석 한계점 및 개선방향

- 깃허브 저장소는 개발자가 자율적으로 저장소를 구성되는 비정형적 형식의 데이터로 일관적이고 체계적인 분석이 어려운 한계점이 존재하며, 산업 및 시장 응용기술이 아닌 기반 기술 위주의 저장소로 운영되고 있어 오픈소스가 실제 제품 및 서비스 개발에 활용되는 현황 분석은 어려움

- 소프트웨어 기술분야 및 기술용어를 탐색하고 민간으로부터 개발되고 있는 현황을 분석하는데 의의가 크며, 민간에서부터의 기술 현황을 반영한 국가적 지능화 기술 생태계 예측 용도로 활용하는 것이 오픈소스의 효과적인 활용방안이라 기대됨
  - 자율주행차 또는 메타버스의 오픈소스 저장소에서 도출된 주요 기술이 해당 제품 및 산업의 기술 구조도에 적합하게 대응되는 사례분석에서도 보듯이, 오픈소스 저장소는 기반 기술 지식의 탐색적 용도로 활용하고 기술기획 및 예측에 응용하여 가치를 확대하는 것이 바람직해보임

Ⅰ 서 론

1 연구배경

가. 지능화 기술 산업 현황

◈ 소프트웨어 기술의 발전

- 컴퓨팅 하드웨어의 초고성능화와 네트워크의 초고속화에 따라 4차 산업혁명 시대에 들어 소프트웨어의 기술수준이 급속도로 발전하고 있음
- 과거 인공지능(AI: Artificial Intelligence)과 관련한 알고리즘이 컴퓨팅 하드웨어의 발전에 미치지 못해 정체되었던 연구들이 근 10년 동안 딥러닝(deep learning)을 대표로 하는 기계학습(machine learning) 알고리즘과 같은 지능화 연구와 개발이 활발히 이루어지고 있음(이진휘, 2020)
- 하드웨어 기술이 몸(body), 소프트웨어 기술이 머리(brain)를 이루면서, 체(體)를 바탕으로 하는 지능화(知能化) 기술이 발전하고 있으며, 특히 데이터(data)의 수집 및 가공과 관련한 처리(processing), 분석(analysis), 표현(representation) 알고리즘 연구가 시스템 차원으로 확장되고 있음
- 국가적으로도 이와 같은 지능화 기술에 대한 관심이 높아지고 있으며, 소프트웨어 기술을 중심으로 하는 제품·서비스·사회혁신이 가속화되면서, 지능화 기술에 대한 모니터링 및 개발계획을 수립하는 것이 경쟁력 확보를 위해 중요해지고 있음

◈ 개방형 체제의 지능화 기술 생태계

- 개방형 체제의 연구개발과 기술혁신이 소프트웨어 분야에 자리잡게 되면서, 개방형 혁신(open innovation)에 따른 오픈데이터(open data), 오픈소스(open source), 오픈억세스(open access) 등이 개방형 생태계(open ecosystem)의 보고(寶庫; repository)로 떠오르고 있음(김성민 외, 2020)
  - 연구개발(R&D)과 기술혁신에 있어서 개방형(openness)이라는 키워드가 2000년대 초반부터 발아되기 시작(Chesbrough, 2003)
  - 특히, 소프트웨어(SW: Software) 관련 기술은 무형의 지능 재산이라는 점에서 공유와 개선에 용이하여, 개발자들은 초기부터 개방에 따른 집단지성을 극대화에 중점을

두어 오픈소스의 데이터 규모가 폭발적으로 증가하고 있음

- 오픈소스 소프트웨어(open source SW)는 데이터 규모 증가뿐만 아니라 협업과 경쟁에 기반하여 개선 가치를 극대화하고 있으며, 학계·산업계를 아울러 연구개발, 교육 등 활용도가 지속적으로 증가하고 있다는 점에서 활용 가치가 높음(김성민 외, 2020; 이진휘, 2020; 이현진, 2021)
  - 세계 빅테크 기업들도 오픈소스를 이용하여 소프트웨어의 공개 및 협업 발전을 주도하고 있으며, 대학이나 연구소에서도 소프트웨어 활용 교육, 연구 및 개발에 대해 자체 검증과 함께 사실상 표준과 같이 오픈소스를 사용하고 있음

나. 오픈소스 소프트웨어 현황

◆ 소프트웨어 기술의 발전

- 소프트웨어란 컴퓨터·통신·자동화 등의 장비와 그 주변장치에 대하여 명령·제어·입력·처리·저장·출력·상호작용이 가능하도록 하게 하는 지시·명령의 집합과 이를 작성하기 위해 사용된 기술서나 기타 관련 자료로 정의됨(소프트웨어산업진흥법 제2조제1호)
  - 여기서 컴퓨터와 상호작용하는 기술서는 대부분 프로그램으로 작성된 기능 로직이기 때문에, 소프트웨어와 프로그램은 유사하게 볼 수 있으나 엄밀히는 소프트웨어가 더 큰 범위로 볼 수 있음
- 즉, 소프트웨어 없이 물리적 기기를 운용할 수 없으며, 소프트웨어는 최근 개발되고 있는 컴퓨터는 물론 전자 장비들의 작동 매뉴얼로 작동하기 때문에 현대 사회에 필수불가결한 요소임
  - 그러나 기계와 장비가 받아들이는 언어는 사람과 달라 이진 코드(0 아니면 1)로 이루어져 있으며, 또 그 코드를 사람의 언어로 번역되는 프로그래밍 언어들이 다양하게 개발되면서 소프트웨어가 발전하고 있음
- 2000년대 이후 인터넷 발전에 따른 웹 발전과 스마트폰 발전에 따른 모바일 웹의 발전으로 다양한 웹 프로그래밍 언어가 개발되면서, 소프트웨어 관련 사업과 시장이 급속도로 확대되었음(최무이 외, 2020)

◆ 오픈소스 소프트웨어의 사용

- 오픈소스 소프트웨어는 자유 SW 개념[1]에서 진화한 개념으로, 소스코드를

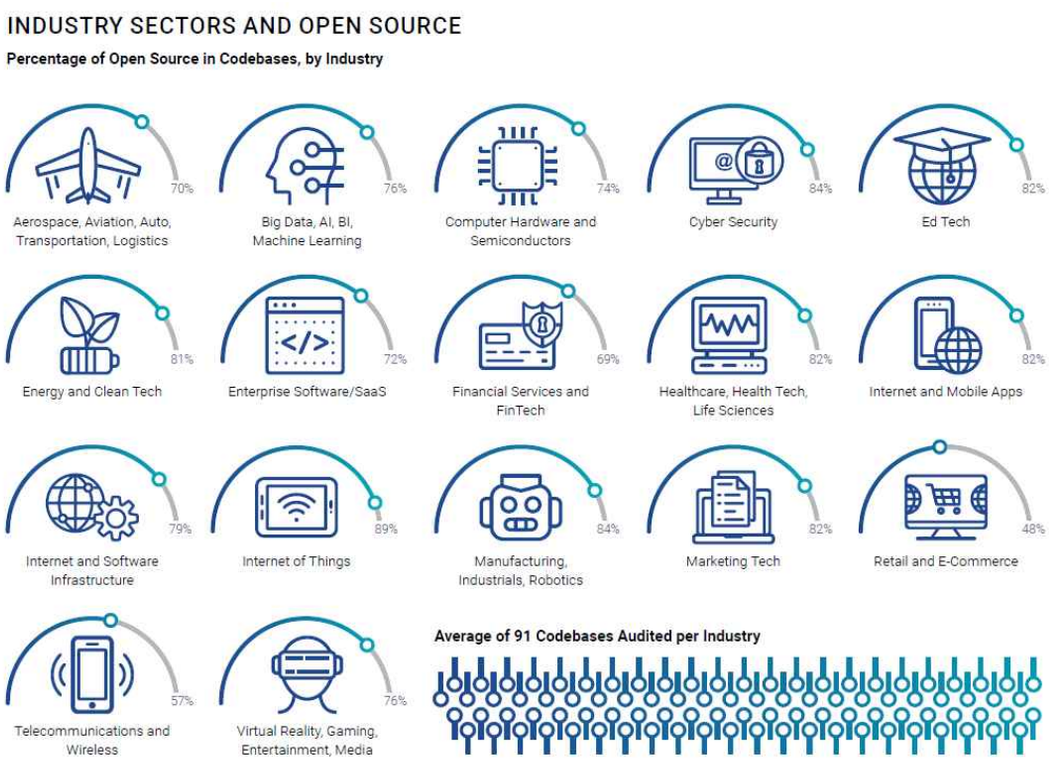
1) 1980년대 리처드 스톨만 주도로 누구나 자유롭게 SW를 사용할 권리를 부여하기 위한 운동

공개하여 개발하는 개방형 SW 개발 방식의 확산을 의미함 (권영환, 2020)

- 기업, 대학, 연구소 등 소프트웨어 및 프로그램 개발을 위해서 오픈소스 플랫폼, 프레임워크, 패키지 등을 사용하고 있음

- 오픈소스는 이미 대부분 산업의 주요 소프트웨어 개발 및 사용 원천으로 운영되고 있으며, 가트너 조사 결과 90% 이상의 응답자가 응용 프로그램 개발에 오픈소스를 활용(Gartner, 2019)하고 있다고 제시함

그림 1-1 산업별 오픈소스 활용도



※ 이미지 출처: Synopsys (2021)

◆ 오픈소스 소프트웨어의 혜택

- 소스코드 공개에 따라 누구나 사용가능하여, 개발에 걸리는 시간이 단축되고, 필요에 따라 수정 및 재공유가 가능함에 따라 네트워크 효과가 커짐

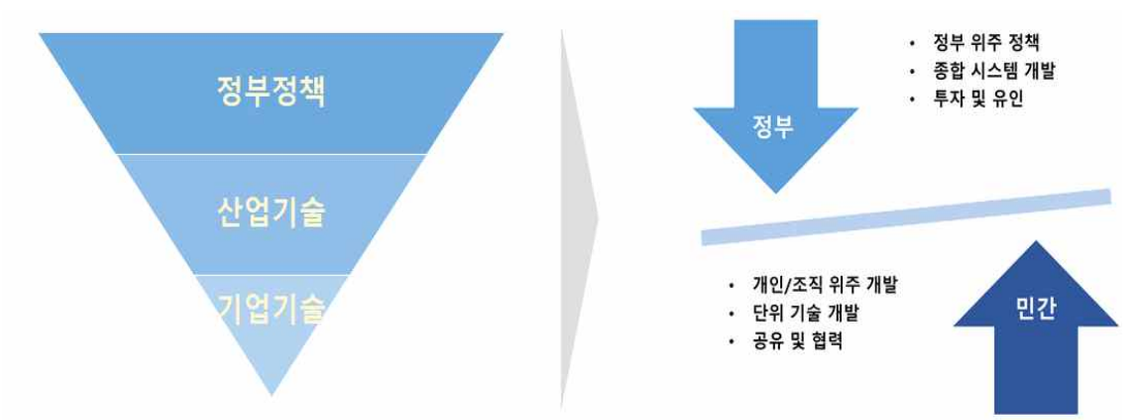
- 수많은 사람이 사용해나가면서 자체 검증이 가능해지고, 개발 및 사용에 대한 경험 공유가 강화되면서 오픈소스 소프트웨어 사용환경이 개선됨

- 최신기술습득, 개발기간단축, 개발비용절감, 중복개발예방 등 SW 개발 효율성은 물론 결과물 품질 보증, 개발자 역량 강화, 성과 공유 및 확산 등 SW 개발 효과성도 확대되고 있음(권영환, 2020)

◈ 오픈소스 소프트웨어 관련 지능화 기술 정책

- 4차 산업혁명 관련 정책 이후 인공지능 정책 및 사업 투자를 통해 국가의 데이터 과학 및 기술의 수준을 향상하기 위해 많은 노력을 기울임
  - 데이터 공개 등 국가에서 데이터 분석과 관련한 인프라를 구성하고, 산학연에 필요한 분야에 활용할 수 있도록 정부에서 주도하고 있음
- 국가 주도의 정책도 데이터 공개 및 연구를 활성화하려는 방안으로, 오픈소스 소프트웨어 발전전략과 같이 운영되어야 함
- 오픈소스 소프트웨어는 개인 또는 기업 단위의 민간 주도의 개발로 이루어지는 분야로 상향식(bottom-up) 접근으로 지능화 기술 생태계를 살펴보는 시각이 요구됨
  - 국가 주도가 아닌 개인이나 기업에서 필요로 하고 개발하고 있는 기술들을 먼저 확인하여 역할을 구분하고 기술투자계획을 세우는 등 균형적인 시각이 필요함
  - 개인 또는 기업 단위의 집단지성에 따른 지능화 기술이 개발되고 이해관계자들 간의 협력 및 경쟁 관계가 형성되는 과정과 발전 프로세스 등을 살펴봐야 함

그림 1-2 오픈소스 관련 지능화 기술 정책 방향



2 연구필요성 및 목적

가. 오픈소스의 지능화 기술 생태계 이해

◆ 오픈소스 소프트웨어 생태계 이해의 필요

- 오픈소스 소프트웨어(이하 오픈소스)는 자율적이고 자발적으로 성장해가는 자원으로, 자원이 체계적으로 자기조직화(self-organizing)되는 분야인지를 생태계 관점에서 그 특성과 과정을 이해하고 확인할 필요가 있음

- 자율적이고 자발적인 성장은 불확실성으로 인해 방향성이 불분명하고, 성장의 내용이나 방향 등을 제대로 확인하기 어렵기 때문에, 실태를 적확하게 파악하기 위한 방법을 고안하여 생태계 구성 여부 및 발전 체계의 이론을 확립하는 것이 중요함

  - 오픈소스는 결국 개발자가 기술을 개발하는 부분으로, 누가 개발하고 있는가? 무엇이 개발되고 있는가? 어떤 형태로 개발되고 있는가? 시장 및 산업 수준으로 확장되고 있는가? 등 사람과 기술을 대상으로 생태계를 분석하고 이해하는 것이 적합함

◆ 정부와 민간의 균형적인 지능화 기술 생태계 구축

- 국내 정보화 기술은 국가 주도의 기술 및 시스템 개발로 이루어져 왔으며, 과거 통신 기술의 정보화 기반뿐만 아니라 최근의 데이터 개방 및 4차 산업혁명 기술까지 정부의 로드맵 구축과 투자로 개발되어오고 있음

- 민간에서 개발되고 있는 오픈소스 위주의 지능화 기술 생태계를 확인하고 그 특성을 이해함에 따라, 정부와 민간 협력 위주의 균형적인 발전전략을 수립하고 오픈소스의 기술 생태계를 구축하여 4차 산업혁명 분야의 국제적인 기술경쟁력을 확보하여야 함

나. 기술 생태계 분석을 위한 오픈소스 데이터 처리 및 가공

◆ 오픈소스의 연구

- 오픈소스가 국제적인 소프트웨어 개발의 보고로 떠오르고, 국가 및 국내 기업에서도 관심도가 올라감에 따라 시장동향 보고서 등이 2020년 이후로 증가하고 있음

  - 한국전자통신연구원(ETRI)에서는 오픈소스 4.0이라는 패러다임 변화에 따른 오픈소스 연

구를 수행하고 있음(김성민 외, 2020)

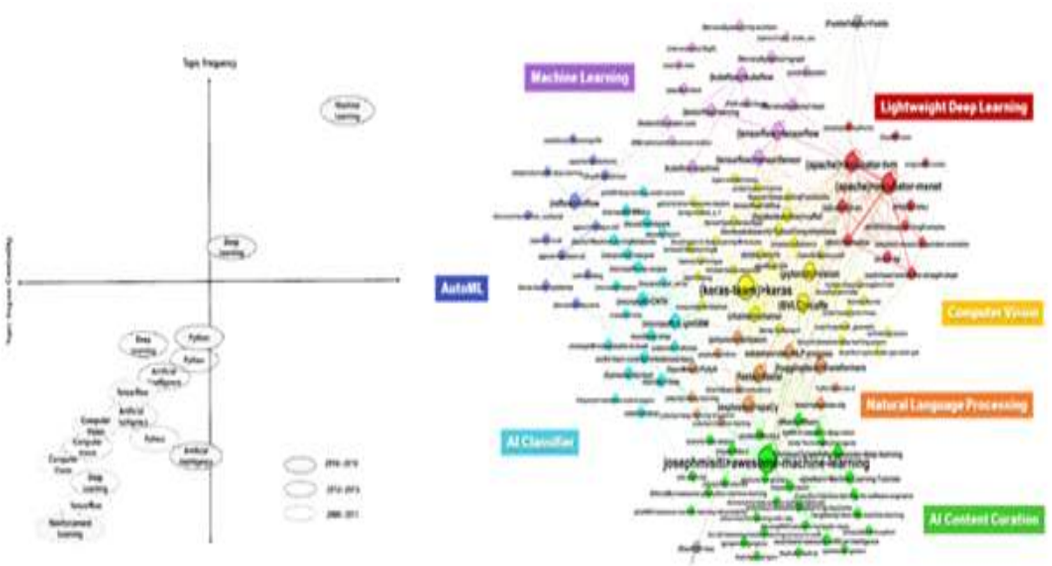
- 정보통신산업진흥원(NIPA)(2020)은 국내 기업의 오픈소스 활용 현황을 깃허브의 등록현황으로 분석하여 국내 오픈소스 소프트웨어 활용 방안을 제시함
- 소프트웨어정책연구소(SPRi)에서는 글로벌 오픈소스 생태계에 따른 해외 주요국 정책을 비교하여 국내 오픈소스 발전 정책수립의 필요성을 제시함(권영환, 2020)

- 그러나 아직까지 대량의 오픈소스에서 개발되고 있는 기술과 기업을 정량적이고 과학적인 방법을 통한 탐색적인 연구는 많이 진행되고 있지 않으며, 오픈소스의 의미나 대표적인 소수의 오픈소스만을 대상으로 보고서가 제안되고 있는 등 구체적인 오픈소스 분석이 요구됨

- 오픈소스의 사용성과 효과성이 증명되면서, 학계에서도 오픈소스 소프트웨어의 개발현황 및 발전 방향에 대해 관심을 기울이고 있음
  - 깃허브 저장소에 등록된 내용을 바탕으로, 깃허브의 주요 저장소 분석과 함께 공동 개발자 네트워크 분석 등 다양한 학술적 분석이 이루어지고 있음

- 학계 역시 오픈소스 저장소의 전체적인 탐색적 분석이나 기업 또는 기술 단위의 분석은 상대적으로 미비하여, 현재 저장소에서 작성되고 공유되는 기술내용이나 수준들을 분석하기 위한 데이터 선정이나 수집 방법, 분석 방법론은 제시되고 있지 않음

그림 1-3 오픈소스 깃허브 데이터 분석 사례



※ 이미지 출처: 정지선 외(2019), 이왕재, 이학연(2020)

◈ 오픈소스의 데이터 분석 방법론 개발 및 실행

- 대량의 오픈소스가 개발되고 발전되는 기술 생태계를 살펴보기 위해, 오픈소스를 자동으로 수집하고, 주요기술 · 주요 개발자 · 특정기술에 따른 기술 유형, 개발현황을 분석하는 빅데이터 분석 방법론을 개발함

- 기술과 개발자 관점에서 현재 개발되고 있는 오픈소스 현황 분석을 통해, 기술이 개발되고 발전하는 프로세스를 모형화하고, 기업 특성을 반영하여 오픈소스의 기술유형은 차이가 있는지, 그리고 특정 기술개발을 위해 오픈소스가 생태계적 측면에서 활용가치가 있는지를 분석함

- 이를 위해 오픈소스 데이터를 수집 및 확보하고, 속성과 내용을 체계적으로 분석하는 방법을 개발하고 제안함

  – 오픈소스를 관리하는 깃허브를 중심으로, 깃허브에 등록된 상위 인지도 기준의 주요 기술을 분석하고, 빅테크 기업 위주의 주요 개발자의 오픈소스 개발 현황, 그리고 특정 기술 단위에서 오픈소스가 개발되고 있는 현황을 분석함

Ⅱ 연구방법

1 데이터 수집

가. 오픈소스 저장소 개요

◈ 오픈소스 저장소의 기능

- 오픈소스를 한 장소에 모아 개발, 검색, 사용, 공유 등을 효과적으로 관리하기 위한 저장소로 협력 및 공유, 경쟁의 장으로 자리매김하고 있음
  - 오픈소스 저장소는 개방성, 투명성, 용이성, 분산과 통제의 균형, 네트워크 외부성 등의 특징으로 생태계를 구성하고 확장하는데 기여함(김성민 외, 2020)
  - 오픈소스 저장소에는 소스코드, 설명 등의 소프트웨어를 업로드하여 공유 및 협력개발하며, 참여현황에 따라 인지도나 수정내용 등을 확인하는 기능을 제공함
- 대표적인 오픈소스 호스팅 서비스로 깃허브(github)가 있으며, 사용자 및 활용 저장소들은 매년 늘어가고 있음
  - 2018년 MS가 75억달러에 인수하여 오픈소스 개발환경을 제공해주고 있음

나. 깃허브 개요

◈ 깃허브 현황

- 깃허브는 전 세계 SW 개발자들은 물론, 빅테크 기업들과 유수의 대학 연구소들에서도 소스코드를 공개하고 협업하여 발전시키는 활동을 지속적으로 진행하고 있음
  - 구글, 마이크로소프트(MS), 페이스북, 애플, 아마존 등이 주도하여 깃허브 등 저장소와 프로젝트를 운영하고 있으며, 전 세계 개발자들이 각 빅테크 기업의 소스코드 사용을 공유하면서 플랫폼화되어가고 있음
  - 대학과 연구소에서도 연구개발 및 논문 작성 시 검증된 알고리즘 활용을 위해 오픈소스를 기반(base)으로 하여 확장하고 개선하는 등 소프트웨어 영역에서 오픈소스는 개발, 검증, 개선을 자발적으로 하며 진화하는 새로운 생태계를 구축하고 있음
- 산학연에 포함된 소프트웨어 개발자들의 활발한 활동으로 깃허브의 저장소

는 폭발적으로 증가하고 있음

- 깃허브의 저장소는 2013년 약 1,000만 개에서부터 2020년 약 1.9억 개, 2021년에는 약 2.4억 개로 증가하고 있음[2)]
- 개발자는 2017년 2,400만 명에서 2021년 약 5,600만 명[3)]으로 규모가 두 배 증가하는 등 소프트웨어 기술 발전과 함께 활용도도 점차 올라가고 있음
- 포춘 50대 기업의 약 72%가 깃허브를 사용[4)]하고 있을 정도로, 기업에서도 민간과의 공유 및 협력을 중시하며, 집단지성을 이용한 개방형 체제의 혁신을 이용함
- 빅테크 기업들은 기업소유 또는 프로젝트 팀 위주로 계정을 만들어, 코드의 기반(infrastructure)을 공개하여, 이를 이용해 발전할 수 있는 시작점을 제공하는 등 생태계의 가장 기본을 이루고 있음

- 오픈소스를 활용하는 깃허브 플랫폼 역시 공개(public)와 비공개(private)로 나누어져, 실제 기업의 활용영역 차원의 코드를 확인하기는 어려움
  - 공개 데이터는 기초 코드의 연산 및 개발, 검증을 위한 개발자들의 협업이 방향이라면, 사설 데이터는 코드를 이용해 기업이 비즈니스에 활용하는 영역으로 실제 사업 전까지는 비공개로 운영하는 경우가 많음

다. 깃허브 제공 데이터

◈ 깃허브 시스템 및 저장소

- 깃허브는 소프트웨어 개발자들이 작성한 코드 또는 시스템을 웹 서비스를 이용하여 자유롭게 공유하고, 가공·수정해나가며 개선하는 개방형 시스템
- 깃허브에 올라온 관련 저장소(repository)가 하나의 목적을 가지고 개발되고 공유되고 있는 코드로서, 오픈소스 소프트웨어라고 지칭할 수 있음
- 기본적으로 개발자(owner)가 상위 수준이며, 개발자가 작성한 코드들을 목적에 맞게 저장소들로 만들고 있음
  - 저장소를 프로젝트로 언급하기도 하나, 엄밀히 말하면 저장소는 개인 프로젝트 성향으로 수정이나 개선 등을 자유롭게 할 수 있다면, 프로젝트는 협업(collaboration)하는 저장소로 개인적으로 수정이나 개선 등을 하기보다는 검증된 코드로 공유하고 확정하는 저장소로 구별될 수 있음
  - 깃허브의 저장소 주소는 개발자/저장소(html 주소상으로는 {owner}/{repo})의 형태로 제공됨

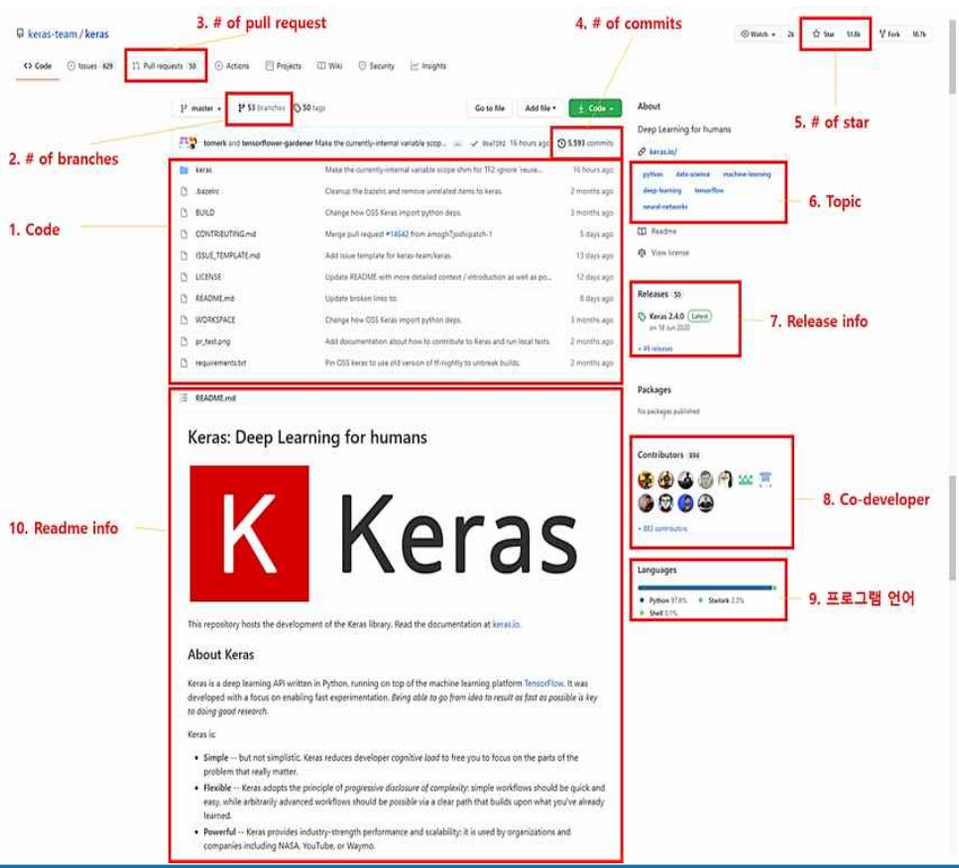
2) ETRI Insight 자료, Github Octoverse 2021 자료
3) ETRI Insight 자료, Github Octoverse 2021 자료
4) Github Octoverse 2021 자료

- 개발자가 kears-team, 저장소가 keras라면, 깃허브 주소는 github/keras-team/keras로 설정됨

◆ 깃허브 오픈소스 저장소 구성

- 깃허브의 주요 활동은 오픈소스 저장소에서 이루어지며, 저장소의 구성은 코드에서부터, 그 코드에 대한 설명, 그리고 저장소의 코드를 확인하고, 사용하고, 공유되는 과정을 나타내는 정보들로 이루어짐
- 깃허브의 정보들은 웹 페이지와 API로부터 확인할 수 있음
- 우선 깃허브의 웹 페이지에서는 코드, 브랜치, 풀 리퀘스트, 커밋, 스타, 토픽, 릴리즈, 기여자, 언어, 소개(readme) 등이 주요 내용이며, 그 외에도 조회(watch), 복제(folk), 이슈 등 저장소에 대한 기본 활용정보를 깃허브 저장소 홈페이지에서 보여주고 있음

그림 2-1 깃허브 저장소의 웹페이지 주요 구성



※ 이미지 출처: 깃허브의 keras-team/keras 홈페이지

그림 2-2 깃허브 저장소의 API의 주요 구성

2 분석방법론

가. 깃허브 오픈소스 저장소 데이터 특성

◈ 기술속성 데이터

- 깃허브의 저장소는 웹 페이지를 접속하여, 필요로 하는 정보검색을 통해 해당 저장소에 대한 정보를 확인할 수 있으며, 자체적으로 최근 개발이나 공유현황 등을 나타내는 활동을 하고 있음
  - 웹 페이지에서 저장소 각각에 대한 정보 확인이 가능하나, 기술 생태계 분석이라는 측면에서 저장소들의 통계, 저장소 간의 관계성, 저장소의 기술 내용 등 종합적인 분석 체계는 제공하지 않고 있음
- 깃허브는 대용량의 오픈소스 정보를 효과적으로 확인하기 위한 도구로 API를 제공하고 있으며, 개발자, 개발환경, 현황 등 여러 기술속성을 검색할 수 있도록 지원하고 있음
  - 이는 깃허브 홈페이지에서 제공되는 화면과는 별개로 관련 항목(key)에 대한 정보를 쿼리(query) 형태로 웹 페이지에서 보여주고 있음
  - 웹 브라우저를 사용할 경우, html의 텍스트 형태로 검색 정보에 해당하는 저장소의 정보를 보여주나, 전용도구(POSTMAN 등)를 사용할 경우, 저장소 정보를 가공하기 쉬운 형태의 언어나 포맷(json, xml 등) 형태로도 저장할 수 있음
  - 기본적으로 웹 페이지를 통해 확인할 수 있는 사항은 API에서도 활용할 수 있으며, 웹페이지에서 확인하기 어려운 개발자의 유형(type: user(개인) 또는 organization(기업) 여부), 시작일자(created) 및 업데이트(updated) 일자 정보도 확인 가능함
  - 저장소에 라이센스(license) 사용 여부 등도 있어, 구축 기반을 확인할 수 있음(대부분 GPL, MIT, Apache 등 허용적 라이센스 활용)
  - 그러나 오픈소스의 특징상 정보가 없는 경우가 많으며, 일관적이지 않고, 저장소마다 정보의 양과 질이 다르다는 점에 유의해야 함
- 기술속성을 나타내는 항목 및 값을 바탕으로 상위 인지도, 공유 정도 등 기술현황을 파악할 수 있으며, 분석 대상을 결정하기 위해 필요한 주요 기본 정보로 활용할 수 있음
  - 본 연구에서는 개발기업과 기술내용의 대상을 결정하기 위해서, 기술속성 중 스타 수에 따라 상위 인지도로 먼저 주요 저장소를 선정하여 분석을 수행하는 등 분석 기준을 결정하기 위해 속성이 사용될 수 있음

표 2-1 깃허브 저장소의 기술속성의 데이터 항목

주요 항목	항목 내용
1. 코드	• 해당 저장소에 대한 코드, 패키지 등 사용 방법
2. 브랜치(branch)	• 현재 저장소를 통해 파생된 프로젝트
3. 풀 리퀘스트(pull request)	• 코드 병합에 대한 요구
4. 커밋(commit)	• 해당 저장소의 코드에 대한 수정 요구
5. 스타(star)	• 개발자들의 북마크 수, 즉, 인지도 또는 인기도
6. 토픽	• 해당 저장소의 코드 및 기능의 주제
7. 릴리즈(release)	• 개발시기에 대한 정보, 버전(version)
8. 기여자	• 공동개발자
9. 언어	• 작성 프로그래밍 언어(e.g. javascript, C, python 등)
10. 소개(Readme)	• 해당 저장소에 대한 자유로운 소개

◈ 기술내용 데이터

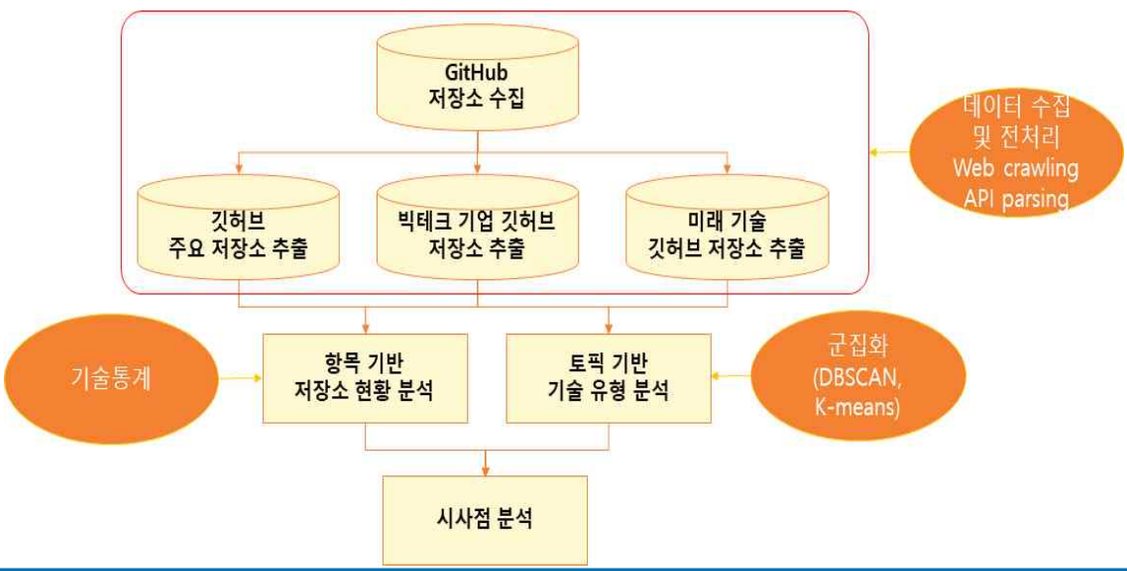
- 깃허브에서 제공하는 데이터에서 기술내용을 파악하고, 전체적인 오픈소스 기술개발 현황을 파악할 수 있는 데이터의 가용성을 파악함
- 깃허브나 관련 분석 웹 사이트 등에서 현재 깃허브에서 개발되고 있는 저장소의 전체적인 기술내용 특성이나 유사성, 그리고 개발규모 등은 파악되고 있지 않다는 점에서, 이를 전체적으로 탐색할 수 있는 데이터 수집 및 가공이 요구됨
- 해당 저장소를 소개하는 tag 개념인 토픽을 수집하여, 유사 토픽을 가지고 개발되고 있는 저장소들의 현황 분석 가능성을 확인함
  - 저장소의 내용을 자유롭게 소개하는 Readme 자료가 포함 및 제공되고 있지만, 오픈소스의 특성으로 인해 그 양이나 질이 일정하지 않아 사용이 어려움
  - 개발자가 가장 중요하고 적절하다고 설명을 하는 토픽 tag 정보를 바탕으로, 깃허브 저장소의 전체적인 토픽 현황을 탐색하는 방향으로 생태계 분석을 수행함

나. 분석절차 및 방법

◈ 분석절차

○ 깃허브의 저장소 분석을 통한 기술현황을 살펴보기 위해, 데이터 수집 및 전처리, 기술통계 및 유형 분석, 시사점 도출의 순서로 진행함

그림 2-3 분석절차



◈ 데이터 수집 및 전처리

○ 앞서 살펴본 웹페이지 데이터와 API 데이터를 수집하는 방법을 개발함

- (수집 데이터) 총 세 가지 데이터베이스를 구축을 목표로 하며, 1) 주요 저장소를 분석하기 위해 상위 인지도를 나타내는 스타 수 속성으로 저장소를 수집하고, 2) 기업의 저장소 개발현황을 살펴보기 위해 기업이 소유자인 저장소를 추출하며, 3) 검색기술의 개발현황을 살펴보기 위해 기술토픽에 해당하는 저장소를 출력함
- (수집 방법) 깃허브 웹페이지는 인터페이스 기반의 페이지 변화와 자동 크롤링을 방지하기 위해 요청시간을 확인하고 있어, 동적 크롤링 기반의 셀레니움(selenium) 패키지를 사용하여 웹페이지의 내용을 크롤링하여 수집함
- (데이터 전처리) 크롤링한 데이터와 API 기반의 json 파일은 태그 또는 key와 “” 등으로 구분된 구분자를 바탕으로 beautifulsoup 패키지를 사용해 파싱하여, 항목별로 데이터베이스화함

◈ 항목 기반 기술통계 분석

- 주요 항목의 빈도 분석, 그래프 분석 등 현황파악을 위한 단순 기술통계분석을 수행함

◈ 토픽 기반 기술유형 분석

- 다수의 기업 또는 기술 저장소의 기술내용을 확인하기 위해 토픽 기반의 기술유형화를 수행함
- 군집화 알고리즘은 k-means 클러스터링과 DBSCAN(Density-Based Spatial Clustering of Applications with Noise) 클러스터링을 사용하여 저장소를 유형화함
  - K-means 클러스터링은 사용자가 임의로 클러스터 개수를 정하여 유연하게 분석을 할 수 있으나 객관적인 클러스터 개수 지정에 어려운 단점이 있다면, DBSCAN은 클러스터 개수를 수치적 지표 기준으로 자동으로 설정하나 전체적인 기술내용을 클러스터 개수를 조절해가며 살펴보기 어려운 단점이 있음
- K-means 클러스터링은 거리 기반 클러스터링 기법으로 유사한 거리에 있는 데이터들끼리 하나의 군집으로 생성하는 분할 알고리즘임(김재원, 신광섭, 2020)
  - (기본 원리) 거리 기반 클러스터링으로 데이터 간의 거리 계산이 원리이며, 일반적으로 반경(radial)을 기준으로 군집을 결정하는 방식
  - (클러스터 개수) k개의 클러스터를 사용자가 임의로 변화하며 클러스터링이 수행 가능한 장점이 있는 동시에, 적정 k를 찾기 어렵다는 단점이 있음
  - (수행 방식) ① 기하공간 상에 k개의 중심점(centroid)를 무작위(random)로 선정

② 이 중심점들과 모든 데이터들의 거리(일반적으로 유클리디안 거리)를 계산하여, 데이터들을 가장 가까운 중심점의 군집으로 배정

③ 군집으로 배정된 데이터들의 중심점을 다시 계산하고, 중심점들과 모든 데이터들의 거리를 다시 계산하고, 다시 가장 가까운 중심점의 군집으로 배정

④ ③을 반복해가며 더 이상 데이터의 군집이 바뀌지 않거나 정체될 때 알고리즘을 정지하고 군집을 정리

  - (파이썬 코드) scikit-learn 라이브러리의 KMeans함수를 사용하여 수행
- DBSCAN은 밀도 기반 클러스터링 기법으로 거리가 아닌 데이터들의 밀집도(공간 대비 데이터 분포)에 따라 데이터를 군집화하는 방법
  - (기본 원리) 공간 크기(Eps)를 결정하고, 그 공간이 클러스터로 인정받기 위한 데이

터들의 최소 개수(minPts)를 설정하여 군집을 형성하는 방식

- (클러스터 개수) 공간 크기와 데이터들의 최소 개수에 따라 군집의 크기가 달라지나, k-dist 지표를 사용하여 최적 공간크기를 설정할 수 있으며, 공간 크기와 최소개수를 설정하면, 그에 따른 클러스터 개수는 이를 만족하는 수준에서 자동으로 설정됨
- (핵심 정의) ① 이웃점(neighborhood of a point) : 한 데이터로부터 반경 내에 존재하는 다른 데이터를 이웃점이라 정의함

② 핵심점(core point) : 최소의 데이터 포인트인 minPts개 이상의 이웃점을 갖는 데이터 포인트

③ 군집(cluster) : 핵심점과 이웃점은 모두 같은 군집으로 묶이며, 만약 이웃점이 또 다른 데이터들의 핵심점이라면 모두 다 같은 군집으로 엮어짐

④ 경계점(border point) : 핵심점은 아니지만, 군집에 속해있는 경우

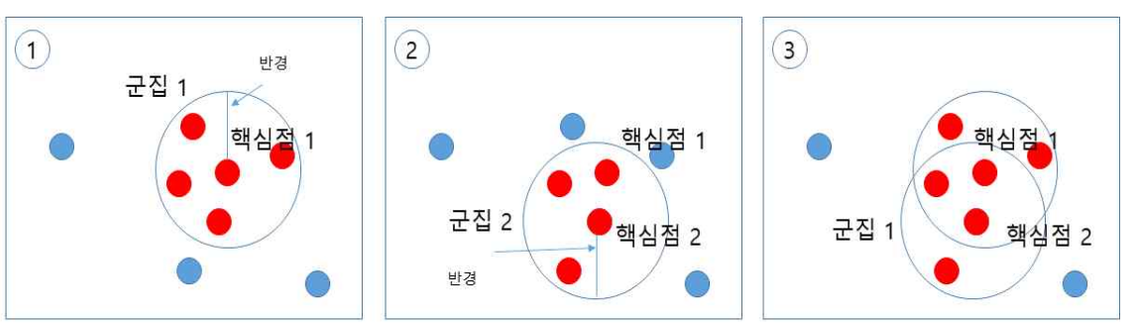
⑤ 잡음점(Noise point) : 핵심점도 경계점도 아닌 데이터

- (수행 방식) ① 핵심점에 해당하는 데이터를 모두 찾아내고, 핵심점과 이웃점을 군집으로 설정(아래 그림에서 핵심점 1과 핵심점 2를 찾아냄)

② 핵심점의 이웃점이 다른 반경의 핵심점인 경우 같은 군집으로 설정(아래 그림에서 핵심점 1과 핵심점 2는 서로의 이웃점으로서 같은 군집으로 통합)

- (파이썬 코드) scikit-learn 라이브러리의 DBSCAN 함수를 사용하여 수행

그림 2-4 DBSCAN 수행방식



MinPts=4

Ⅲ 연구결과

1 깃허브 주요 저장소 분석

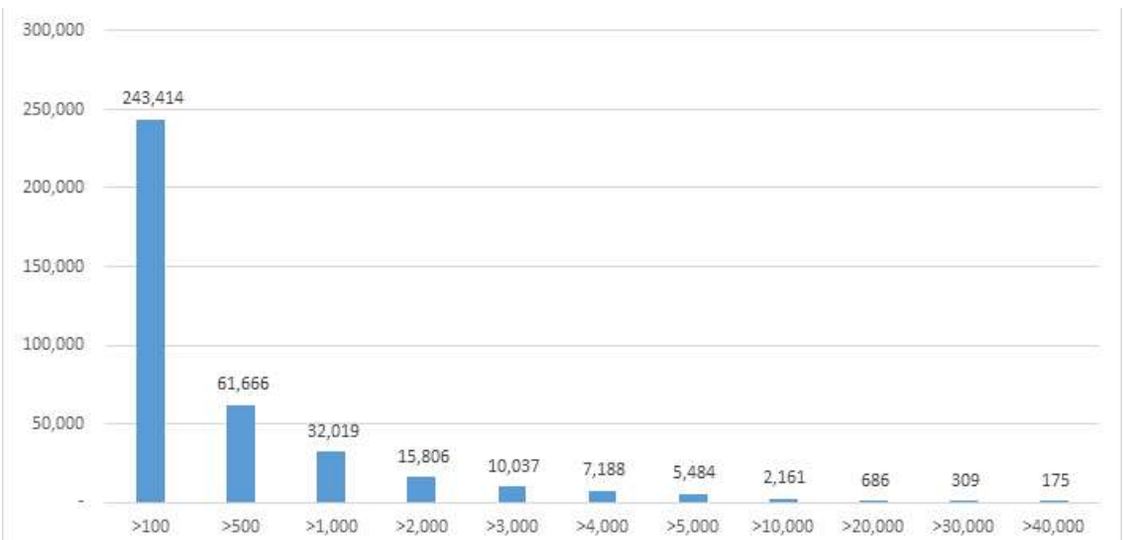
가. 상위 인지도 저장소 데이터 수집

◆ 인지도(star) 기반 저장소 수 현황

○ 깃허브의 저장소들은 조회와 스타(star, 북마크)를 통해 대중적 인지도를 확인할 수 있으며, 특히 북마크는 의도를 가지고 있는 인지도라는 점에서 깃허브 내의 전체에서 의미있는 저장소를 확인하기에 용이함

- 깃허브에서는 전수의 저장소를 검색할 수가 없어, 깃허브에서 개발되고 있는 주요 저장소를 확인하기 위해서는 인지도에 기반한 방법이 가장 용이함
- 스타 수가 100개 이상인 저장소의 경우 243,414개이나, 1,000개 이상은 32,019개, 10,000개 이상은 2,161개로 급격히 감소함[5]을 볼 수 있음
- 이는 전체 저장소가 약 2.4억 개에 달하지만, 실제 주로 공유되고 개발·활용되는 저장소는 스타 수 100개 이상으로 봤을 때 0.1% 정도에 머무르고 있음(스타 수 10,000개 이상 기준으로 할 경우 0.001%)

그림 3-1 인지도(star)에 기반한 저장소 수



※ 2021.10.25. 기준

5) 2021.10.25. 기준으로 “stars:” 쿼리로 검색

◈ 상위 인지도 저장소

- 스타 수를 기준으로 상위 10개의 저장소를 분석하면, 대부분 교육용, 자료 모음이며, 웹 개발, 기계학습 용도의 저장소도 나타남
  - 교육이나 자료는 오픈소스의 주요 목적으로, 사용자가 무료로 소프트웨어 기술과 내용을 교육받고 사용할 수 있도록 운영하고 있음
  - 웹 개발 기술의 Front-end, Back-end 등 모든 웹 플랫폼 또는 어플리케이션 개발 주기 상의 프레임워크 또는 패키지를 개발하고 있음
  - 기계학습에 특화된 프레임워크나 패키지도 높은 인지도를 지님

표 3-1 상위 인지도 저장소와 개발 내용

저장소	스타 수	내용
freeCodeCamp/freeCodeCamp	334,000	오픈소스 교육을 위한 콘텐츠 제공
EbookFoundation/free-programming-books	209,000	프로그래밍 언어 E-book 개발 저장소
jwasham/coding-interview-university	196,000	소프트웨어 프로그래밍 교육 주제 제공
vuejs/vue	190,000	Vue 관련 프레임워크 개발
facebook/react	177,000	Facebook의 자바스크립 기반 react 프레임워크 개발
kamranahmedse/developer-roadmap	175,000	웹 개발(frontend, backend, devops)과 관련한 기술 개발
sindresorhus/awesome	173,000	웹 개발 관련 주제별 모음 저장소
public-apis/public-apis	165,000	소프트웨어 및 웹 개발용 무료 API 저장소
tensorflow/tensorflow	160,000	기계학습 프레임워크 오픈소스(구글 프로젝트)
twbs/bootstrap	154,000	Frontend 기술 개발 저장소

나. 상위 인지도 저장소 주제 분석

◈ 상위 인지도 저장소 토픽 탐색

- 깃허브의 저장소들은 개발자가 관련 주제를 토픽이라는 정보로 해쉬태그처럼 이용하고 있으며, 이 토픽은 앞서 말한 대로 개발자가 저장소를 설명할 수 있는 용어로 정의했다고 볼 수 있음

- 스타 수가 높은 저장소의 토픽들을 데이터 input으로 하여 군집화를 시도하고, 각 군집의 특성에 따라 현재 깃허브에서 주로 개발되고 있는 기술 추이를 살펴보는 것이 필요함

◈ 상위 인지도 저장소 토픽 군집화

- 상위 인지도 저장소 총 500개(스타 수 10,000건 이상 중 토픽을 기록한 상위 500개 저장소)를 대상으로 하여, 현재 저장소 토픽 현황을 살펴봄

- 저장소-토픽 매트릭스를 구축하고, 클러스터링 알고리즘을 사용하여 총 23개로 군집화함

  - 1개씩 클러스터링 된 이상치(outlier) 17개를 제거한 총 483개가 23개로 군집되었으며, 클러스터링 알고리즘은 클러스터 개수를 자동으로 결정하는 밀도 기반의 DBSCAN 알고리즘을 사용함

- 깃허브의 저장소는 대부분 소프트웨어 개발자들이 주도하고 있으며, 특히 공개 저장소에서는 소프트웨어 연구개발 및 검증이 강해, 활용영역보다는 기술개발 영역 위주의 토픽들로 구분되어 있음을 확인함

  - 특히, 소프트웨어 분야에 웹 기술과 기계학습 기술이 중요해지면서, 저장소에 가장 많은 토픽을 차지하고 있음

  - 웹 기술은 관리자가 다루는 데이터베이스부터 서버 관리, 네트워크 연결에서부터 사용자가 요청한 데이터를 직접 화면을 보고 인터페이스로 운영하는 모든 기술에 대해 저장소에서 주로 다뤄지고 있음

  - 이와 같은 웹 기술은 사용자 측면의 Front-end 기술, 관리자 측면의 Back-end 기술로 구분되며, 군집 1, 3, 6, 8, 11, 12, 17, 22 등과 같이 세부기술 주도로 저장소가 운영되고 있음을 확인함

※ 가장 많은 저장소가 군집화된 1번 클러스터를 DBSCAN을 이용해 재군집화해보아도 Front-end와 Back-end 웹 기술 플랫폼 구분이 주류를 이루고 있음

  - 기계학습은 군집 2에 딥러닝, 파이썬, NLP, 텐서플로우, 파이토치, scikit-learn 등 데이터(이미지, 자연어 등), 알고리즘(딥러닝), 언어(파이썬), 패키지(텐서플로우, 파이토치,

scikit-learn 등)의 토픽을 가지고 분류되어 있음

- 그 외에 OS 기반(리눅스, 애플 등), 앱 기반(위챗, 페이스북 등), 기술 기반(게임, IoT, 동영상) 등으로도 인지도가 높은 저장소들이 확인됨

- 깃허브 저장소들에서 개발되고 공유되는 프로그램 또는 코드 등의 활용영역 측면에서는 확인하기 어려움
  - 이는 앞서 언급한 대로 개발자 위주의 오픈소스 생태계인 이유가 있으며, 활용과 관련해서는 비공개 저장소를 이용하는 등 깃허브 자체적으로 분석하기가 어려움
  - 활용이나 사업 측면의 오픈소스 플랫폼을 새롭게 만든다거나 정리하지 않는 한, 기업이나 기술 키워드로 검색하여 현재 깃허브 내의 개발현황을 살펴봐야 함

표 3-2 깃허브 상위 인지도 500개 저장소 군집화 결과

군집	수	내용	주요 토픽(괄호 안의 숫자는 토픽 출현 횟수)
1	213	Front-end/ Back-end 통합기술	('javascript', 71), ('interview', 54), ('awesome', 43), ('python', 43), ('sql', 40), ('web', 39), ('css', 38), ('nodejs', 35), ('data', 34), ('git', 30), ('algorithm', 29), ('html', 27), ('go', 25), ('android', 25), ('hacktoberfest', 21)
2	47	기계학습	('machinelearning', 45), ('deep-learning', 40), ('neural', 21), ('python', 20), ('nlp', 17), ('tensorflow', 14), ('search', 13), ('face-swap', 13), ('data', 9), ('pytorch', 9), ('distributed', 5), ('scikit-learn', 5), ('note', 5), ('java', 4), ('tutorial', 3)
3	34	Front-end 중 사용자 인터페이스	('react', 47), ('vue', 42), ('ui', 20), ('javascript', 16), ('material', 10), ('web', 9), ('typescript', 6), ('awesome', 6), ('hacktoberfest', 5), ('flutter', 5), ('system', 4), ('admin', 4), ('components', 4), ('mobile', 4), ('css', 3)
4	31	위챗 기반 기술	('api', 33), ('wechat', 14), ('security', 10), ('rest', 8), ('cloud', 7), ('python', 6), ('swagger', 6), ('javascript', 5), ('web', 4), ('hacktoberfest', 3), ('nodejs', 3), ('react', 3), ('vue', 3), ('style', 3), ('macos', 3)
5	17	리눅스 기반 기술	('vim', 14), ('editor', 8), ('typescript', 4), ('text-editor', 4), ('wysiwyg', 4), ('javascript', 4), ('browser', 3), ('cross-platform', 3), ('c', 2), ('neovim', 2), ('nvim', 2), ('rich-text-editor', 2), ('vscode', 2), ('monaco-editor', 2), ('spacemacs', 2)
6	14	Back-end 기술(spring, redis)	('spring', 54), ('java', 6), ('distributed', 5), ('mybatis', 4), ('microservice', 4), ('alibaba', 4), ('docker', 3), ('redis', 3), ('swagger', 3), ('admin', 3), ('vue', 3), ('shiro', 3), ('dubbo', 3), ('framework', 2), ('mongodb', 2)
7	11	게임 기술	('game', 17), ('drag', 12), ('javascript', 4), ('sort', 3), ('ui', 2), ('html', 2), ('react', 2), ('reordering', 2), ('component', 2), ('open-source', 1), ('multi-platform', 1), ('godotengine', 1), ('godot', 1), ('gui', 1), ('tools', 1)
8	9	Back-end 기술(php)	('php', 22), ('framework', 3), ('laravel', 2), ('hacktoberfest', 2), ('awesome', 2), ('design', 2), ('symfony', 1), ('bundle', 1), ('symfony-bundle', 1), ('oop', 1), ('code-examples', 1), ('test', 1), ('hack', 1), ('hhvm', 1), ('hacklang', 1)
9	9	리눅스 기반 기술	('docker', 27), ('containers', 2), ('orchestration', 2), ('awesome', 2), ('moby', 2), ('go', 1), ('cli', 1), ('inspector', 1), ('tui', 1), ('explorer', 1), ('social-network', 1), ('activity-stream', 1), ('microblog', 1), ('mastodon', 1), ('web', 1)

10	9	분류불능	('hacktoberfest', 8), ('programming', 2), ('apollo', 1), ('nasa', 1), ('agc', 1), ('applications', 1), ('coding', 1), ('ideas', 1), ('links', 1), ('books', 1), ('cs', 1), ('sites', 1), ('dotnet', 1), ('aspnetcore', 1), ('productivity', 1)
11	9	Front-end 기술(redux)	('redux', 9), ('react', 5), ('redux-saga', 3), ('web', 2), ('javascript', 2), ('i18n', 1), ('style', 1), ('offline-first', 1), ('scaffolding', 1), ('immer', 1), ('middleware', 1), ('sagas', 1), ('effects', 1), ('immutable', 1), ('reducer', 1)
12	9	애플 플랫폼 (swift)	('swift', 30), ('ios', 11), ('awesome', 5), ('react', 5), ('detection', 3), ('xcode', 3), ('cocoapods', 3), ('carthage', 3), ('apple', 3), ('json', 3), ('animation', 3), ('request', 2), ('response', 2), ('list', 2), ('server', 2)
13	9	IoT 기술	('automation', 11), ('iot', 4), ('python', 2), ('mqtt', 2), ('twitter', 2), ('monitoring', 2), ('zsh', 2), ('raspberry-pi', 1), ('internet-of-things', 1), ('asyncio', 1), ('hacktoberfest', 1), ('notifications', 1), ('agent', 1), ('rss', 1), ('scraper', 1)
14	8	분류불능	('open-source', 14), ('humans', 3), ('data', 1), ('aaron-swartz', 1), ('awesome', 1), ('android', 1), ('react', 1), ('javascript', 1), ('web', 1), ('document', 1), ('hacktoberfest', 1), ('ssh', 1), ('ansi', 1), ('stunnel', 1), ('tor', 1)
15	8	테스트 관련 저장소	('test', 31), ('nodejs', 4), ('javascript', 3), ('security', 2), ('hacking', 2), ('hacktoberfest', 2), ('mocha', 2), ('tdd', 2), ('react', 2), ('ava', 2), ('android', 1), ('awesome', 1), ('reverse-engineering', 1), ('bug-bounty', 1), ('fuzzing', 1)
16	8	분류불능	('language', 9), ('programming', 3), ('compiler', 2), ('javascript', 1), ('typechecker', 1), ('typescript', 1), ('rust', 1), ('science', 1), ('machinelearning', 1), ('hpc', 1), ('julia', 1), ('scientific', 1), ('numerical', 1), ('v', 1), ('syntax-highlighting', 1)
17	7	Front-end 기술(CSS)	('material', 18), ('javascript', 3), ('css', 3), ('framework', 3), ('angular', 3), ('bootstrap', 3), ('android', 2), ('ios', 2), ('web', 1), ('sprites', 1), ('icons', 1), ('design', 1), ('mdl', 1), ('html', 1), ('uikit', 1)
18	6	분류불능	('package', 8), ('npm', 2), ('dependency-manager', 2), ('python', 2), ('javascript', 1), ('yarn', 1), ('publishing', 1), ('lerna', 1), ('monorepo', 1), ('ruby', 1), ('macos', 1), ('homebrew', 1), ('brew', 1), ('php', 1), ('composer', 1)
19	6	C++ 기반 기술	('c-plus-plus', 6), ('cpp', 4), ('crypto', 2), ('c', 2), ('awesome', 2), ('bitcoin', 1), ('p2p', 1), ('lists', 1), ('list', 1), ('libraries', 1), ('resources', 1), ('programming', 1), ('ffmpeg', 1), ('live-streaming', 1), ('vedio', 1)
20	6	동영상 기술	('vedio', 15), ('html', 3), ('player', 3), ('android', 2), ('ffmpeg', 2), ('ijkplayer', 2), ('python', 1), ('animation', 1), ('javascript', 1), ('flash', 1), ('hls', 1), ('dash', 1), ('ios', 1), ('macos', 1), ('swift', 1)
21	5	분류불능	('kubernetes', 4), ('cncf', 4), ('go', 2), ('containers', 2), ('cluster', 1), ('mini', 1), ('chart', 1), ('charts', 1), ('helm', 1), ('cats', 1), ('corgis', 1), ('cars', 1), ('rocket-ships', 1), ('more-cats', 1), ('cats-over-dogs', 1)
22	4	페이스북 Back-end 기술(데이터)	('graphql', 7), ('rest', 4), ('server', 4), ('sql', 3), ('postgres', 2), ('automation', 2), ('api', 1), ('bigquery', 1), ('access-control', 1), ('hasura', 1), ('notifications', 1), ('nodejs', 1), ('relay', 1), ('mongodb', 1), ('backend', 1)
23	4	모바일 앱 기술	('mobile', 4), ('android', 3), ('ios', 3), ('web', 3), ('touch', 2), ('windows', 1), ('macos', 1), ('dart', 1), ('material', 1), ('desktop', 1), ('app-framework', 1), ('skia', 1), ('linux', 1), ('fuchsia', 1), ('dart-platform', 1)

표 3-3 1번 클러스터(Front-end, Back-end 통합기술) 재군집화 결과

군집	수	내용	주요 토픽(괄호 안의 숫자는 토픽 출현 횟수)
1-1	31	Front-end 기술(CSS, html)	('css', 29), ('html', 11), ('javascript', 8), ('frontend', 7), ('interview', 3), ('sass', 2), ('scss', 2), ('style', 2), ('animation', 2), ('web', 2), ('test', 2), ('awesome', 2), ('bootstrap', 1), ('lists', 1), ('checklist', 1)
1-2	28	Back-end 데이터베이스 기술(SQL, 데이터)	('sql', 30), ('data', 12), ('vision', 5), ('business', 5), ('javascript', 4), ('bi', 3), ('dashboard', 3), ('analytics', 3), ('distributed', 2), ('slack', 2), ('postgres', 2), ('clojure', 2), ('reporting', 2), ('metabase', 2), ('orm', 2)
1-3	25	프로그래밍 언어(GO)	('git', 21), ('go', 4), ('golang', 4), ('awesome', 3), ('gogs', 3), ('devops', 3), ('hacktoberfest', 3), ('python', 1), ('lsif-enabled', 1), ('list', 1), ('cli', 1), ('homebrew', 1), ('pull-request', 1), ('tips', 1), ('tips-and-tricks', 1)
1-4	25	안드로이드 패키지	('android', 20), ('kotlin', 8), ('java', 5), ('animation', 2), ('c', 1), ('ffmpeg', 1), ('sdl2', 1), ('screen', 1), ('libav', 1), ('recording', 1), ('mirroring', 1), ('graalvm', 1), ('samples', 1), ('programming', 1), ('compiler', 1)
1-5	23	파이썬	('python', 19), ('queue', 4), ('task', 3), ('awesome', 2), ('terminal', 2), ('collections', 1), ('emoji', 1), ('syntax-highlighting', 1), ('markdown', 1), ('progress', 1), ('traceback', 1), ('ansi', 1), ('rich', 1), ('hacktoberfest', 1), ('tables', 1)
1-6	20	Back-end 서버 기술	('nodejs', 20), ('javascript', 9), ('version', 4), ('hacktoberfest', 3), ('koa', 3), ('windows', 2), ('nvm', 2), ('posix', 2), ('framework', 2), ('typescript', 2), ('macos', 1), ('linux', 1), ('mit', 1), ('runtime', 1), ('express', 1)]
1-7	19	저장소 모음	('awesome', 18), ('list', 5), ('nodejs', 2), ('lists', 1), ('unicorns', 1), ('resources', 1), ('cloud', 1), ('privacy', 1), ('selfhosted', 1), ('hosting', 1), ('self-hosted', 1), ('javascript', 1), ('computer-science', 1), ('courses', 1), ('beginner-project', 1)
1-8	17	웹 자바 기술	('web', 19), ('javascript', 5), ('nodejs', 3), ('html', 2), ('audio', 2), ('angular', 1), ('typescript', 1), ('pwa', 1), ('svg', 1), ('canvas', 1), ('augmented-reality', 1), ('virtual-reality', 1), ('3d', 1), ('nwjs', 1), ('desktop', 1)
1-9	15	자바스크립트 기술	('javascript', 11), ('maps', 1), ('leaflet', 1), ('clipboard', 1), ('rxjs', 1), ('polyfill', 1), ('fetch', 1), ('promise', 1), ('screenshot', 1), ('dom', 1), ('ember', 1), ('hacktoberfest', 1), ('ramda', 1), ('cookie', 1), ('interpreter', 1)
1-10	5	자바 기술	('java', 4), ('flow', 1), ('rxjava', 1), ('react', 1), ('guava', 1), ('algorithm', 1), ('jvm', 1), ('netty', 1), ('programming', 1), ('basic-java', 1), ('excel', 1), ('xlsx', 1), ('xls', 1), ('poi', 1), ('oom', 1)
1-11	5	어플리케이션 개발 프레임워크 ((electron)	('electron', 5), ('windows', 5), ('macos', 5), ('linux', 5), ('javascript', 2), ('nodejs', 2), ('react', 2), ('atom', 1), ('editor', 1), ('shell', 1), ('terminal', 1), ('desktop', 1), ('tron', 1), ('science-fiction', 1), ('touch', 1)

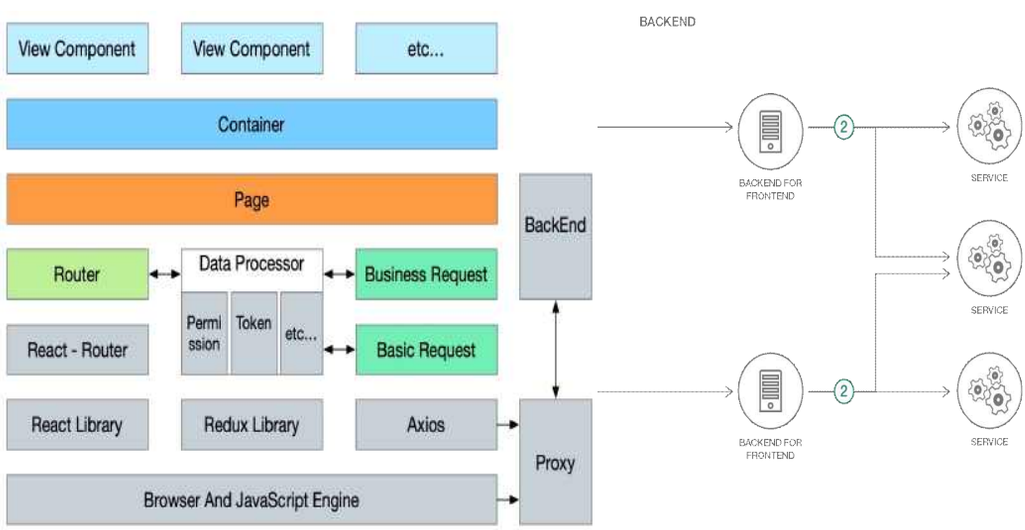
다. 상위 인지도 저장소 분석 소결

◈ 프로그램 개발자 중심의 생태계 구성

- 깃허브의 저장소들은 대부분 프로그램 개발자 또는 사용자가 많으며, 그중에서도 PC, 모바일 웹 개발의 범용적 어플리케이션, 플랫폼 개발이 주를 이루고 있음을 확인함

- 깃허브에서도 주요 기술은 Front-end, Back-end 아키텍쳐로 나뉘고 있으며, 통합 아키텍쳐 플랫폼 또는 전문 플랫폼 프로그램으로 개발되고 있음
  - 클러스터 1의 토픽을 살펴보면, Javascript, CSS 등 Front-end 기술과 SQL, NodeJS 등 Back-end 기술의 통합적인 오픈소스들로 구성되어, 그 수가 많이 나왔으리라 예상되며, 다른 상위 인지도 클러스터들은 각 아키텍쳐에서 전문적으로 개발되고 있음

그림 3-2 Front-end와 Back-end 구성도



※ 이미지 출처: Gong, Y., Gu, Feng, Chen, K., and Wang, F. (2020); IBM developer 홈페이지

◈ 오픈소스 조합을 통한 웹 Front-end 및 Back-end 기술 개발

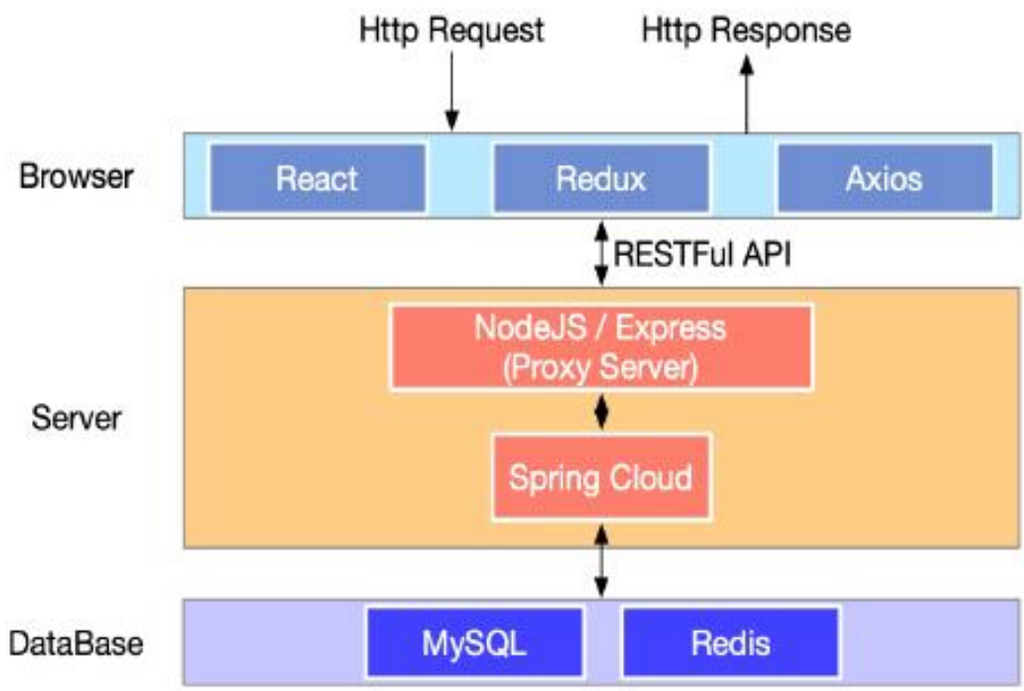
- Front-end, Back-end 기술과 관련한 오픈소스들이 깃허브에 공개됨에 따라, 관련 오픈소스 패키지 및 프로그램을 이용하고 조합하여, 개발상황에 적합한 웹 어플리케이션 개발이 가능하리라 고려됨

- 사례로서 캠퍼스 정보시스템에 적용된 Front-end, Back-end 기술 개발 사

례를 살펴보면, 깃허브의 상위 인지도 저장소 기술들의 충분한 활용가능성이 있음

- React, Redux, Axios는 클러스터 3과 11에 포함되는 Front-end 기술로 사용될 수 있으며, Back-end 기술 중 서버에 해당하는 NodeJS는 클러스터 1, 클러스터 4, 15, Spring은 클러스터 6, 데이터베이스에 해당하는 MySQL와 Redis는 클러스터 6의 오픈소스들을 활용할 수 있음

그림 3-3 서버 및 데이터베이스 관련 Back-end 기술구조 사례



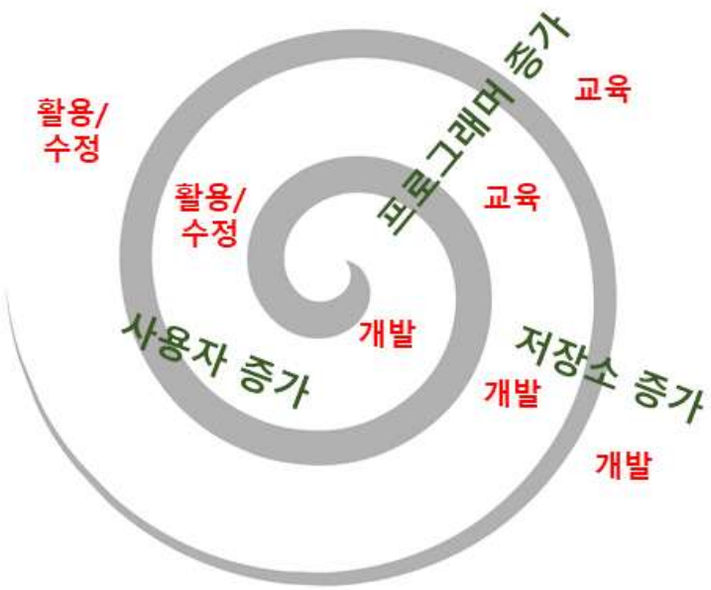
※ 이미지 출처: Gong, Y., Gu, Feng, Chen, K., and Wang, F. (2020)

◆ 소프트웨어 개발, 활용, 교육 등 협력 및 공유의 공간

- 개인-기업-연구소 등의 관계자들이 모여 프로그램을 개발하고, 수정해나가면서 고도화시키고 있는 것은 물론, 깃허브의 상위 저장소는 프로그램 또는 소프트웨어를 교육하는 목적으로 운영되고 있는 경우가 많음
- 개발·활용·교육 측면에서 검증과 개발을 반복하고, 새로운 프로그래머를 양성하는 등 소프트웨어 R&D 모델인 나선형(spiral) 구조의 발전 생태계를 자체적으로 형성하고 있음

- 나선형 구조의 생태계 발전은 개발과 활용/수정 단계에서 사용자가 증가하고, 이 오픈소스를 교육용으로 공유하면서 프로그래머가 증가하고, 그 프로그래머가 다시 저장소를 개발하여 저장소를 증가시키는 선순환 구조를 가짐
- 사용자 증가와 프로그래머 증가에 따른 오픈소스 개발 증가를 통한 선순환적 구조로 생태계가 발전하고 있음

그림 3-4 나선형 구조의 R&D 생태계



※ 이미지 출처: vectorstock.com/28382542

2 빅테크 기업 저장소 분석

가. 빅테크 기업 저장소 데이터 수집

◈ 빅테크 기업 현황

- 빅테크 기업으로는 미국, 중국, 한국의 주요 4차 산업혁명 기반 기업들을 대상으로, 깃허브 내에서 저장소 운영과 활동 관련 사항을 확인함
  - 미국의 구글, 마이크로소프트(MS), 인텔, 페이스북, 애플, 아마존, 중국의 BAT, 한국의 네이버, 카카오, 삼성전자를 대상으로 분석
  - 기업마다 주력 제품이나 서비스가 다르기 때문에, 깃허브의 운영 및 활동의 차이점을 살펴볼 필요가 있음
  - 기업들의 원천적인 경쟁우위가 웹 서비스, OS, 소프트웨어, 모바일, 전자제품 등 다른 상황이지만, 4차 산업혁명의 융합 시대에서 소프트웨어 활용영역이나 수준을 오픈소스에서 비교해볼 수 있음
- 깃허브의 공개 저장소의 수로는 MS가 4,378개, 구글이 2,121개, 인텔이 815개 등 미국의 빅테크 기업이 주도[6]하고 있음

나. 빅테크 기업 저장소 주제 분석

◈ 빅테크 기업 주요 저장소 및 토픽

- 빅테크 기업의 저장소의 주제는 범용적인 웹 또는 기계학습 기술 위주의 저장소가 다수 확인되나, 기업별로 자체적으로 사용하고 있는 플랫폼 위주의 front-end, back-end 기술이 주로 개발되고 있는 경향이 있음
  - 구글이나 MS의 경우, 다른 기업보다 많은 저장소를 가지고 운영 중이며, 기계학습·딥러닝 관련의 소프트웨어 원천지식기술을 많이 운영하고 있으며, 동영상(Exoplayer)이나 자율주행(AirSim)과 같은 응용분야에 기반한 저장소도 운영함
  - 인텔은 하드웨어 기반의 최적화 기술이나 기계학습 기술이 다수 보이며, 페이스북과 애플은 모바일이나 웹 서비스 측면에서 Front-end 기술(react) 활용이 많았으며, 앱 개발에 용이한 자체 API 플랫폼 기술(create-react-app, facebook-ios-sdk, swift 등)도 깃허브에서 많이 운영하고 있음
  - 아마존의 경우는 자체 웹서비스인 AWS(Amazon Web Service) 기반의 클라우드 컴

6) 2021.10.7. 기준

퓨팅 서비스를 위한 응용 프로그램 분야 등 Back-end 기술이 주로 개발하고 있으며, 상거래를 위한 데이터베이스나 서버 등의 저장소(lambda)도 제공하고 있음

- 중국과 한국 기업은 자체적으로 개발한 앱이나 OS 기반으로 저장소를 운영하고 있으며, 미국의 범용적인 기술보다는 회사의 주력 제품 및 서비스 분야에 특화된 기술 위주로 개발하고 있음
  - 중국의 바이두는 검색엔진 서비스 회사로 front-end 웹 기술 위주로 저장소를 가지고 있으며, 알리바바는 전자상거래 회사로 back-end 웹 기술 위주로 클라우드 등 관련 기술, 텐센트는 자사의 대표적 앱인 위챗(wechat) 기반의 인터페이스 기술을 개발
  - 한국의 네이버는 웹의 front-end에서 시각화된 정보를 게시하기 위한 API를 제공하고 있으며, 카카오 역시 front-end의 시각화된 정보 제공 기술 위주로 자연어 처리 저장소를 가지고 있다는 점이 특기할 만한 점이며, 삼성전자는 자사가 개발에 참여한 타이젠 OS를 이용한 멀티플랫폼 범용 기술이나 GearVR 관련 벡터 계산 등 저장소 등 전자제품에서 적용 가능한 저장소를 활용하고 있음

표 3-4 빅테크 기업의 깃허브 주요 저장소 및 토픽

기업	저장소 수	주요 저장소(스타 수 순서)	주요 토픽
Google	2,121	Guava, Material-design-lite, styleguide, leveldb, googletest, zx, iosched, python-fire, gson, web-starter-kit, Exoplayer, flexbox-layout, flatbuffers, dagger, fonts, jax, mediapipe, libphonenumber, cadvisor, WebFundamentals, yapf,	android, go, golang, java, tensorflow, machinelearning, deep-learning, security, kotlin, javascript,
MS	4,378	vscode, terminal, TypeScript, PowerToys, Web-Dev-For-Beginners, playwright, monaco-editor, calculator, ML-For-Beginners, cascadia-code, api-guidelines, CNTK, winget-cli, react-native-windows, vcpkg, LightGBM, dotnet, fluentui, AirSim, recommenders, frontend-bootcamp, nni	azure, vscode, typescript, python, machinelearning, makecode, hactoberfest, deep-learning, powershell, aiforearth, react, pytorch
Intel	815	hyperscan, acat, appframework, haxm, nemu, linux-sgx, caffe, copmuter-runtime, isa-l, fastuidraw, llvm, media-driver	swrepo, hardware, machinelearning, deeplearning, llvm
Facebook	111	react, create-react-app, jest, docusaurus, flow, rocksdb, draft-js, folly, flux, hhvm, fresco, relay, zstd, prepack, prophet, infer, stetho, react-devtools, watchman, chisel, buck, litho, proxygen, facebook-ios-sdk, jscodeshift, hermes, pyre-check	react, javascript, java

Apple	137	swift, swift-evolution, foundationdb, turicreate, swift-package-manager, swift-nio, swift-corelibs-foundation, swift-protobuf	swift, swiftnio
Amazon (AWS)	301	aws-cli, chalice, serverless-application-model, aws-cdk, aws-sdk-go, aws-sdk-js, amazon-sagemaker-examples, aws-sam-cli, aws-sdk-php, s2n-tls, containers-roadmap, aws-sdk-java, aws-sdk-ruby, aws-lambda-go, amazon-freertos, copilot-cli, amazon-ecs-agent, aws-sdk-net	AWS, hactoberfest, aws-lambda, mxnet, keyspace, kubernetes, cassandra, sigv4
Baidu	98	amis, san, uid-generator, braft, bfs, lac, Familia, AnyQ, sofa-pbrpc, openrasp, tera, Senta, bigflow, CUP, DuReader, bfe-book, BaikalDB, brpc-java, DDParser, NoahV	san, dialogue, word-based(chinese, word, speech, parser)
Alibaba	381	arthas, p3c, druid, fastjson, flutter-go, easyexcel, canal, spring-cloud-alibaba, nacos, weex, Sentinel, ice, ARouter, tengine, vlayout, DataX, atlas, rax, hooks	java, spring, react, android, flutter, kubernetes, tangram,
Tencent	133	weui, wepy, tinker, mars, weui-wxss, vConsole, MMKV, QMUI_Android, ncnn, omi, APIJSON, VasSonic, rapidjson, wcdb, matrix, secguide, xLua, libco, QMUI_iOS, Hippy	android, wechat, blueking, java, microservice
Naver	189	billboard.js, fe-news, egjs-flicking, ngrinder, d2codingfont, egjs-infinitegrid, egjs, sqlova, biobert-pretrained, deep-image-retrieval, yobi, android-pull-to-refresh, r2d2, arcus, lucy-xss-filter, egjs-view360, android-imagecropview, claf, kapture, smarteditor2	egjs, javascript, react, typescript,
Kakao	40	khaiii, buffalo, n2, s2graph, DaumEditor, kakao_flutter_sdk, web2app, awesome-tech-newletters, hbase-region-inspector, network-node-manager, cuesheet, credit-card-sms-parser, adt, d2hub, hbase-tools, mango, java_thread_dump_analyzer, hbase-packet-inspector, cmux	machinelearning, hbase,
Samsung	134	veles, rlottie, TizenRT, GearVRf, netcoredbg, jalangi2, ADBI, cotopaxi, ONE, Tizen-CSharp-Samples, TizenTVApps, KnowledgeSharingPlatform, react-native-tizen-dotnet, TizenFX, libtuv, ChromiumGStreamerBackend, cordova-plugin-toast	svg, javascript

◆ 구글의 깃허브 저장소 현황

- 구글의 상위 인지도 저장소 현황으로는 안드로이드용 머티리얼 디자인부터 시작으로 자바, 구아바(구글이 작성한 자바 오픈소스 라이브러리), 자바스크립트, 또는 파이썬 기반의 저장소, 컨텐츠 사용 저장소 등이 있음
- 기본적인 프로그램 언어 및 프레임워크 외에도 브랜치, 리퀘스트, 커밋 등 다른 속성을 확인할 경우, 미디어 재생 라이브러리인 Exoplayer 등 컨텐츠 관련 기술 저장소 역시 활성화되어 있음

표 3-5 구글의 깃허브 주요 저장소 현황

저장소 이름	Star 수	Branch 수	Request 수	Commit 수	관련 Topic
material-design-icons	44,000	2	12	154	'android', 'ios', 'web', 'material', 'material-design', 'sprites', 'icons'
guava	42,500	15	76	5,652	'java', 'guava'
material-design-lite	31,900	74	27	2,872	'material', 'material-design', 'material-components', 'mdl', 'material-design-lite'
styleguide	28,900	9	108	436	'styleguide', 'style-guide', 'cpplint'
zx	23,100	1	2	203	'nodejs', 'javascript'
iosched	20,800	6	18	3,101	'android', 'kotlin', 'conference', 'architecture', 'coroutines'
python-fire	20,200	5	26	259	'python', 'cli'
ExoPlayer	18,300	13	25	11,890	'android', 'java', 'exoplayer', 'mediaplayer'
flexbox-layout	17,000	4	9	370	'android', 'flexbox', 'android-library'
flatbuffers	16,900	10	23	2,374	'javascript', 'python', 'c', 'java', 'go', 'c-sharp', 'rust', 'c-plus-plus', 'serialization', 'typescript', 'protobuf', 'cross-platform', 'flatbuffers', 'zero-copy', 'marshalling', 'grpc', 'rpc', 'json-parser', 'mmap', 'serialization-library'

◈ 구글의 깃허브 저장소 주요 유형

- 구글이 운영하고 있는 깃허브의 저장소 중 토픽을 제시한 총 466개의 저장소를 대상으로 k-means 클러스터링을 수행하여, 주요 유형을 살펴봄
- 안드로이드를 이용한 기계학습이나 크롬 웹 서비스가 기술 저장소로 가장 많은 양을 나타냈으며, 음성/언어 기술, 3d 이미지 기술, 구글 원격 프로그램(gRPC: Google Remote Procedure Call) 등 다양한 컨텐츠 서비스를 기술 저장소로 운영하고 있음

표 3-6 구글의 깃허브 저장소 주요 유형

군집	수	내용	주요 토픽(괄호 안의 숫자는 토픽 출현 횟수)
1	83	안드로이드 기계학습	('android', 9), ('golang', 7), ('java', 5), ('rust', 5), ('python', 4), ('web', 4), ('google', 4), ('deep-learning', 4), ('tensorflow', 4), ('go', 4), ('test', 4), ('static', 3), ('machinelearning', 3), ('kotlin', 3), ('security', 2)
2	72	안드로이드, 크롬 웹 사용	('android', 7), ('security', 4), ('git', 4), ('golang', 4), ('google', 3), ('chrome', 3), ('ssh', 3), ('python', 3), ('test', 3), ('deep-learning', 3), ('javascript', 2), ('java', 2), ('gwt', 2), ('j2cl', 2), ('rust', 2)
3	52	암호/보안 기술	('crypto', 5), ('go', 4), ('golang', 3), ('test', 3), ('deep-learning', 3), ('chrome', 3), ('python', 3), ('swift', 3), ('forensics', 2), ('vision', 2), ('bioinformatics', 2), ('stackdriver-monitoring', 2), ('nodejs', 2), ('docker', 2), ('kubernetes', 2)
4	42	음성, 언어 등 기계학습	('python', 9), ('machinelearning', 4), ('android', 4), ('golang', 3), ('java', 3), ('google', 3), ('audio', 2), ('speech', 2), ('javascript', 2), ('pipeline', 2), ('perception', 2), ('open-source', 2), ('ios', 2), ('computer-vision', 2), ('dependency-injection', 2)
5	42	자바 기술	('java', 4), ('test', 4), ('javascript', 3), ('go', 3), ('google', 3), ('api', 3), ('python', 2), ('rust', 2), ('json', 2), ('security', 2), ('closure', 2), ('ssh', 2), ('automation', 2), ('performance', 2), ('tensorflow', 2)
6	41	3d 이미지 기술	('android', 8), ('kotlin', 4), ('vulkan', 3), ('machinelearning', 3), ('deep-learning', 3), ('security', 3), ('jax', 3), ('spirv', 2), ('computer-vision', 2), ('docker', 2), ('sandbox', 2), ('sql', 2), ('vedio', 2), ('hugo-theme', 2), ('hugo', 2)
7	38	기계학습 테스팅	('test', 5), ('go', 4), ('security', 4), ('android', 4), ('tensorflow', 4), ('machinelearning', 4), ('java', 3), ('cpp', 3), ('jax', 3), ('neural', 3), ('golang', 3), ('data', 3), ('web', 3), ('python', 3), ('crypto', 2)
8	34	구글 원격 프로그램	('security', 5), ('cpp', 4), ('nodejs', 3), ('web', 3), ('nlp', 3), ('grpc', 3), ('grpc-service', 3), ('picoprod', 3), ('angular', 2), ('benchmark', 2), ('linux', 2), ('machinelearning', 2), ('ctf', 2), ('test', 2), ('computer-vision', 2)
9	34	프로그래밍 언어	('docker', 4), ('crypto', 3), ('typescript', 3), ('go', 3), ('javascript', 3), ('java', 3), ('cplusplus', 3), ('test', 3), ('grpc-service', 3), ('picoprod', 3), ('privacy', 2), ('android', 2), ('semantic-web', 2), ('schema-org', 2), ('json-ld', 2)
10	28	머티리얼 디자인	('android', 5), ('web', 4), ('material', 4), ('jax', 2), ('vulkan', 2), ('security', 2), ('python', 2), ('google', 2), ('deep-learning', 2), ('html', 2), ('firestore', 2), ('real-time', 1), ('open-source', 1), ('metal', 1), ('graphics', 1)

◆ MS의 깃허브 저장소 현황

- MS의 상위 인지도 위주의 깃허브 주요 저장소 역시 웹 기술(front-end) 관련 저장소가 주로 나타나고 있으며, 윈도우와 비주얼 스튜디오를 플랫폼으로 하는 저장소도 많이 나타나고 있음
  - 일반적으로 구글보다 대부분 속성에서 더 높은 값을 나타내는 등 MS 관련 저장소가 더 활성화되어 있음을 확인함(이는 깃허브를 MS가 인수하고 적극적으로 활용하기 때문으로 해석할 수도 있음)
  - 그러나 구글의 경우, 텐서플로우(스타 수 161,000)와 같이 구글이 독립적으로 운영하고 있는 다른 저장소들이 많기 때문으로 볼 수도 있음
- 기계학습 분야에서도 초심자를 위한 교육 용도로 저장소(ML-For-Beginners)를 운영하고 있으며, 자체 기계학습 도구로서 CNTK(Cognitive Toolkit)가 활성화됨을 확인함

표 3-7 MS의 깃허브 주요 저장소 현황

저장소	Star 수	Branch 수	Request 수	Commit 수	관련 Topic
vscode	122,000	429	252	88,190	'electron', 'microsoft', 'editor', 'typescript', 'visual-studio-code'
terminal	77,600	282	45	2,572	'console', 'terminal', 'command-line', 'wsl', 'cmd', 'windows-console', 'windows-terminal'
TypeScript	74,700	481	265	32,639	'javascript', 'language', 'typechecker', 'typescript'
PowerToys	61,300	30	13	5,416	'windows', 'color-picker', 'desktop', 'keyboard-manager', 'powertoys', 'fancyzones', 'microsoft-powertoys', 'powerrename'
Web-Dev-For-Beginners	35,500	2	5	1,113	'javascript', 'css', 'html', 'learning', 'education', 'curriculum'
playwright	27,800	22	33	5,907	'electron', 'javascript', 'testing', 'firefox', 'chrome', 'automation', 'web', 'webkit', 'hacktoberfest', 'e2e-testing', 'playwright'

monaco-editor	26,300	2	4	664	'editor', 'typescript', 'browser', 'vscode', 'monaco-editor'
calculator	22,800	28	5	695	'windows', 'xaml', 'csharp', 'cpp', 'uwp', 'windows-10'
ML-For-Beginners	22,700	2	4	1,265	'python', 'data-science', 'machine-learning', 'scikit-learn', 'machine-learning-algorithms', 'ml', 'machinelearning', 'hacktoberfest', 'machinelearning-python', 'scikit-learn-python'
CNTK	17,100	1,069	85	16,116	'python', 'java', 'c-sharp', 'c-plus-plus', 'machine-learning', 'deep-neural-networks', 'deep-learning', 'neural-network', 'cntk', 'distributed', 'cognitive-toolkit'

◈ MS의 깃허브 저장소 주요 유형

- MS가 운영하는 저장소의 특징을 살펴보기 위해, 토픽을 가지고 있는 총 902개의 저장소를 k-means 클러스터링을 이용해 10개로 유형화하였음
- MS는 구글과 다른 측면으로 클라우드 기반의 기계학습 저장소인 azure를 바탕으로 알고리즘, 데이터베이스, 이미지, 자연어 처리, 플랫폼 등을 활성화하고 있음이 특기할 만한 점으로 나타남
- 프로그래밍 위주로 편집하는 코드 교육(클러스터 7)이나 C++을 이용한 기계학습 등 다양성을 추구하고 있으며, 펌웨어를 오픈소스 저장소(클러스터 10)를 이용해 MS 위주의 기기 최적화를 제공하고 있음
- 무엇보다 MS의 경우, “aiforearth”라는 주제로 관련 저장소를 다수 운영하면서 환경보호 챌린지를 국제적으로 활성화하고 있어, 기업의 사회적 책임 또는 ESG 경영을 오픈소스를 이용해 수행하고 있음을 살펴볼 수 있음

표 3-8 MS의 깃허브 저장소 주요 유형

군집	수	내용	주요 토픽(괄호 안의 숫자는 토픽 출현 횟수)
1	157	클라우딩 기계학습, 인공지능	('machinelearning', 18), ('microsoft', 17), ('azure', 17), ('windows', 15), ('test', 14), ('python', 12), ('vision', 12), ('vscode', 11), ('react', 11), ('deep-learning', 10), ('typescript', 8), ('csharp', 7), ('ai', 7), ('pytorch', 6), ('hacktoberfest', 6)
2	98	클라우딩 기계학습, 데이터베이스	('microsoft', 14), ('azure', 14), ('machinelearning', 13), ('typescript', 11), ('python', 10), ('hacktoberfest', 9), ('react', 9), ('windows', 7), ('deep-learning', 6), ('data', 6), ('java', 5), ('vscode', 5), ('vision', 5), ('sql', 4), ('csharp', 4)
3	93	클라우딩 기계학습 이미지, 자연어처리	('azure', 20), ('deep-learning', 17), ('machinelearning', 16), ('microsoft', 15), ('computer-vision', 11), ('nlp', 10), ('python', 9), ('java', 8), ('cosmosdb', 7), ('typescript', 6), ('csharp', 6), ('neural', 6), ('data', 5), ('kubernetes', 5), ('tensorflow', 5)
4	88	클라우딩 플랫폼, 개발 커스텀	('azure', 22), ('microsoft', 12), ('sdk', 6), ('telemetry', 6), ('nlp', 5), ('nodejs', 5), ('monitoring', 5), ('application-insights', 5), ('logging', 5), ('typescript', 4), ('docker', 4), ('kubernetes', 4), ('python', 4), ('machinelearning', 4), ('powershell', 4)
5	87	기계학습 코딩	('machinelearning', 13), ('microsoft', 11), ('makecode', 11), ('microbit', 9), ('python', 9), ('azure', 6), ('data', 6), ('scikit-learn', 5), ('template', 5), ('cookiecutter', 5), ('vscode-extension', 4), ('vscode', 4), ('javascript', 4), ('typescript', 4), ('pxt', 4)
6	86	기계학습 코딩	('microsoft', 8), ('python', 8), ('azure', 8), ('vision', 7), ('hacktoberfest', 7), ('machinelearning', 7), ('data', 6), ('msftnet', 6), ('qsharp', 5), ('windows', 5), ('bot', 5), ('microsoft-bot-framework', 5), ('typescript', 4), ('cpp', 4), ('quantum', 4)
7	78	프로그래밍 편집	('makecode', 9), ('hol', 7), ('ocp-isv', 7), ('vision', 6), ('test', 5), ('nodejs', 5), ('microsoft', 4), ('dotnet', 4), ('sdk', 4), ('azure', 4), ('vscode', 4), ('jacdac', 4), ('uwp', 3), ('cpp', 3), ('windows', 3)
8	76	환경보호 기술 챌린지	('azure', 13), ('aiforearth', 8), ('git', 6), ('java', 6), ('python', 5), ('react', 5), ('kubernetes', 5), ('api', 4), ('typescript', 4), ('docker', 4), ('android', 3), ('nodejs', 3), ('powerbi', 3), ('powershell', 3), ('test', 3)
9	70	C++ 기계학습	('microsoft', 11), ('cpp', 10), ('azure', 8), ('python', 7), ('mcw', 5), ('nlp', 5), ('sdk', 5), ('cognitive-services', 5), ('csharp', 4), ('typescript', 4), ('react', 4), ('javascript', 4), ('machinelearning', 4), ('git', 4), ('sample', 4)
10	69	펌웨어/가상현실(홀로그램)	('projectmu', 6), ('windows', 6), ('azureml', 6), ('mixed-reality', 5), ('makecode', 5), ('microsoft', 4), ('python', 4), ('test', 3), ('hololens', 3), ('tfs', 3), ('vsts', 3), ('work-item-control', 3), ('vsts-extension', 3), ('uefi', 3), ('mlops', 3)

다. 빅테크 기업 저장소 분석 소결

◈ 기업 활동 특성에 따른 오픈소스 활동

- 빅테크 기업들은 기업의 IT 특징에 맞춰, 구글이나 MS는 범용적인 기계학습이나 웹 플랫폼 기술(front-end, back-end)을 개발하고 있음
- 기업 활동에 특화되어, 하드웨어 기반 최적화 또는 서비스 기술(인텔, 삼성전자), 웹 front-end 서비스(페이스북/애플), 서버 및 클라우드 등 back-end 서비스(아마존, 알리바바 등) 기술 등이 집중되어 개발되고 있음
- 또한, MS의 경우는 사회적 활동을 목적으로 환경보호 챌린지를 위한 저장소를 운영하는 등 기업의 사회적 책임이나 ESG 경영 등에 있어서도 오픈소스가 사용될 가능성을 확인할 수 있음

◈ 기업의 자체 플랫폼 주도의 오픈소스 활동

- 빅테크 기업들은 서비스에 따라 대부분 자체 프레임워크(android, azure, react, AWS 등) 또는 언어(go, swift 등)를 사용하고 있으며, 오픈소스 역시 해당 프레임워크 또는 언어를 중심으로 발전되고 있음
- 빅테크 기업들이 자체 프레임워크 및 언어를 개발함으로써, 사용자를 늘려 개발 시장의 주도권을 잡고, 사업활동을 확장하고 제품개발의 편의성을 확보하려는 오픈소스 활동의 시도라고 고려됨

◈ 기반 기술 위주 공개 저장소의 데이터 분석 한계

- 현재 공개(public)로 운영하고 있는 기업의 저장소들은 대부분 기반 기술들로서 범용적인 내용을 담고 있다면, 사설(private)로 사용하는 저장소는 목적이나 대상 위주의 더욱 구체적인 프로그램 개발이 진행될 가능성이 큼
- 공개 저장소는 기반 기술을 공개하고 수정, 검증하여 개선시키는 활동만을 본다는 점에서, 실제 빅테크 기업의 시장 진출을 위한 기술전략이나 제품개발 활동에 오픈소스가 미치는 영향을 분석하는데는 한계가 있음

3 미래기술 저장소 분석

가. 자율주행자 저장소 분석

◆ 자율주행차 저장소 및 토픽 현황

- 자율주행차 관련 깃허브 오픈소스 소프트웨어 개발현황을 살펴보기 위해, “autonomous vehicles”의 토픽을 가지는 저장소를 10개로 군집화함
  - 240개의 저장소가 가진 토픽을 바탕으로 k-means 클러스터링 알고리즘을 사용해 10개로 군집화하였음

※ 실제 900개 이상의 저장소가 검색되나, 스타 수가 10개 이상인 저장소만 선택하여 군집화하였음

- 전체적으로 토픽은 컴퓨터 비전·기계학습·시뮬레이션·주행제어 등 데이터 처리 및 학습 기술, ROS·로보틱스 등 로봇 기술, 그리고 lidar(light detection and ranging), slam(simultaneous localization and mapping), 차선, 신호등 등 신호처리와 관련한 정보 자원들이 도출되었음

표 3-9 자율주행차 관련 깃허브 저장소의 군집 및 주요 토픽

군집	수	내용	주요 토픽(괄호 안의 숫자는 토픽 출현 횟수)
1	92	차선감지 Lidar 기술 맵핑 기술 주행제어	('robotics', 13), ('detection', 10), ('path-planning', 8), ('lidar', 8), ('slam', 7), ('algorithm', 5), ('localization', 5), ('control-systems', 5), ('mapping', 4), ('cpp', 4), ('navigation', 4), ('reinforcement-learning', 4), ('simulation', 4)
2	39	기계학습 기업프로젝트 시뮬레이터	('awesome', 5), ('robotics', 5), ('machinelearning', 4), ('python', 4), ('udacity', 4), ('apollo', 3), ('reinforcement-learning', 3), ('simulator', 3), ('carla', 3), ('carla-simulator', 3), ('perception', 3), ('mpc', 3), ('software', 3)
3	25	ROS 시뮬레이터 컴퓨터비전	('ros', 25), ('robotics', 10), ('robot', 5), ('open-source', 4), ('detection', 3), ('simulator', 3), ('ai', 3), ('simulation', 3), ('robots', 3), ('agv', 3), ('computer-vision', 2), ('robot-simulator', 2), ('webots', 2), ('robotics-simulation', 2)
4	21	딥러닝 기계학습 강화학습	('deep-learning', 22), ('computer-vision', 7), ('machinelearning', 5), ('reinforcement-learning', 5), ('neural', 3), ('simulation', 2), ('robotics', 2), ('pytorch', 2), ('carla', 2), ('artificial-intelligence', 2), ('intelligent-transportation-systems', 2), ('python', 2), ('recognition', 2), ('detection', 2)

5	17	컴퓨터비전 차선감지 차선추적	('python', 8), ('computer-vision', 8), ('lane-detection', 5), ('lane-detector', 4), ('autonomous-car', 4), ('lane-finding', 3), ('open-source', 3), ('tracking', 2), ('lane-lines', 2), ('adas', 2), ('lane-lines-detection', 2), ('tensorflow', 2), ('cnn', 1), ('pytorch', 1)
6	16	딥러닝 컴퓨터비전 시뮬레이터	('deep-learning', 15), ('computer-vision', 6), ('machinelearning', 6), ('artificial-intelligence', 5), ('python', 4), ('convolutional-neural-networks', 3), ('autonomous-car', 3), ('ros', 2), ('carla', 2), ('carla-simulator', 2), ('tensorflow', 2), ('keras', 2), ('behavioral-cloning', 2)
7	14	딥러닝 컴퓨터비전 데이터	('deep-learning', 14), ('computer-vision', 9), ('convolutional-neural-networks', 9), ('tensorflow', 6), ('neural', 5), ('cnn', 4), ('artificial-intelligence', 3), ('keras', 3), ('ros', 3), ('autonomous-car', 3), ('lane-detection', 3), ('dataset', 3), ('semantic-segmentation', 3), ('machinelearning', 2)
8	12	ROS 컴퓨터비전 RC 카	('autonomous-car', 5), ('ros', 5), ('computer-vision', 3), ('open-source', 3), ('neural', 3), ('ros-melodic', 3), ('deep-learning', 2), ('autonomous', 2), ('selfdriving', 2), ('rc-car', 2), ('self-driving', 2), ('self-driving-cars', 2), ('python', 2)
9	2	딥러닝 신호등 차량제어	('neural', 2), ('deep-learning', 2), ('ros', 2), ('autonomous', 2), ('autonomous-car', 2), ('traffic-light', 2), ('ssd-mobilenet', 2), ('vehicle-control', 2)
10	2	유다시티 우분투 캡스톤	('car', 2), ('udacity', 2), ('ubuntu', 2), ('capstone', 2), ('ros', 2), ('automation', 2), ('traffic-light', 2)

- 클러스터링 결과 기업 기반 클러스터가 자율주행차 관련 깃허브의 대부분을 나타냈으며, 기술보다는 기업이 주도하여 생태계가 구성됨을 확인 가능함
  - Apollo(클러스터 2), Carla(클러스터 2, 6), AirSim(클러스터 8), Autoware(클러스터 3) 위주로 깃허브의 오픈소스로 자율주행 기술이 개발되고 있음
  - Apollo는 중국의 Baidu, Carla는 인텔/도요타의 후원, AirSim은 MS, Autoware는 일본 국립산업기술종합연구소의 프로젝트로 미국-중국-일본의 자율주행차 기술이 공개 저장소로 활성화되고 있음
  - 기술 클러스터로 감지기술, 위치추정, 지도기술(slam), 기계학습, 내비게이션, 시뮬레이터 기술들도 자율주행의 주요 토픽으로 도출되었음
  - 자율주행 테스트용 RC 카(클러스터 8)나 산업용 이송로봇(agv, 클러스터 3)에 대한 자율주행도 오픈소스로 개발되고 있음
  - 유다시티(udacity), 캡스톤 등 자율주행 관련하여 교육용으로 오픈소스를 개발한 사례도 보이고 있음

표 3-10 자율주행차 관련 깃허브 저장소의 주요 개발자

군집	수	주요 개발자
1	92	'AtsushiSakai/PythonRobotics', 'generalized-intelligence/GAAS', 'pptacher/probabilistic_robotics', 'h1st-ai/h1st', 'BichenWuUCB/SqueezeSeg', 'zhm-real/MotionPlanning', 'AtsushiSakai/MATLABRobotics', 'karanchawla/GPS_IMU_Kalman_Filter', 'vojtamolda/autodrome', 'intel/ad-rss-lib'
2	39	'ApolloAuto/apollo', 'daohu527/dig-into-apollo', 'thibo73800/metacar', 'Amin-Tgz/awesome-CARLA', 'daohu527/awesome-self-driving-car', 'maudzung/RTM3D', 'philbort/awesome-self-driving-cars', 'cedricxie/apollo_perception_ros', 'erdos-project/pylot', 'abhisheknaik96/MultiAgentTORCS'
3	25	'Autoware-AI/autoware.ai', 'cyberbotics/webots', 'linorobot/linorobot', 'AutoRally/autorally', 'ArduPilot/apm_planner', 'usdot-fhwa-stol/carma-platform', 'kostaskonkk/datmo', 'StatueFungus/autonomous_driving', 'RiccardoGiubilato/ros_autonomous_car', 'SMARTlab-Purdue/ros-tutorial-gazebo-simulation', 'linorobot/ros_dwm1000', 'Autonomous-Racing-PG/ar-tu-do', 'IvLabs/autonomous-delivery-robot', 'mrsd16teamd/loco_car', 'uf-mil/NaviGator
4	21	'manfreddiaz/awesome-autonomous-vehicles', 'AndreiBarsan/DynSLAM', 'StanfordASL/Trajectron-plus-plus', 'ika-rwth-aachen/Cam2BEV', 'dotchen/LearningByCheating', 'jiachenli94/Awesome-Decision-Making-Reinforcement-Learning', 'mohamedameen93/German-Traffic-Sign-Classification-Using-TensorFlow', 'anshulpaigwar/GndNet', 'IvLabs/Stair-Climber', 'datacluster-labs/Datacluster-Datasets', 'sapan-ostic/deep_prediction', 'thayerAlshaabi/DeepEye'
5	17	'cfzd/Ultra-Fast-Lane-Detection', 'tier4/AutowareArchitectureProposal.proj', 'mmajewsk/Tonic', 'aljosaosep/ciwt', 'visualbuffer/copilot', 'vamsiramakrishnan/AdvancedLaneLines', 'eriklilland/pyMHT', 'ahmdtaha/constrained_attention_filter', 'RuPingCen/mick_robot', 'kenshiro-o/CarND-Advanced-Lane-Lines', 'cfizette/road-sign-cascades', 'mohamedameen93/Lane-lines-detection-using-Python-and-OpenCV'
6	16	'carla-simulator/carla', 'microsoft/AutonomousDrivingCookbook', 'sigmaai/self-driving-golf-cart', 'MankaranSingh/Auto-Birds-Eye', 'DeepTecher/SecondaryAwesomeCollection', 'init27/MIT-6.S094-Deep-Learning-for-Self-Driving-Cars', 'enginBozkurt/carla-training-data', 'Ekim-Yurtsever/DeepTL-Lane-Change-Classification', 'Zhenye-Na/e2e-learning-self-driving-cars'
7	14	'jiachenli94/Awesome-Interaction-aware-Trajectory-Prediction', 'dctian/DeepPiCar', 'datlife/jetson-car', 'fabvio/ld-lsi', 'ndrplz/dreyeve', 'fabvio/Cascade-LD', 'fabvio/TuSimple-lane-classes', 'lgsvl/lanefollowing', 'javiermcebrian/glcapsnet', 'singlavinod/Self-Driving-Car-NanoDegree-Udacity', 'bborja/wasr_network', 'bborja/modd', 'naiveHobo/Rambo', 'sakaridis/MGCDA
8	12	microsoft/AirSim', 'DeepTecher/awesome-autonomous-vehicle', 'Habrador/Self-driving-vehicle', 'enginBozkurt/Visualizing-lidar-data', 'GigaFlopsis/rc_car_ros', 'yconst/burro', 'jmscslgroup/catvehicle', 'szenergy/szenergy-public-resources', 'benjaminykim/self-driving-car-simulator', 'Ansheel9/End-to-End-Self-Driving-Car', 'dineshresearch/Autonomouscar', 'khanhvu207/FPT-DigitalRace2020

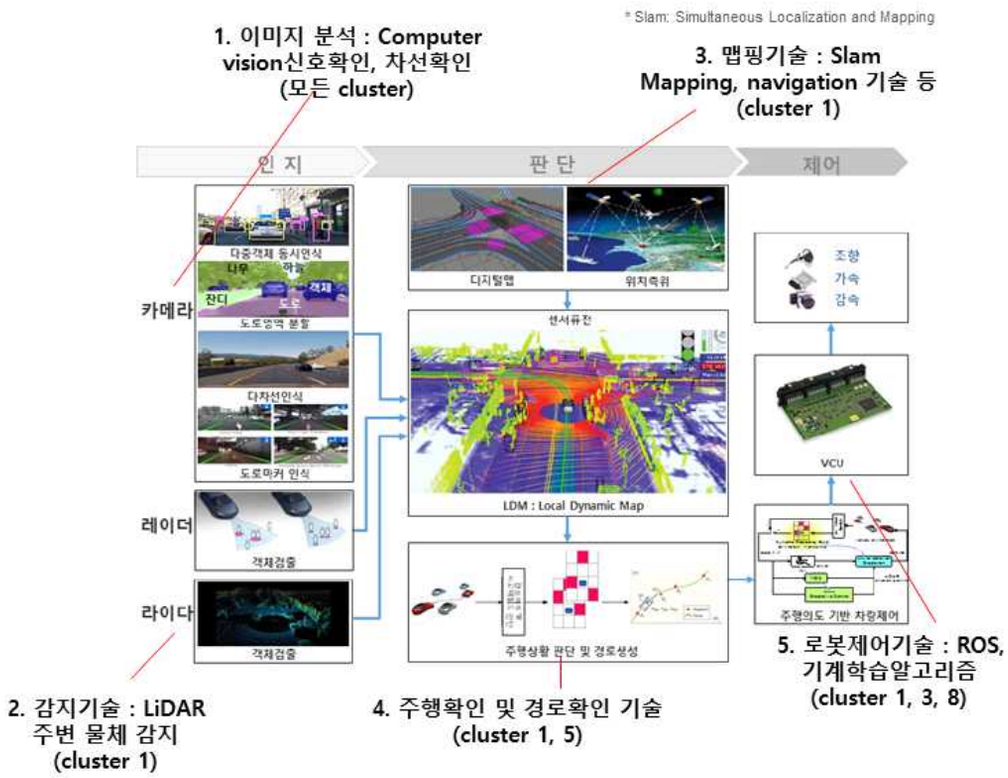
표 3-11 깃허브 저장소의 주요 개발자 현황

기업	Star 수	Branch 수	Request 수	Commit 수	Folk 수	Created	Language
Apollo	19,595	27	51	18,019	7,859	2017.7.4	C++
Carla	6,862	134	16	5,439	2,100	2017.10.24	C++, Python
AirSim	12,396	17	19	3,121	3,487	2017.2.14	C++, C#
Autoware	4,999	1	0	3,570	2,022	2015.8.24	-

◈ 자율주행차 오픈소스의 제품 기여도 분석

- 자율주행차 관련 저장소 및 오픈소스 코드가 현재 자율주행차의 감지기술, 위치추적기술, 이미지 분석, 로봇제어기술 연구개발에 활용 가능함
- 자율주행차는 주요기술은 인지-판단-제어에 따라 감지기술, 위치기술, 전자제어기능, 이미지처리기능, 레이더기술, 보안기술이라는 점에서, 현재 깃허브에서 공유되고 개발되고 있는 기술분야들이 많음
  - Lidar 기반의 감지기술이나 카메라를 이용한 차선 및 신호등 확인 등의 데이터 처리 및 기계학습 기술과 GPS 등 위치추적을 이용한 추적기능 및 지도형성기능, 내비게이션 등도 slam 기술을 통해 개발되고 있음
  - 데이터 처리를 통한 차량의 제어 및 운전을 위해서는 로봇운영체제 및 기계학습알고리즘이 사용되고 있음
  - 그러나 레이더 기술이나 보안기술 측면에서는 깃허브의 공개 오픈소스 저장소에서는 많이 활용되고 있지 않음

그림 3-5 자율주행차 기술구조도 및 오픈소스 활용 가능성



※ 이미지 출처: 윤경수, 김봉섭(2021)

- 자율주행차는 비교적 장기간 연구된 분야이며, 이에 따라 오픈소스에서도 다양한 분야의 기술이 이미 개발되고 있음을 살펴볼 수 있음

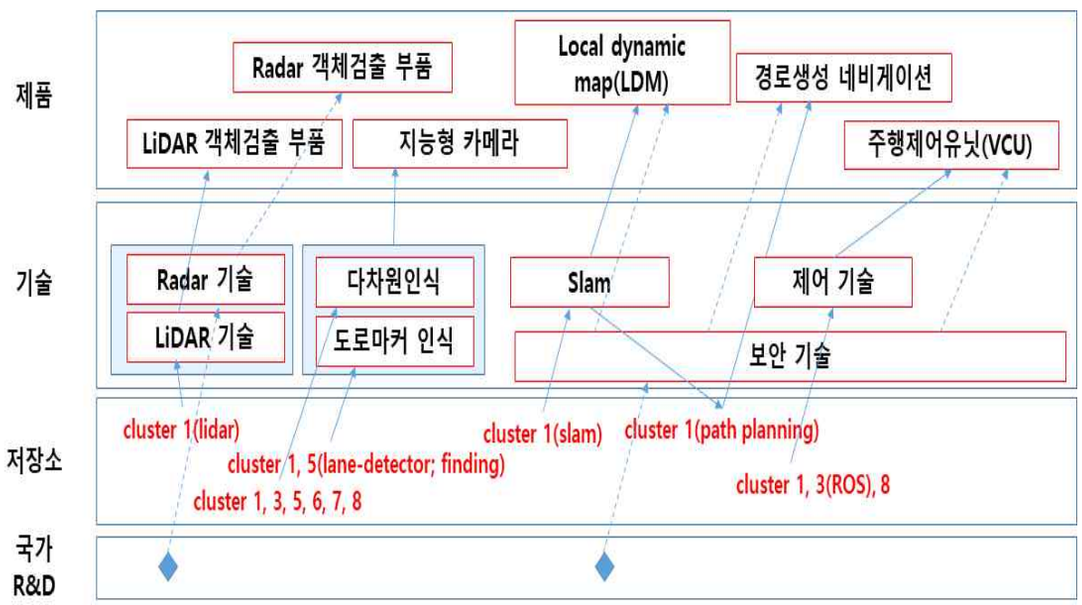
- 로드맵은 기존의 R&D-기술-제품으로 이어지는 로드맵 레이어 구성에서, R&D 부분을 오픈소스 저장소와 국가 연구소 투자로 구분하여, 개발의 신속성과 민간-정부의 기술투자 연계성을 확인할 수 있도록 구축할 수 있음

  - 실제로 지능형자율주행차의 주요 기술구성에 따라 저장소에서 유형화한 클러스터 간의 연계성을 바탕으로, 센서, 카메라, 매핑, 내비게이션 등 기술과 제품에 영향을 미칠 수 있는 근거를 확인할 수 있음
  - 현재 오픈소스에서 개발되고 있는 자율주행차의 주요 기술들의 수준을 확인하여 저장소에서 기술을 획득하는 대신, 레이더 또는 보안과 같이 자율주행차 오픈소스에서 개발이 더딘 부분에 대해 국가(또는 기업)가 R&D를 보완하도록 기획함
  - LiDAR 기술, 카메라 기반의 이미지 분석, 매핑 기술, 경로생성, 로봇제어 등 이미 많은 부분 깃허브에서 개발되고 있는 부분은 전문가 자문을 통해 오픈소스로 기술획득을 할 수 있는지 평가하고, 기술개발 사각지대에 기술투자를 하는 방향의 로드맵을

구성하여 제품개발 및 투자 효과성을 높일 필요가 있음

- 이와 같이 미래기술 대상의 R&D 추이를 오픈소스를 바탕으로 확인하여 기술 로드맵을 구축하고, 기술획득 방법을 의사결정할 수 있는 방향으로 활용 방안을 모색할 수 있으리라 기대됨

그림 3-6 오픈소스 기술 기반 자율주행차 기술-제품 로드맵



나. 메타버스 저장소 분석

◈ 메타버스 저장소 및 토픽 현황

- 메타버스 관련 깃허브 오픈소스 소프트웨어 개발현황을 살펴보기 위해, “metaverse”의 토픽을 가지는 저장소를 10개로 군집화함
  - 70개의 저장소가 가진 토픽을 바탕으로 k-means 클러스터링 알고리즘을 사용해 10개로 군집화하였음

※ 실제 100개 이상의 저장소가 검색되나 메타버스가 최근에야 개발되고 있으므로, 상대적으로 스타 수가 적어, 스타 수 1개 이상인 저장소를 대상으로 수행하였음

- 메타버스는 아직까지 대부분 디지털 자산 관련 암호화폐, NFT(Non Fungible Token: 대체불가능토큰), 블록체인 기술 저장소로 나타났으며, 메타버스를 수행할 AR/VR 등 기기와 관련한 토픽들도 주로 도출되었음

- NFT나 암호화폐는 메타버스 내에서 거래를 위한 안전한 자산으로 사용하기 위해 개발되고 있는 것으로 보이며, MS의 FIM(Forefront Identity Management) 등 사용자를 확인할 수 있는 정보보호 체계도 토픽으로 지닌 저장소가 개발되고 있음
- 그 외에도 저장소 수는 적지만, 공간정보 관련 토픽(geodistributedsystems)을 가진 저장소와 웨비나 등 비대면 원격 시스템(web, conference, webinar)에서의 메타버스 활용 가능성을 보여주는 저장소 등 서비스 측면의 개발사례도 확인되었음

표 3-12 메타버스 관련 깃허브 저장소의 군집 및 주요 토픽

군집	수	내용	주요 토픽
1	33	메타버스 일반	('metaverse', 33), ('microservice', 3), ('cloud', 2), ('metaverse-infrastructure', 2), ('r', 2), ('meta-analysis', 2), ('metaverse-cloud', 2), ('fim', 2), ('docker', 2), ('nft', 2), ('nfts', 2), ('react', 2), ('ethereum-dapp', 2), ('api', 1), ('micro', 1)
2	13	블록체인, 디지털 자산	('blockchain', 13), ('metaverse', 13), ('digital-assets', 2), ('digital-identity', 2), ('web', 2), ('bitcoin', 2), ('blocktrack', 2), ('c', 1), ('go', 1), ('rust', 1), ('golang', 1), ('crypto', 1), ('llvm', 1), ('smart-contracts', 1), ('hypercube', 1)
3	6	VR 관련 메타버스 서비스	('virtual-reality', 6), ('metaverse', 6), ('socialvr', 3), ('hazel', 2), ('mayaverse', 2), ('roleplay', 1), ('lsl', 1), ('opencollar', 1), ('lsl-scripts', 1), ('social', 1), ('unity', 1), ('vrchat', 1), ('virtualreality', 1), ('unity3d', 1), ('sandbox-game', 1)
4	5	암호화폐, NFT와 메타버스	('ethereum', 5), ('metaverse', 5), ('crypto', 1), ('erc721', 1), ('blockchain', 1), ('prettier', 1), ('polygon', 1), ('nft', 1), ('mona', 1), ('prettier-config', 1), ('monaverse', 1), ('bitcoin', 1), ('etp', 1), ('auction', 1), ('nfts', 1)
5	3	VR, AR 기반 서비스	('vr', 3), ('ar', 3), ('metaverse', 3), ('xr', 3), ('web', 3), ('linux', 1), ('open-source', 1), ('virtual-reality', 1), ('reality', 1), ('immersion', 1), ('3d', 1), ('game', 1), ('avatars', 1), ('vrm', 1), ('gltf', 1)
6	2	웹 VR	('vr', 2), ('aframe', 2), ('metaverse', 2), ('threejs', 1), ('webvr', 1)
7	2	오큘러스 VR	('vr', 2), ('metaverse', 2), ('3d', 2), ('game', 1), ('vive', 1), ('oculus', 1), ('oculus-rift', 1), ('virtual-reality', 1), ('virtual', 1), ('mit-license', 1), ('daydream', 1), ('janusvr', 1), ('oculus-quest', 1), ('sql', 1), ('php', 1)
8	2	VR, AR 기반 서비스	('web', 4), ('a-frame', 2), ('augmented-reality', 2), ('vr', 2), ('webvr', 2), ('ar', 2), ('aframe', 2), ('virtual-reality', 2), ('babylonjs', 2), ('metaverse', 2), ('javascript', 1), ('html', 1), ('threejs', 1), ('iframe', 1), ('3d', 1)
9	2	공간정보 시스템	('metaverse', 2), ('geodistributedsystems', 2), ('yomo', 2), ('yomo-use-case', 2), ('virtualhq', 2), ('metaverse-cloud', 2), ('macrometa', 1)
10	2	웨비나 메타버스	('web', 2), ('unity', 2), ('conference', 2), ('vr', 2), ('metaverse', 2), ('webinar', 2), ('xr', 2), ('oculus-quest', 2), ('desktop', 1), ('virtualreality', 1), ('virtual-reality', 1)

- 메타버스의 주요 개발자는 아직까지 개인 위주로 빅테크 기업보다는 개별적인 저장소를 개발하는 형태로 이루어지고 있음
  - 그러나 메타버스가 융합기술이라는 측면에서, 토픽을 메타버스로 가지고 있지 않더라도, 구글이나 MS 등에서 메타버스 관련 콘텐츠, 미디어, AR/VR 기술에서 메타버스를 다루고 있을 가능성도 있어, 트렌드 변화를 살펴볼 필요가 있음

표 3-13 메타버스 관련 깃허브 저장소의 주요 개발자

군집	수	주요 개발자
1	33	'micro/micro', 'yomorun/yomo', 'omigroup/OMI', 'rmetaverse/metaverse', 'micro-community/micro', 'srcnalt/avatar-view', 'sorengranfeldt/mre', 'woowa-techcamp-2021/store-6', 'jb55/protoverse', 'mvs-org/mvsd-mysql-sync', 'amirgamil/flora', 'SnowCrashDAO/metavoxel', 'datatogether/dataset_registries', 'r37616/FIMMV'
2	13	'hypercube-lab/hypercube', 'mvs-org/metaverse', 'mvs-org/lightwallet', 'mvs-org/metaverse-vm', 'canguruhh/metaversejs', 'mvs-org/mvs-live', 'mvs-org/new-frontiers', 'monaverse/eslint-config-monaverse', 'wakandalabs/worldofwakanda', 'ConscienceLand/coinwrapper-etp', 'blocktrack/worker', 'madjin/m3-panel', 'blocktrack/merkleizer'
3	6	'OpenCollar/opencollar', 'owlboy/greatpug-public', 'Vytek/MayaVerse', 'Vytek/MayaVerse03', 'CreoverseTeam/CreoversePublic', 'PapaSmurf/metaverse'
4	5	'contextart/nfte', 'monaverse/prettier-config-monaverse', 'mvs-org/mvs-coin-bridge', 'reneDescartess/bored-elon-unicorn-club', 'Digiworlds/frax-bucks'
5	3	'vircadia/vircadia', 'madjin/vrm-samples', 'metaversityfoundation/metaversityfoundation.github.io'
6	2	'gmaliandi/metavrse-toolbar', 'webvrse/webvrse'
7	2	'jbaicoianu/janusweb', 'ptsource/VRGrid
8	2	'exokitxr/exokit-web', 'exokitxr/exokit-browser'
9	2	'yomorun/yomo-vhq-backend', 'yomorun/yomo-vhq-nextjs
10	2	'Snowapril/VRoom', 'Snowapril/VRoomEditor'

◈ 메타버스 오픈소스의 제품 기여도 분석

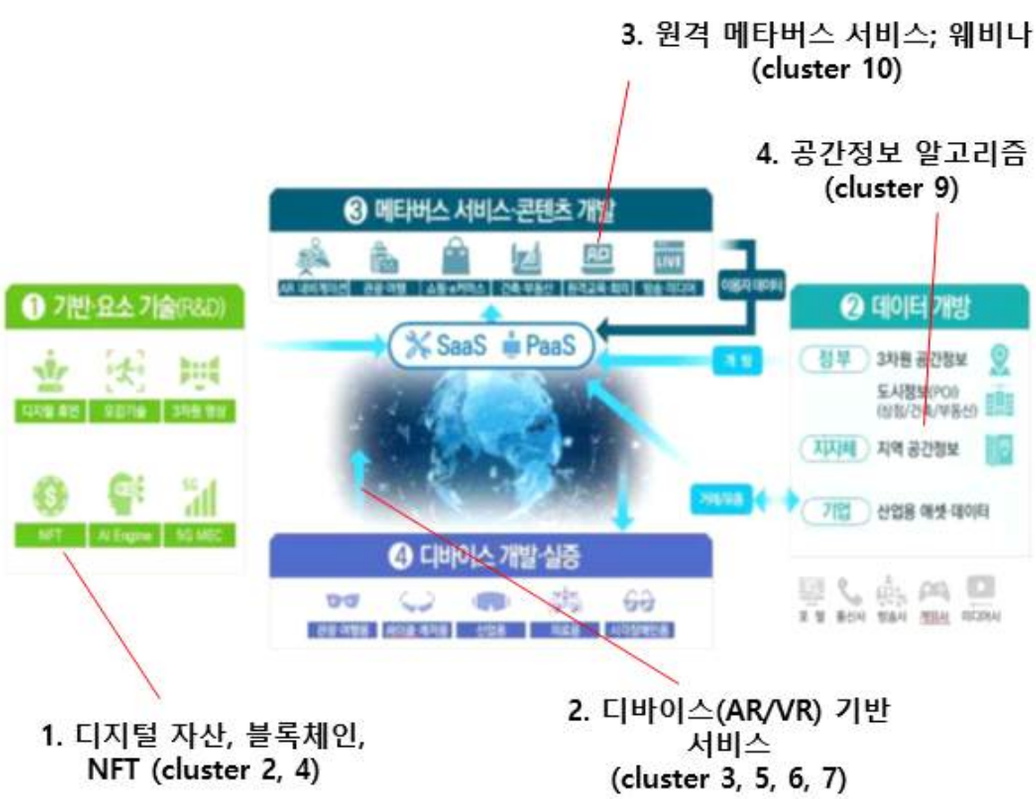
- 메타버스 기술은 국가 주요전략기술로서 ① 기반·요소 기술, ② 데이터 개방, ③ 서비스 및 콘텐츠 개발, ④ 디바이스 개발 및 실증으로 구분하여 실행계획로드맵을 수립(한국인터넷진흥원, 2021)하였음
  - 기반·요소 기술 : 디지털 휴먼, 오감기술, 3차원 영상, NFT, 인공지능, 5G
  - 데이터개방 : 정부·지자체·기업의 정보(3차원 공간정보, 도시정보, 지역정보, 산업용 데이터 등) 개발

- 서비스 및 콘텐츠 개발 : 내비게이션, 관광여행, 커머스, 건축부동산, 원격교육 및 회의, 방송미디어 등 메타버스 용도의 서비스 및 콘텐츠 개발
- 디바이스 개발 및 실증 : 메타버스 서비스 및 콘텐츠를 사용할 수 있도록 관광여행용, 레저용, 산업용, 의료용, 시각장애인용 등 현장에 적합한 맞춤화된 디바이스 개발

- 메타버스 오픈소스 분석에서도 보듯이, NFT, 디바이스, 원격교육 등은 이미 저장소 형태로 깃허브에서 시작되고 있으며, 이는 기반요소 기술, 서비스 및 콘텐츠, 디바이스 관련 기술에 대해 오픈소스가 참고자료가 될 수 있음

- 인공지능·기계학습 등 데이터 분석 분야나 디바이스 관련한 분야는 메타버스라는 토픽을 포함하지 않은 저장소에서도 개발될 소지가 있어, 메타버스가 추후 깃허브의 주요 토픽으로 자리잡을 때까지는 분석의 한계점이 있음
  - 향후 메타버스 오픈소스의 발전방향을 살펴보고, 오픈소스 저장소 측면에서 해결할 기술 획득 분야와 국가 또는 기업이 전략적으로 투자해야 할 기술을 구분하여 메타버스 기술 로드맵을 구성해야 할 것임

그림 3-7 메타버스 기술구조도 및 오픈소스 활용 가능성



※ 이미지 출처: 한국인터넷진흥원(2021)

다. 미래기술 저장소 분석 소결

◈ 미래기술 대상에 대한 기반 기술 확인 용이

- 미래기술 대상에 따라 개발되고 있는 기반 기술을 확인하기에 용이하며, 개별적으로 개발되고 있는 기술 저장소를 종합하여 제품 및 서비스 개발의 로드맵을 구성하는 근거로 활용 가능함

- 현재 미래기술 대상에서 사용될 수 있는 기술을 확인하고, 필요한 기술들을 대상으로 발전계획을 수립할 수 있는 근거를 마련할 수 있음
  - 예를 들어, 자율주행차에서는 레이더 기술과 보안 기술이 부족하고, 메타버스에서는 현재 분야별 서비스 개발이 부족하다는 점을 확인할 수 있음

◈ 오픈소스 개발 수(數) 또는 수준에 따른 기술평가 가능

- 미래기술에 필요한 기술의 오픈소스의 수를 계량적으로 확인하고, 개발자 또는 코드 수준에 따른 오픈소스의 내용을 평가하여, 미래기술의 내용을 평가할 수 있음
  - 자율주행차와 메타버스의 두 가지 경우만 보더라도, 기술 트렌드 상 자율주행차의 기술 수준이 높다고 평가될 수 있음

◈ 국가와 민간이 주도해야 할 기술개발 영역을 기획하기 위해 활용 가능

- 민간에서 개발되고 있는 오픈소스 SW 기술을 확인하여, 기술을 민간 저장소에서 할지, 국가투자를 할지 등에 대한 의사결정의 근거로 사용 가능함

- 전문가 의견을 바탕으로 오픈소스로부터의 기술획득 가능성 등을 판단하여, 국가 또는 기업이 책임질 기술분야를 확인하는 것이 요구됨

Ⅳ 시사점 및 개선방향

가. 오픈소스 데이터 분석의 시사점

◆ 데이터 분석 가용성 탐색 및 확인

➡ 오픈소스 현황 파악을 위한 주요 데이터 탐색

○ 깃허브에서 제공하는 저장소의 기본 속성(스타수, 브랜치 수, 커밋 수, 개시일 등)과 저장소의 기술내용을 확인할 수 있는 토픽들을 중심으로 오픈소스의 개발현황을 살펴볼 수 있는 연구방법을 제안하였음

➡ 빅데이터 분석 가능성 확인

○ 상위 인지도 기준의 주요 저장소, 빅테크 기업의 저장소, 특정 기술의 저장소 관점에서 오픈소스의 기술통계와 유형화(군집화) 분석을 수행하여, 오픈소스 기술생태계를 이해할 수 있는 데이터 수집 및 분석 방법을 개발함

- 데이터 수집에서는 웹 페이지의 봇(bot) 방지를 회피하기 위한 동적 크롤링의 적용에서부터 웹 페이지 정보 속성과 API 정보 속성을 전처리할 수 있는 방법론을 개발하여 필요 시 기업 또는 기술 등 검색어 위주의 분석 코드를 제공함
- 대량의 오픈소스 저장소 데이터에서 기술내용을 요약적으로 정리할 수 있는 데이터로서 “토픽”에 초점을 맞추고, 토픽 위주로 상위 인지도 저장소, 기업 저장소, 기술 저장소의 기술내용을 차별적으로 비교할 수 있는 유형화 분석 방법을 제시함

◆ 오픈소스 주요 기술 확인 및 기술기획 가능성

○ (주요 저장소 기술 현황) 오픈소스의 주요 기술개발 형태가 산업이나 시장 위주가 아닌 깃허브의 태생적인 특성에 기반한 프로그래머 위주의 범용적 기술 단위의 개발이 이루어짐을 확인하였음

- 특히, 웹 기술인 front-end, back-end 기술과 관련한 저장소가 대다수였으며, 기업마다 자체 프레임워크 또는 플랫폼을 사용하여 저장소가 운영될 수 있도록 하는 등 웹 기술과 관련한 오픈소스 내 주도권을 확인함
- 구글과 MS 위주로 기계학습 관련 저장소가 많이 개발되고 있음을 확인하였으며, MS는 특히 자체 클라우드 기계학습 플랫폼인 Azure를 사용하여 많은 양의 저장소 개발 및 활용을 확보하였음

- (빅테크 기업 저장소 기술 현황) 기업의 특징에 따라 고객 방향의 서비스를 강조하는 front-end 저장소를 운영(페이스북, 애플 등)하거나 상거래 등의 서비스를 위한 서버나 데이터베이스 등 back-end 저장소를 주로 운영하는 기업(아마존, 알리바바) 등 기술주기에 따른 기업 생태계도 제시하였음
  - 하드웨어 기반의 기업(인텔, 삼성전자)들은 디바이스 최적화나 디바이스 연계 소프트웨어 등의 기술 저장소들이 개발되고 있는 등 기업 특징에 따른 저장소 운영 현황을 확인함
- (특정기술 저장소 기술 현황) 특정기술에 따른 기술유형을 살펴보고, 기술이 응용될 수 있는 제품과 비교해봤을 때, 오픈소스에서 개발된 대다수의 저장소들이 산업의 기술 및 제품 구조도와 높은 연관성을 보임을 확인하였음
  - 자율주행차의 경우, 물리적인 차량 기술 외에 IT 요소에서는 레이더와 보안 기술을 제외하고 센서·최적화·제어·시뮬레이팅 등 오픈소스 저장소에서 이미 많은 필요기술 부분을 구성하고 있었음
  - 메타버스의 경우, 아직 개발 초기라 주목을 받고 있지 않은 저장소로 확인되었으나, 메타버스 내에서 사용할 디지털 자산 관련 암호화폐(NFT) 및 블록체인 기술, AR/VR 등의 디바이스 기술, 원격교육 관련 메타버스 등 서비스와 콘텐츠 저장소가 태동하고 있음을 볼 수 있음
- 오픈소스의 기술개발현황을 바탕으로, 기업들의 현재 개발기술 목록 및 수준을 확인하여 기술협력을 제시하거나 오픈소스 저장소를 이용한 기술개발 기획을 수립하는 등 기술응용 관점에서 오픈소스를 바라볼 필요가 있음
  - 자율주행차 또는 메타버스 기술을 예로 들면, 현재 오픈소스에서 개발되고 있는 기술 현황과 수준을 살펴보아, 협력 및 공유 관계를 전략적으로 체결할 수 있음
  - 국가나 기업에서는 해당 제품을 새로 개발할 경우, 기술획득전략을 수립하는데 중요한 참고자료로 사용할 수 있으리라 기대됨
  - 그러나 아직까지 사설(private) 저장소에서 개발하고 있는 오픈소스 기술현황을 살펴보기가 어려우며, 특허를 받은 기술에 대해 오픈소스에 제시하는 등의 전략을 취하는 경우에는 오픈소스의 자연스런 기술획득이 어려울 수 있음
- 오픈소스의 기술획득전략을 바탕으로 신속한 기술로드맵 작성과 함께 오픈소스를 이용한 지식재산권 강화(표준특허와 같이) 등 다양한 기술기획전략을 고려해야 함
  - 자율주행차 로드맵 사례에서도 제안하였지만, 상용수준으로 사용되고 검증된 경우 오픈소스의 경우에는 신속히 기술획득을 처리하고, 부족한 부분의 기술역량에 대해 투자를 진행하는 등 기술로드맵 의사결정을 지원할 수 있음
  - 표준특허와 같이 특허를 받은 기술을 오픈소스로 제공하여, 활용과 교육을 증가시킨다면, 시장에서 경쟁우위를 확보하기 위해 도움이 되리라 기대됨

◈ 오픈소스의 기술 생태계 조성 방식 확인

- 오픈소스 저장소는 기술을 개발하면, 이를 개발자와 관계자가 활용/수정해나가며 사용자를 증가시키고, 활용과정에서 검증된 기술을 교육형태로 저장소를 운영해 프로그래머를 증가시킨 뒤, 증가한 프로그래머들이 다시 저장소를 증가시키는 나선형의 선순환 발전생태계 구조를 보임

- 이는 기술생태계가 "개발-활용/수정-검증-교육-개발-활용/수정-검증-교육"의 과정을 반복하며 오픈소스를 이해하고 선호하는 이해관계자를 확대하는 방향으로 진행되고 있음을 시사하고 있음

- 기업 간의 생태계는 협력 구조도 있지만, 기업 독자적인 프레임워크 및 플랫폼을 활성화시키며 저장소를 늘려가고 있으며, 기업의 제품 및 시장 특징에 맞춰 저장소가 범용적이거나 특화된 경우로 생태계가 구분되고 있음
  - 오픈소스라는 이름으로 협력 및 공유하여 지식의 확대 및 팽창을 달성하면서도, 그 확대의 중심을 독자적으로 개발한 프레임워크나 플랫폼으로 위치하여 경쟁우위를 선점하려는 기업의 시도도 엿보임

나. 한계점 및 개선방향

◈ 자율적 데이터 생성에 따른 비정형 분석의 한계점

- 깃허브 저장소에서 제공하는 속성과 토픽 등 다양한 정보를 활용하려고 시도하였지만, 오픈소스라는 자율적인 구축 특성에 따라 비정형적인 데이터의 분석 어려움은 태생적인 한계점으로 보임
  - 저장소의 정보소개 및 제공 수준이 다르며, 정보가 누락되거나 일관성이 없는 등 표준화되어 있지 않은 정보라 분석이 어렵고 결과의 신뢰성을 담보하지 못할 수 있음
  - 예를 들어, 저장소에서 토픽이 없는 경우가 많으며, 저장소를 설명하는 Readme 역시 텍스트, 이미지, 동영상 등 표준화되지 않은 형식으로 올라와 분석이 어려움
  - 깃허브 자체에서 너무나 많은 저장소 수 때문에 전수를 검색할 수 없어, 검색 위주의 분석만 가능하다는 한계점이 있음
  - 웹 페이지 제공정보와 API 제공정보의 유형이나 형식이 달라 범용적이고 전체적인 분석을 수행하기 어려움
  - 데이터의 본질적인 문제로, 기업이나 기술 등 대상을 명확하게 하여 저장소를 수집하여 탐색적 분석의 의미로 수행하고, 추가적인 정성적 조사가 뒷받침될 필요가 있음

◈ 기술 위주의 저장소 운영에 따른 해석의 한계점

- 깃허브는 프로그래머 위주의 기술개발 운영을 통해 저장소가 개발되고 있으나, 개발한 기술이 적용되거나 응용된 제품이나 산업, 시장을 제시하고 있지 않아 전체적인 개발 프로세스를 확인하고 해석하기 어려움

- 제품이나 산업 위주로 검색(자율주행차나 메타버스와 같이)하는 경우, 그에 적용된 저장소들이 검색되지만, 저장소 자체만으로 대표할 수 있는 기술특성을 살펴보기 어려운 구조를 가지고 있음
  - 본 연구에서는 이를 해결하기 위해, 저장소 작성자가 게시한 토픽들을 기준으로 분석하였지만, 앞서 언급했듯이 많은 저장소에서 토픽을 제시하지 않고 있으며, 토픽의 개수 역시 저장소마다 차이가 큰 한계점이 있음

- 텐서플로우, 케라스 등 유명한 저장소의 경우, 학계의 연구나 논문, 기업의 연구개발활동 등에 사용된다는 보고서나 기사는 있지만, 저장소 자체에서 실적으로 게시하지 않는 한, 오픈소스의 기술내용 분석에는 제약이 많음

- 이에 따라 오픈소스 기술에서 파급될 수 있는 산업과 시장효과를 직접적으로 판단할 수 없으며, 특정기술 사례 위주로 오픈소스 기술분석과 별도의 시장조사를 바탕으로 오픈소스 기술의 효과성을 살펴볼 수밖에 없으리라 고려됨

◈ 기술의 탐색적 확인 및 기술예측에 활용성 강화

- 저장소에서 포함하고 제공하고 있는 데이터 품질 한계점에도 불구하고, 상위 인지도 저장소나 특정기술 검색을 통한 저장소의 분석은 생태계의 기술내용이나 현황을 탐색하는 용도로 적합하다고 보임
  - 특히, 토픽에 달린 세부기술 수준만 확인하더라도, 비전문가가 해당 기술 생태계 분야의 주요 기술이나 최신 기술용어를 검토하는데 기여도가 크리라 기대됨
  - 실제, NFT, gRPC, slam, LiDAR 등 해당 분야의 기술용어 등을 탐색할 수 있음

- 오픈소스 저장소의 탐색적 분석을 통해 해당 분야의 기본현황 및 기술지식 기반을 검토한 뒤, 산업과 시장 또는 특허와 논문 등 다른 분야의 정보를 결합하여 기술예측을 하는 통찰의 근거로 활용하는 것이 적합하리라 기대됨
  - 자율주행차 또는 메타버스의 오픈소스 저장소에서 도출된 주요 기술이 해당 제품 및 산업의 기술 구조도에 적합하게 대응되는 사례분석에서도 보듯이, 오픈소스 저장소는 기반 기술 지식의 탐색적 용도로 활용하고 기술기획 및 예측에 응용하여 가치를 확대하는 것이 바람직해보임

참고문헌

국내자료

권영환 (2020), 글로벌 오픈소스(공개SW) 생태계와 주요국 정책, Issue Report, 소프트웨어정책연구소(SPRi).

김성민, 홍아름, 최새솔, 연승준 (2020), 오픈소스 4.0 – 협력과 경쟁을 위한 혁신의 도구–, 기술정책 이슈 2020-15, 한국전자통신연구원(ETRI).

김재원, 신광섭 (2020), 클러스터링 기반의 최적 차량 운행 계획 수립을 위한 비교연구, 한국빅데이터학회지, Vol.5, No.2, pp.155-180.

윤경수, 김봉섭 (2021), 자율주행 기술 및 평가 동향, 주간기술동향(9.15), 정보통신기획평가원(IITP).

이왕재, 이학연 (2020), 깃허브 오픈소스 프로젝트 데이터를 활용한 인공지능 기술 개발 동향 분석. 대한산업공학회지, Vol.46, No.5, pp.548-557.

이진휘 (2020), AI 기술동향과 오픈소스, 이슈리포트 2020-제3호, 정보통신산업진흥원(NIPA).

이현진 (2021), 인공지능과 기계학습 개요 및 산업응용, K 뉴딜산업 INSIGHT 보고서 -4, 한국수출입은행 해외경제연구소.

정보통신산업진흥원 (2020), 2020년 오픈소스 SW(OSS) 시장 동향 조사보고서, 과학기술정보통신부.

정지선, 김동성, 이홍주, 김종우 (2019), 텍스트 마이닝 기법을 활용한 인공지능 기술 개발 동향 분석 연구: 깃허브 상의 오픈 소스 소프트웨어 프로젝트를 대상으로. 지능정보연구, Vol.25, No.1, pp.1-19.

최무이, 최재운, 김정민, 전이슬 (2020), 2020년 소프트웨어산업 전망, Research Report, 소프트웨어정책연구소(SPRi).

한국인터넷진흥원 (2021), 위치정보 산업동향 보고서, Weekly Report 7월1호, KISA.

한국저작권위원회 (2012), 2012 소프트웨어 관리 가이드, 문화체육관광부.

국외자료

Chesbrough, H. (2003), "Open Innovation: The New Imperative for Creating and Profiting from Technology", Harvard Business Press.

Gartner (2019), "Technology Insight for Software Composition Analysis", Gartner.

Gong, Y., Gu, Feng, Chen, K., and Wang, F. (2020), "The architecture of micro-services and the separation of front-end and back-end applied in a campus information system", 2020 IEEE International Conference on Advances in Electrical Engineering and Computer Applications.

Synopsys (2021), "2021 Open Source Security and Risk Analysis Report", Synopsys.

웹사이트

Github 홈페이지: https://www.github.com/

IBM developer 홈페이지: https://developer.ibm.com/patterns/create-backend-for-frontend-application-architecture/

지능화 기술 생태계 분석을 위한 데이터 수집 및 가공

www.etri.re.kr

ETRI Electronics and Telecommunications Research Institute
34129 대전광역시 유성구 가정로 218
TEL.(042) 860-6114 FAX.(042) 860-6504