

신입 개발자 직무별 요구 기술 분석 프로젝트

2025-2 데이터시각화(001)



학과: 데이터사이언스학과

학번: 23011816

이름: 황유리

목차

1. 문제 인식	1
1.1. 연구 배경	
1.2. 연구 목적 및 기대 효과	
2. 분석 데이터	2
2.1. 데이터 소스 및 수집 대상	
2.2. 수집 방법 및 주요 항목	
2.3. 데이터 선정 이유	
3. 시각화 결과	3
3.1. 시각화 기준 및 범례	
3.2. 직무별 시장 규모 비교	
3.3. 직무별 핵심 요구 기술 분석	
3.4. 직무 간 기술 스택 교차 비교	
3.5. 기술 스택의 유형별 비중 분석	
4. 토론 및 결론	7
4.1. 분석 결과에 대한 심층 논의	
4.2. 연구의 한계 및 제언	
4.3. 결론 및 최종 제안: '데이터 기반 커리어 솔루션' 개발	
5. 부록 – 데이터 및 소스 코드	8

1. 문제 인식

1.1. 연구 배경

데이터사이언스학을 전공하며 데이터를 통해 문제를 해결하는 방법을 배우고 있지만, 정작 학생들의 가장 큰 고민인 진로 문제 앞에서는 데이터가 아닌 막연한 소문에 의존하는 경향이 있다. 백엔드, 프론트엔드 등 각 개발 직무가 실제 기업에서 어떤 기술 스택을 요구하는지에 대한 객관적인 정보가 부족하기 때문이다.

수백 개의 채용 공고는 그저 거대한 텍스트의 나열일 뿐, 그 안에서 직무별 차이를 한눈에 파악하기란 불가능하다. 본 프로젝트는 이처럼 복잡한 텍스트 데이터를 ‘데이터 시각화’를 통해 분석하여, 정보의 비대칭을 해소하고자 한다.

1.2. 연구 목적 및 기대 효과

정확한 데이터의 부재는 비효율적인 노력과 불안감을 야기하므로, 이를 해결할 경우 다음과 같은 효과를 기대할 수 있다.

첫째, 비효율적인 노력을 줄일 수 있다. 직무별 요구 기술을 명확히 모른 채 공부하면, 적성에 맞지 않는 분야에 소중한 시간을 낭비할 위험이 크다. 데이터 분석을 통해 각 분야의 핵심 요구 기술을 미리 파악한다면, 한정된 시간 속에서 최대의 효율을 낼 수 있도록 학습 계획을 전략적으로 세울 수 있다.

둘째, 합리적인 진로 설계가 가능하다. 각 직무의 요구 기술 스택을 데이터로 비교하면, 본인의 강점과 흥미가 어떤 분야에 더 적합한지를 객관적으로 파악할 수 있다. 시각화 된 자료는 복잡한 기술 관계도를 한눈에 보여주어, 텍스트만으로는 파악하기 힘든 직관적인 이해를 돋는다.

셋째, 교육 과정 개선의 근거를 마련할 수 있다. 만약 분석 결과, 우리 학과 커리큘럼이 실제 산업 현장의 요구와 차이가 크다는 점이 드러난다면, 이 데이터는 구체적인 수치를 바탕으로 학교 측에 개선을 요구하는 설득력 있는 자료가 될 것이다.

본 프로젝트는 흩어진 채용 정보를 유의미한 시각적 자료로 재구성하여, 데이터에 기반한 합리적인 진로 선택의 기준을 제시하는 것을 최종 목표로 한다.

2. 분석 데이터

앞서 제기한 문제의 해결을 위해 가장 객관적이고 강력한 재료인 '**실제 채용 공고 데이터**'를 수집 및 분석하였다.

2.1. 데이터 소스 및 수집 대상

데이터 소스는 국내 주요 IT 채용 플랫폼 중 하나인 '**원티드(Wanted)**'로 한정하였다. 분석의 정확성을 위해 '신입(경력 1년 미만)' 필터를 적용한 뒤, 원티드의 '개발' 직군 중 다음 3개 직무의 공고를 수집 대상으로 선정했다.

- 서버 개발자
- 프론트엔드 개발자
- 데이터 사이언티스트

2.2. 수집 방법 및 주요 항목

데이터 수집은 Python의 Selenium과 BeautifulSoup 라이브러리를 활용한 **웹 스크레이핑 (Web Scraping)** 기법을 사용했다. 특정 기간 게시된 공고의 상세 페이지에 접근하여 다음과 같은 항목을 수집했다.

- **핵심 분석 항목:** 기술 스택 (Technical Stack)
- **기타 항목:** 회사명, 공고 제목

2.3. 데이터 선정 이유

채용 공고 데이터를 선정한 이유는 이 데이터가 가진 세 가지 특성이 프로젝트 목표 달성에 가장 적합하기 때문이다.

- **객관성:** 개인의 경험담이 아닌 기업의 실제 요구사항을 반영한다.
- **정량적 비교 가능성:** 기술 키워드의 빈도수를 측정하여 직무별 특성을 수치로 비교 할 수 있다.
- **시의성:** 빠르게 변화하는 IT 시장의 최신 기술 트렌드를 담고 있다.

3. 시각화 결과

본 장에서는 웹 스크레이핑을 통해 수집된 원시 데이터(Raw Data)를 분석 목적에 맞게 정제하고, 이를 시각화하여 도출된 주요 인사이트를 기술한다. 먼저 분석의 정확도를 높이기 위해 텍스트 전처리 과정을 수행하였다. 구체적으로는 결측치가 포함된 공고를 제외하고, 'React.js'와 'React', 'Amazon Web Services'와 'AWS' 등 동일한 기술임에도 상이하게 표기된 키워드를 통일하는 **정규화(Normalization)** 작업을 거쳤다.

데이터 가공 및 집계에는 **Python의 Pandas** 라이브러리를 사용하였으며, 시각화 구현에는 **Matplotlib** 및 **Seaborn** 라이브러리를 활용하여 데이터의 분포와 관계를 직관적으로 표현하였다. 분석 결과는 거시적인 **시장 규모(3.2)** 비교를 시작으로, 미시적인 **직무별 핵심 기술(3.3)**, 그리고 직무 간 관계를 조망하는 **교차 비교(3.4)** 및 **기술 유형 분석(3.5)** 순으로 전개된다.

최종적으로 기술 스택 정보가 명확히 기재된 유효 데이터를 선별한 결과, **서버 개발자 44건**, **프론트엔드 개발자 46건**, **데이터 사이언티스트 29건**으로 총 119건을 확정하여 본 분석의 모집단으로 삼았다.

3.1. 시각화 기준 및 범례

본 보고서의 모든 시각화 자료는 데이터의 특성을 효과적으로 전달하기 위해 다음과 같은 기준을 적용하였다.

- **색상 기준 (Color):** 데이터의 성격에 따라 두 가지 색상 기준을 적용하였다.
 - **양적 정보 (그림 1~5):** 채용 공고 수나 기술 요구 비율과 같은 '수치'를 비교하는 그래프에서는, 값이 클수록 더 진하고 어두운 색으로 표현하여 중요도를 직관적으로 나타냈다.
 - **범주형 정보 (그림 6):** 기술의 '유형(Category)'을 구분해야 하는 그래프에서는 서로 다른 색상(Hue)을 사용하여 각 항목의 경계를 명확히 하였다.
- **격자선 (Grid):** 수치 비교를 돋기 위해 회색 점선의 격자선을 데이터 레이어 뒤쪽에 배치하여 데이터의 가독성을 해치지 않도록 하였다.

3.2. 직무별 시장 규모 비교

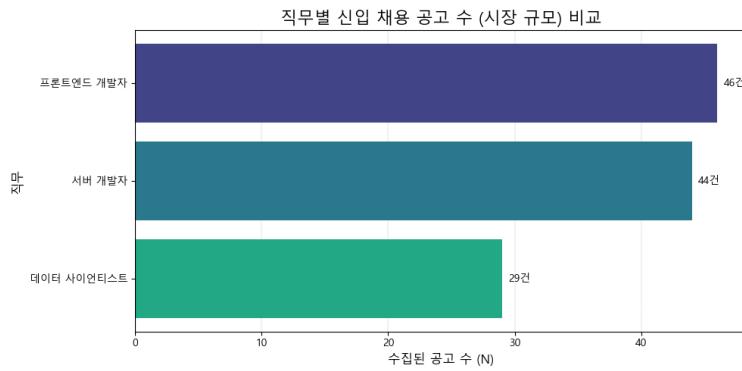


그림 1. 직무별 신입 채용 공고 수 (시장 규모) 비교

[그림 1]은 각 직무별로 수집된 공고의 개수(N)를 시각화한 것이다. 분석 결과 서버 개발자와 프론트엔드 개발자의 공고 수는 유사한 수준인 반면, 데이터 사이언티스트 직무의 신입 공고 수(29건)는 다른 두 직무에 비해 상대적으로 적음을 확인할 수 있다. 이는 데이터 관련 직무의 신입 진입 장벽이 상대적으로 높거나, 신입 시장의 규모 자체가 작음을 시사한다.

3.3. 직무별 핵심 요구 기술 분석

각 직무에서 어떤 기술 스택이 가장 중요하게 요구되는지 파악하기 위해, 기술별 요구 비율(%)을 막대그래프로 시각화 했다. 텍스트 레이블이 긴 범주형 데이터의 특성을 고려하여 가로형 차트를 채택하였다.

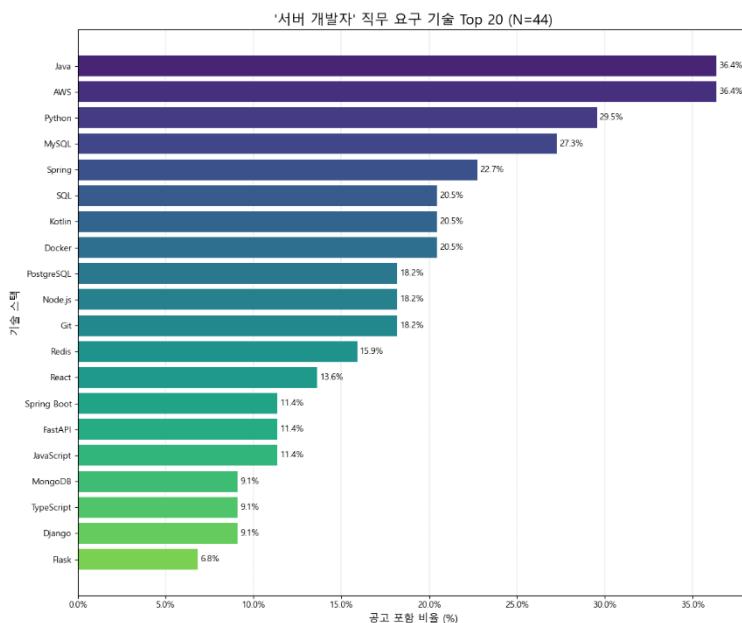


그림 2. '서버 개발자' 직무 요구 기술 Top 20 (N=44)

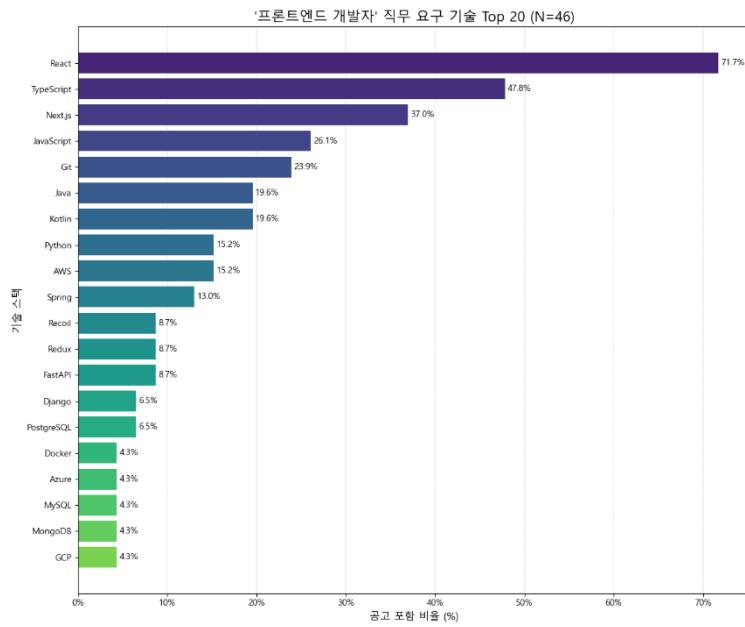


그림 3. '프론트엔드 개발자' 직무 요구 기술 Top 20 (N=46)

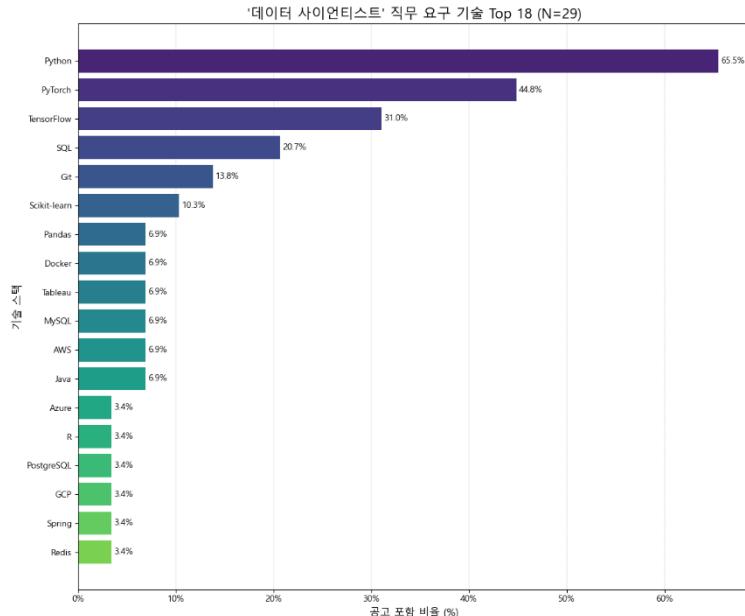


그림 4. '데이터 사이언티스트' 직무 요구 기술 Top 18 (N=29)

분석 결과, **서버 개발자(그림 2)**는 Java(36.4%)와 AWS(36.4%)가 가장 높은 비중을 차지했으며 Python, MySQL, Spring이 그 뒤를 이었다. 이는 국내 서버 개발 시장에서 Java/Spring 프레임워크 활용 능력과 클라우드(AWS) 및 DB 역량이 필수적임을 보여준다. **프론트엔드 개발자(그림 3)**는 React(71.7%)가 압도적인 1위를 기록해 사실상의 표준 기술로 자리 잡았음을 확인했으며, TypeScript(47.8%)의 중요성 또한 매우 높게 나타났다. 반면 **데이터 사이언티스트(그림 4)**는 Python(65.5%)이 압도적이었으며 PyTorch, TensorFlow 등 딥러닝 프레임워크가 상위권을 차지했다.

3.4. 직무 간 기술 스택 교차 비교

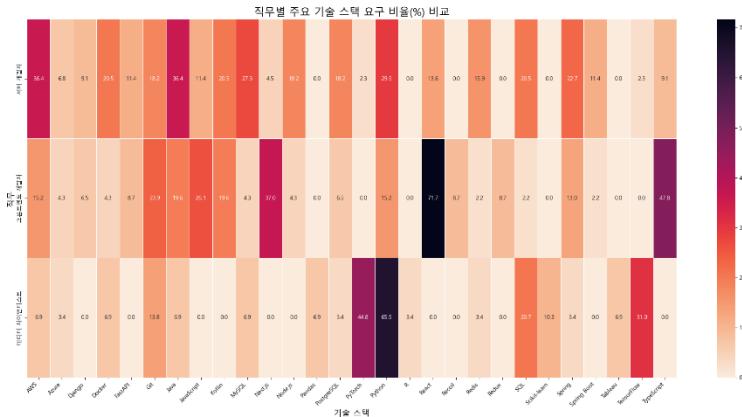


그림 5. 직무별 주요 기술 스택 요구 비율(%) 비교 히트맵

[그림 5] 분석 결과, 직무별 기술 생태계의 분리 현상이 뚜렷했다. **공유 기술** 측면에서는 Git과 SQL이 직무와 무관하게 전반적으로 요구(11~24%)되어 개발자의 기초 소양임을 증명했다. 반면 **특화 기술** 측면에서는 React/TypeScript(프론트), Python/PyTorch(DS), Java/Spring(서버)이 각 직무 행(Row)에서만 진한 색상으로 나타나, 직무 간 기술 장벽이 명확히 존재함을 시각적으로 확인하였다.

3.5. 기술 스택의 유형별 비중 분석

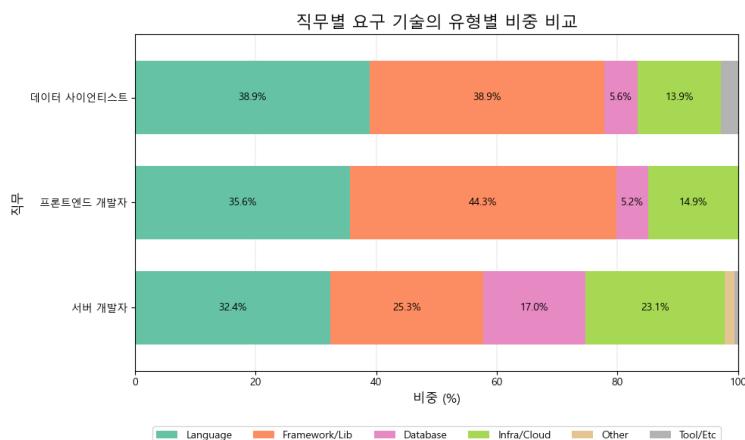


그림 6. 직무별 요구 기술의 유형별 비중 비교

[그림 6]의 100% 누적 막대그래프(Stacked Bar Chart)를 통해 직무별 기술 구조의 차이를 발견했다. **서버 개발자는** 인프라/클라우드 비중이 23.1%로 가장 높아 운영 능력(DevOps)의 중요성이 부각되었으며, **프론트엔드 개발자는** React 등 프레임워크/라이브러리 의존도가 44.3%로 압도적이었다. 반면 **데이터 사이언티스트는** 언어와 라이브러리가 각각 38.9%를 차지해, 특정 도구보다는 Python 언어 자체와 분석 패키지 활용 능력이 핵심임이 나타났다.

4. 토론 및 결론

4.1. 분석 결과에 대한 심층 논의

본 연구는 채용 공고 데이터를 시각화하여 직무별 기술 스택의 구조적 차이를 규명하였으며, 이를 통해 도출된 주요 시사점은 다음과 같다.

첫째, **신입 채용 시장에서의 ‘기술 파편화’와 ‘전문화’ 경향이 뚜렷하다.** [그림 5]의 히트맵 분석은 개발 직군 내에서도 직무 간 기술 호환성이 매우 낮음을 보여준다. 서버 개발자는 Java/Spring/AWS 중심의 생태계가, 프론트엔드 개발자는 React/TypeScript 중심의 생태계가 구축되어 있다. 이는 “언어 하나만 깊게 파라”는 식의 조언이 더 이상 유효하지 않음을 시사한다. 오히려 초기 진로 설정 단계에서부터 목표 직무를 명확히 하고, 해당 생태계의 표준 기술 조합을 패키지 형태로 학습하는 ‘타겟팅 전략’이 필수적이다.

둘째, **‘DevOps(개발 운영)’ 역량의 보편화 현상이다.** [그림 2]와 [그림 6]에서 서버 개발자의 AWS(36.4%)와 Docker(20.5%) 요구 비율이 상위권에 랭크된 점은, 과거 인프라 엔지니어의 영역이었던 배포와 운영 업무가 애플리케이션 개발자에게 이관되고 있음을 의미한다. 기업은 코드를 작성하는 ‘코더’가 아니라, 서비스를 클라우드에 배포하고 운영할 수 있는 ‘엔지니어’를 원하고 있다. 따라서 학생들은 비즈니스 로직 구현을 넘어 CI/CD 및 가상화 기술을 필수적으로 습득해야 한다.

셋째, **데이터 사이언스 직무의 ‘높은 진입 장벽’과 ‘R&D 중심의 기술 요구’이다.** [그림 1]에서 확인된 데이터 사이언티스트의 적은 공고 수(N=29)와 [그림 4]의 PyTorch(44.8%), TensorFlow(31.0%)의 높은 비중은, 해당 직무가 단순 분석을 넘어 딥러닝 모델링이 가능한 석사 이상의 역량을 요구함을 방증한다. 이는 학부 커리큘럼 외에 별도의 심화 프로젝트나 대학원 진학 등 고도화된 전문성 확보 전략이 필요함을 시사한다.

4.2. 연구의 한계 및 제언

본 연구는 유의미한 시사점을 도출했으나 다음과 같은 한계를 가지며, 이를 보완하기 위한 후속 연구를 제언한다.

첫째, **데이터 수집의 편향성이다.** 본 분석은 ‘원티드’ 플랫폼에 국한되어 스타트업과 IT 기업 위주의 트렌드만을 반영했다는 한계가 있다. 향후 ‘사람인’, ‘링크드인’ 등으로 데이터 소스를 확장하고 분기별 시계열 분석을 수행한다면, 보다 일반화된 결론을 얻을 수 있을 것이다.

둘째, **분석 기법의 한계이다.** 현재의 빈도 분석은 기술 키워드의 단순 등장 횟수만 집계하여, ‘필수 자격’과 ‘우대 사항’의 문맥적 중요도를 구분하지 못했다. 향후 자연어 처리(NLP)를 도입하여 중요도를 가중치로 반영하고, 연관 규칙 분석을 통해 함께 쓰이는 기술 조합을 도출한다면 분석의 정교함을 높일 수 있을 것이다.

4.3. 결론 및 최종 제안: '데이터 기반 커리어 솔루션' 개발

본 프로젝트를 통해 확인한 가장 큰 문제는 학생들의 학습 방향과 기업의 요구 사항 간에 존재하는 '정보의 비대칭'이다. 정적인 보고서는 현 시점의 현황을 보여줄 뿐, 매일 변하는 채용 시장의 흐름을 실시간으로 반영하지 못한다. 이에 본 연구의 분석 파이프라인을 발전시켜, <예비 개발자를 위한 맞춤형 기술 갭(Gap) 분석 및 학습 로드맵 추천 서비스> 개발을 최종적으로 제안한다.

이 서비스는 다음과 같은 가치를 제공해야 한다.

1. **개인화된 역량 진단:** 사용자가 자신이 보유한 스택을 입력하면, 본 연구에서 구축된 직무별 데이터베이스와 대조하여 시장 경쟁력을 '매칭 점수'로 수치화해 보여준다.
2. **데이터 기반 학습 가이드:** 목표 직무의 상위 20% 공고에서 공통적으로 요구하지만 사용자에게 없는 '미싱 스킬(Missing Skill)'을 우선순위대로 시각화 하여, 무엇을 먼저 공부해야 할지 명확한 가이드를 제공한다.
3. **트렌드 모니터링 (Monitoring):** "요즘 뜨는 기술이 무엇인가요?"라는 질문에 감이 아닌 데이터로 답한다. 주요 기술의 채용 공고 수 증감 추이를 시계열 그래프로 제공하여, 학생들이 기술 학습의 투자 가치를 판단하는 지표로 삼게 한다.

결론적으로, 방대한 텍스트 형태의 채용 공고를 **데이터 시각화를 통해 유의미한 정보로 구조화하는 과정은** 막막한 취업 준비생들에게 '지도'와 '나침반'을 줘여주는 일과 같다. 본 제안이 구현된다면, 학생들은 불확실한 미래에 대한 불안감을 덜고, 보다 효율적이고 전략적으로 자신의 커리어를 주도해 나갈 수 있을 것이다.

5. 부록 – 데이터 및 소스 코드

- **데이터 출처:** '원티드 (Wanted)' 채용 공고
- **GitHub 저장소:** <https://github.com/tangerineYR/dev-skill-analyzer-project.git> (전체 소스코드 및 원본 데이터 포함)