



인공지능 잘 활용하기

임재영

MLOps Engineer, NHN AD

MLOps Engineer

Jaeyoung Lim

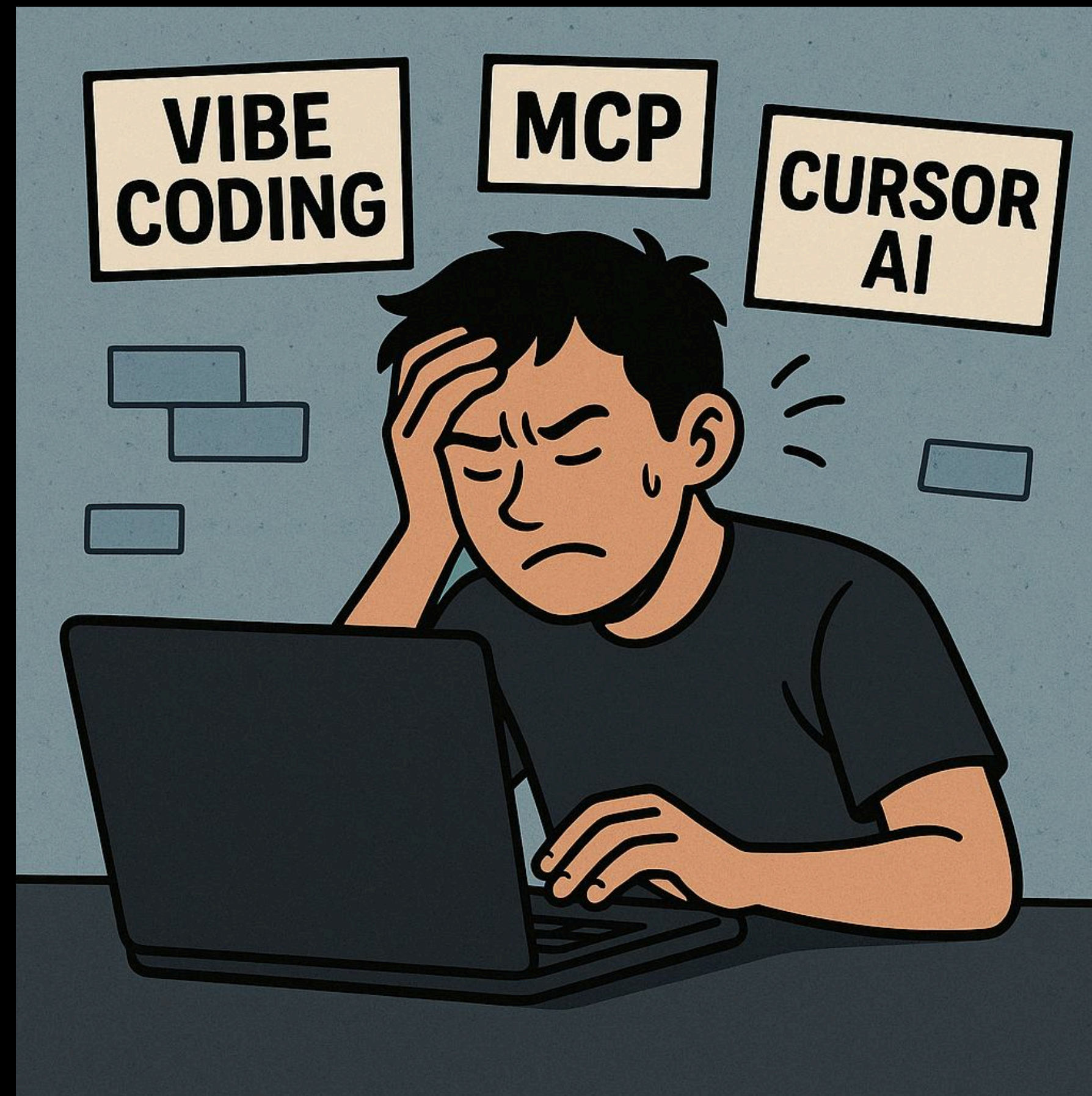
경력 및 학력

- NHN AD | 2024.08 ~ 현재
AI/ML 기반 Ad-Tech 플랫폼 개발
- 상명대학교 서울캠퍼스 | 2019.03 ~ 2025.02
휴먼지능정보공학전공 졸업 (컴퓨터과학 부전공)

교내 이력

- SMUMC | 2023.03 ~ 2024.02
5기 Server 파트장
- 이니로 | 2019.03 ~ 2020.12
임원진 부회장





자고 일어나면 새로운 기술들이 쏟아져 나오는 세상
모든 기술을 섭렵하며 반드시 따라가야 할까요?



대환장 갤러리
@bighwanjang1

요즘 직장인들이 회사에서 제일 많이 듣는 말



2024년 06월 02일 · 5:13 오후 · 조회수 114천회

952 재게시물 108 인용 2,085 마음에 들어요

우리 사장님 아님

인공지능을 잘 활용하는 방법



너, 내 동료가 되어라



헛소리 하지마 임마

AI 시대, 코더는 망했다? 개발자에게 필요한 새로운 역량

단순한 코더에서 문제 해결사로

- 인공지능이 단순 코딩 작업을 더 빠르고 정확하게(?) 처리할 수 있음
- 단순한 클론 코딩보다, 실제 문제를 해결한 프로젝트 경험이 더 높은 가치를 지님

인공지능과 함께 빠르고 효율적인 협업

- 협업을 위한 팀원이라 생각하고, 부족한 부분을 인공지능으로 빠르게 보완하자
- 인공지능을 디자인, 성능 최적화, 코드 리뷰 등 다양한 작업을 자동화하여 생산성을 높이자

능력 확장을 통해 얻는 새로운 시각

- 분야를 한정하지 말자 → “나는 백엔드 개발자야! Typescript 잘 몰라!”
- 자신의 영역을 넘어서는 다양한 분야에서 인공지능을 적극적으로 활용하고 인사이트를 얻으며 시각을 넓히자



공격수라면, 골을 넣는 것에 집중하면 된다!
하지만 우리 팀 진영으로 백패스가 날라온다면?

환각에 빠지지 않는 방법

AI 환각(Hallucination)

- 인공지능이 실제로 존재하지 않는 정보나 코드를 그럴듯하게 생성하는 현상
- 존재하지 않는 함수나 라이브러리를 사용하는 코드를 생성하는 경우

존재하지 않는 패키지과 함수

```
from financial_analytics import calculate_roi

roi = calculate_roi(investment=1000, revenue=1500)
print("ROI:", roi)
```

- financial_analytics 패키지는 실제로 존재하지 않음
- 개발자의 문제 해결이 지연될 수 있음

비효율적인 코드

```
def sort_numbers(nums):
    for i in range(len(nums)):
        for j in range(i + 1, len(nums)):
            if nums[i] > nums[j]:
                nums[i], nums[j] = nums[j], nums[i]
    return nums
```

- 얼핏보면 버블 정렬 같아 보인다
- 웬다구나! 하고 저 코드를 Production 환경에 넣어버린다면?

스트롭스트롭은 특히 AI 기반 코드 생성 도구들이 **생각 없이 코딩하는 습관**을 확산시키고 있다고 지적했다.

그는 AI는 **이미 존재하는 코드를 학습하고 재조합하는 데 특화**돼 있다며

그 결과 **과거의 잘못된 패턴, 낡은 스타일이 그대로 반복**될 가능성이 크다고 밝혔다.

(중략)

이에 그는 AI가 제공하는 '정답처럼 보이는 코드'에 의존하는 개발자들이 늘어날수록
프로그래밍의 본질인 **문제 인식과 추상화 설계 능력은 퇴화**할 수밖에 없다고 경고했다.

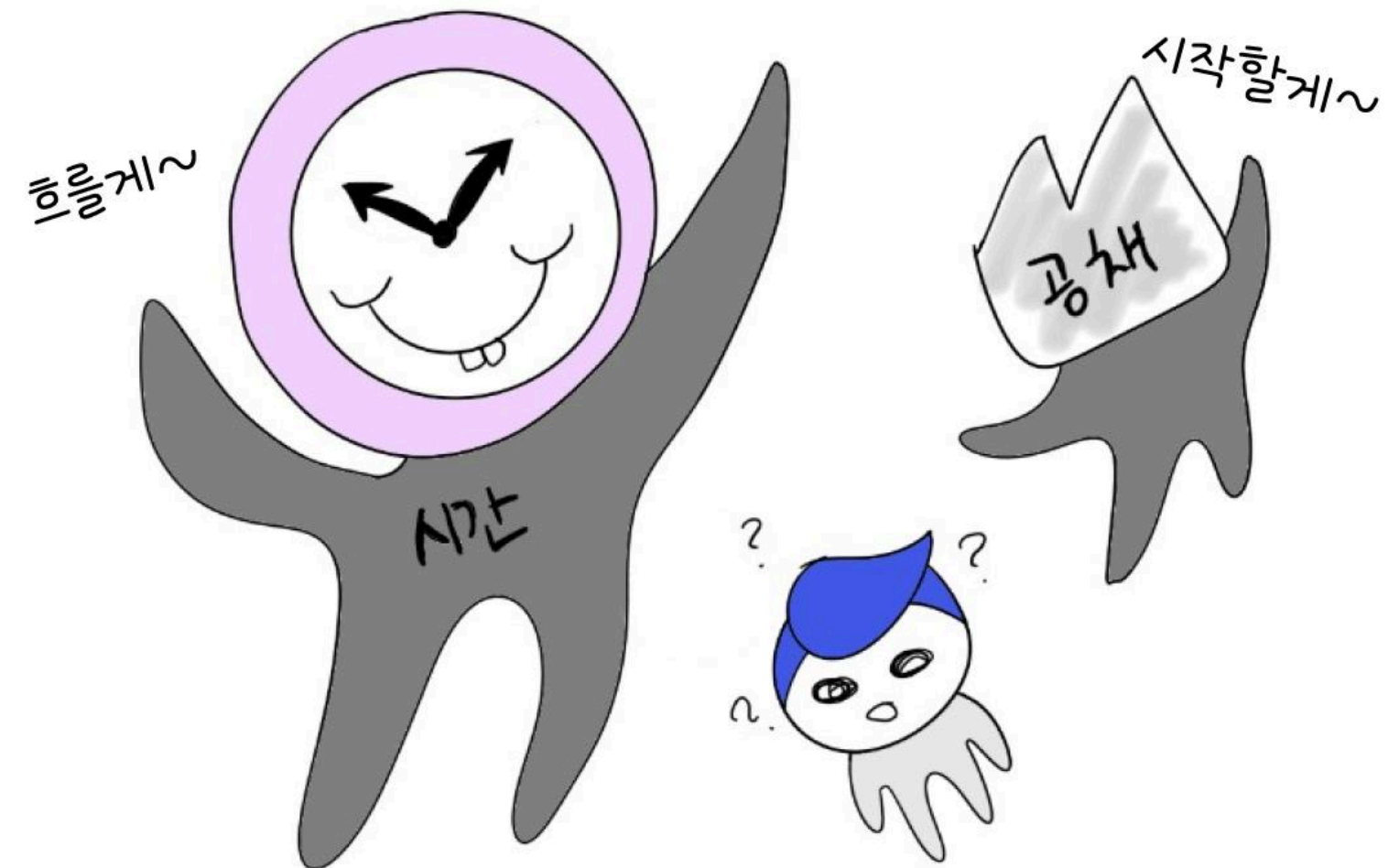
더불어 그는 **좋은 개발자는 문제를 정의하고 그에 맞는 모델을 구성할 수 있어야 한다**며

코딩은 그 결과물에 불과하다며 문제 인식 자체를 AI에 위임하는 순간 개발자는 더 이상 기술자가 아니라고 단언했다.

C++ 창시자 “바꿔야 할 건 언어가 아니라 개발자의 사고 방식 中

LEtwin

흥청망청 놀던 대학생살이 지나고...
저에게도 어느덧
슬슬 취업준비의 시간이 왔습니다



조금이나마 도움이 되길 바라는 마음에서!

포트폴리오 꾸준히 정리하기

특히 고학년 학생들에게

- “이제 본격적으로 만들어볼까?” 하면 막상 정리할 내용이 너무 많고 방향도 안 잡힘
- 원하는 회사 공고가 떴을 때 급하게 시작하면 부족한 시간에 쫓겨 퀄리티도 보장할 수 없음

정리는 습관처럼!

- 경험을 쌓을 때마다 바로바로 기록하기 (소중한 나의 경험 날려버리면 너무 아까움)
- 자기소개서도 기업마다 큰 질문은 비슷하기 때문에 계속 써보는 연습이 중요

도구 추천

- Notion : 커스터마이징 좋아하는 사람에게 추천
- 랠릿 : 간편하게 정리하고 싶은 사람에게 추천

팀 프로젝트 & 해커톤 경험 많이 쌓기

프로젝트로 실전 경험 쌓기

- 협업 경험은 많을 수록 좋고, 현업과 크게 차이 없음
- 저학년일수록 다양한 프로젝트 경험, 고학년 때는 하나의 프로젝트를 깊이 디벨롭해보는 경험 추천
처음부터 설계하는 것 vs 기존 코드를 리팩토링하는 것 → 각 과정에서 배우는 점이 다르기 때문
- 운이 좋다면 데모데이 때 만난 팀원들과 계속 함께할 수 있지만, 없다면 여러 커뮤니티 활용 가능

해커톤 유관력 기르기

- 사전 준비는 전략! (간단한 CRUD 코드, 기초 인프라 미리 준비)
쿠버네티스 사전 학습해서 가장 먼저 서버 배포한 경험 → FE 파트와 협업 효율 극대화한 경험
- 팀을 이끄는 포지션을 맡아보자! (파트리더 역할 추천)
리딩 경험은 큰 경쟁력이자 차별점 → 수동적인 태도보다 적극적인 태도로 임하면 얻는 것들 많아짐

전공 수업 열심히 듣기

전공 지식은 기본 체급

- 수업에서 듣는 내용은 현업에서도 많이 사용
 - 관심 있는 수업이 있다면 들어보기 (물론 수강신청 때 잡아야 말이지만.. 🤖)
 - “전공 수업 내용을 실제로 프로젝트나 해커톤에 적용했다” → 학부생 입장에서 매력적인 전략
- 분산시스템 듣고 MSA 구조에 관심이 있어 프로젝트에 적용
- 고급객체지향프로그래밍 수업을 듣고 디자인 패턴을 사용하여 코드 구조를 최적화한 경험

미래를 위한 현재의 집중

- 따로 시간 내서 정리하기에는 귀찮기도 하고, 정리해도 잘 안 보게 됨
 - 그때그때 열심히 듣고, 과제 있으면 열심히 하고, 트러블슈팅 있으면 정리만 해놓기
- 나중에 희미하게라도 남아 있는 기억을 면접 준비할 때 활용하자!

인공지능을 활용할 수 있는 시야를 넓히기
미래 채용 시장에서는 'AI 활용 능력' 도 중요하게 볼 것