

TRIP REPORT

ICSE 2023

23.05.13 ~ 23.05.21 MELBOURNE, AUSTRALIA

ICSE 2023

긴 겨울이 지나고 꽃이 피기 시작했다. ICSE 2023 또한 호주에서 화려하게 피어났다. 내가 경험했던, 그리고 경험하게 될 학회들 중 가장 아름다운 학회였다고 장담할 수 있을 정도로 훌륭했다.

감동이 컸기에 느낀 점도 많았다. 특히, 용기를 많이 얻었다. 나보다 거대한 것을 마주하면 두렵기 마련이다. 그러나, SE 라는 단 하나의 공통 관심사 아래 모두 동일 선상에 서게 된다. 이것이 얼마나 큰 행운이며 소중한 경험인지 이번에 깨닫게 되었다.

논문 소개

IMPACT OF CODE LANGUAGE MODELS ON AUTOMATED PROGRAM REPAIR

Code Language Model (CodeT5, InCoder...)을 이용한 APR 에 대해 조사한 스터디 논문이었다. 인상 깊었던 점은 직관과는 전혀 다른 발견을 한 것이다.

Finding 2: While buggy lines (code lines that need to be modified) are useful to guide fixing, CLMs fail to make good use of them and fix fewer bugs when the buggy lines are explicitly given.

비교 실험은 다음과 같았다. CLM 에게 buggy line 을 주석으로 주었을 때 Repair 성과와 아예 지운 상태로 주었을 때 Repair 성능을 비교했다. 보통 정보가 많으면 많을수록 더 좋은 성능을 내는 것이 옳다. buggy line 정보가 추가적으로 있으면 당연히 더 성능이 좋을 것이라고 예상된다. 그러나, Finding 2 에서는 직관과는 반대로 buggy line 을 지웠을 때 더 성능이 좋았다고 한다.

나는 이 발견과 관련하여 늘 갖고 있던 생각이 있었다. 해당 실험은 Defect4j 에서 이루어졌다. 당연히 CLM 은 학습데이터로 Github 프로젝트들을 포함했을 것이다. 그렇다면, 당연히 Defect4j 의 데이터 또한 학습 데이터에 포함이 되었을 것이다. 이 상황에서 Defect4j

데이터를 준다면, 학습 모델은 해당 데이터를 외워서 정답을 낼 수도 있다.

이러면 Finding 2가 다음과 같이 설명이 된다. Buggy line 을 주석으로 주면 학습했던 데이터와 달라지기 때문에 성능이 떨어진다. 반면에, Buggy line 을 지워서 주면 학습된 정답 데이터에서 정답코드 부분만 없는 상태이므로 정답코드를 (외워서) 채워주기만 하면 된다.

하지만, 제대로 파보면서 낸 결론이 아니라 추측에 불과하기 때문에 이 점을 분명히 짚고 싶었다. 그러나, 영어 실력이 부족하기도 하고 지식이 전무한 상태에서 떠오르는 질문이어서 Q&A 에서 물어보지 않고 따로 만나서 물어보려고 했었다. 그런데, 발표장이 좁아서 사람들이 입구를 꽉 채울 정도로 너무 비좁았다. 발표장을 떠나면서 저자랑 엇갈렸는지 밖에서 저자를 만나지 못하였다. 아쉽지만 다음 번에도 기회가 있겠거니 하고 그냥 자리를 떴다.

때는 ICSE 에서 제공하는 디너파티, 기회는 약 4 시간만에 찾아왔다. 맥주와 와인을 마시면서 이리저리 돌아다니고 있었는데, 이 논문의 저자가 내 앞을 지나가는 것이었다. 알코올도 몸에 돌겠다, 냅다 어깨를 툭툭 치고 저자한테 말을 걸었다. 그리고 돌아온 대답은 예상 밖이었다.

자신들이 생각하기에는 주석을 주석으로 이해하지 못하고 프로그램의 일부로 간주하여 문제가 발생하는 것이라고 한다. 즉, CLM 은 주석 처리된 프로그램 코드를 주석으로 받아들이지 못하고 프로그램에 영향을 주는 코드라고 생각하는 것이다. (단순히 string 을 넣어놓은 document 주석은 이에 해당하지 않는다.) 학습 데이터를 외워서 풀었다는 관점은 부정을 했다. 듣기 실력이 부족하여 그 이유에 대해서는 이해를 잘 못하였지만, 일단 외워서 푸는 것은 아니라고 하였다. 지금 생각해 보아도, 주석 처리가 되었을 때 합성하는 코드를 보면 저자의 관점이 더 합리적인 것 같다.

TESTING DATABASE ENGINES VIA QUERY PLAN GUIDANCE

논문의 구성이 매우 탄탄하였다. 괜히 Distinguished Paper 를 받은 것이 아니라는 생각을 하였다. 구성은 다음과 같았다.

1. Study 과정과 결과
2. Study 결과 기반 알고리즘 제안
3. 평가 결과

Study 결과가 앞선 논문과 동일하게 직관과 다른 결과였다. 단순히 생각해 보면, DB 엔진에 취약점이 있으면 그 부분만 공격하면 버그를 많이 찾을 수 있을 것 같다. 또한, 쿼리가 복잡하면 복잡할수록 테스트에 도움이 될 것 같다. 그러나, Study 결과는 완전 반대였다.

Query plans of the queries in previously-found bugs vary significantly, as 57.28% of the query plans are unique.

Query plans of the queries in previously-found bugs are compact and simple, as the average number of operations in a query plan is only 2.59.

이전 버그들을 조사 했을 때, 약 57%의 쿼리(정확히는 쿼리 플랜)가 유니크 하였다. 즉, 비슷한 쿼리에서는 최대 2 개의 버그만 찾을 수 있다는 것이다. 따라서, 취약점을 찾았을 때(버그를 찾았을 때) 그 부분만 공격하는 것(버그를 유발하는 쿼리만 생성)은 효과적이지 않다는 것이다. 또한, 버그를 유발하는 쿼리는 평균 연산량 2.59 로 굉장히 간단하다.

좋은 study 결과가 있으니, 나조차도 논문을 보지 않아도 알고리즘은 너무 자연스레 머리 속에 떠올랐다. Test 생성 가이드를 처음 보는 쿼리플랜을 사용하도록 하면 되는 것이다. 해당 논문의 알고리즘 섹션은 한 장에 불과하다. 복잡하지 않아도 유니크한 쿼리플랜으로 가이드만 하면 되기 때문에 작은 규모의 간단한 알고리즘이었다. 그러나, 좋은 근거를 가지고 만든 알고리즘이기에 효과는 굉장했다.

이 논문의 전개 방식을 보고 자아성찰을 해보았다. 최근, 연구를 진행하면서 '이러면 되지 않을까'하며 통계를 내보지 않고 느낌만 가지고 엔지니어링을 시작한 경험이 많았던 것 같다. 물론, 수많은

케이스 스터디를 통해 결론을 내린 내 직관이 틀리다고 생각하진 않는다. 그러나, 이를 수치화해서 보는 것은 분명 다를 것이다. 당장 한국으로 돌아가면 내 직관의 통계를 내보고 싶다. 그렇다면, 내 연구를 처음 보는 분들도 바로 고개를 끄덕일 것이다. 마치 내가 이 논문을 보고 자연스레 이해하고 납득했던 것처럼 말이다.

학회 이야기

몰랐던 것들을 알아가다

나는 현재 대학원 3 학기를 진행 중이다. PL/SE 관련 용어나 지식들과 한창 배우고 있기에 모르는 것들이 굉장히 많았다. 특히, 내가 진행했던 APR 이 아니라면 더더욱 용어와 지식은 생소했다.

예를 들어, BIC(bug induced commit)이 그러한 예시였다. 유신 교수님 학생 분 중, 안가빈님의 발표를 듣던 무렵이었다. BIC 를 어떻게 더 효율적으로 찾을 것인가에 대한 내용이었다. 직관 중 하나로 fault localization score 가 높은 line 과 관련된 commit 이 BIC 일 확률이 높다는 것이었다. 나는 BIC 가 버그를 유발한 commit 을 찾는 문제라고 알고 있었지만, 적용 상황 자체를 오해하고 있었다. 나는 어떤 commit 을 올릴 때 해당 commit 이 예기치 못한 버그를 유발하는지 판단하는 문제인 줄 착각하고 있었다. 그래서 해당 직관을 전혀 이해하지 못한 채로 가우뚱 하며

논문도 찾아 보았다. 그러나, 이 직관이 당연하다는 듯이 쓰여 있어
혹시 놓친 점이 있는지 혼자 고민을 많이 하였다.

고민 점을 연구실 분들께 말하자, 문제 상황을 바로 정정해주었다.
과거에 있었던 commit 들 중 어떤 commit 이 문제가 되었는지 찾는
문제였던 것이다. 문제를 제대로 이해하자 FL score 직관은 바로
이해가 되었고, 논문에서도 당연하다는 듯이 써 있는 것이 납득
되었다.

또 다른 예로는, 사람에 관한 이야기이다. Repair with AI 세션에서
누군가가 AI Repair 성능 스터디 논문을 발표하고 있었는데, fault
localization 부분을 어물쩍 넘어가는 것이었다. 모두가 비슷한
생각을 했는지 Ahbik 이라는 분이 Q&A 시간에 정확히 이 부분을
짚었었다.

내가 깨우침을 얻었던 것은 숙소에서 저녁을 먹으며 연구실
분들이랑 이야기를 하고 있을 때였다. 이 때까지만 해도 몰랐었는데,
Ahbik 은 사실 Repair 분야의 대가였던 것이다. 질문 수준도
상당하고 이야기를 하는데 여유로움이 느껴졌었는데, 대단한
분이시니 그럴만 했다. 이 이야기를 듣고나니 FSE 2022 에서 발표할
때도 Ahbik 이 체어였음을 깨달았다. 이제 더 이상 Ahbik 이란
이름을 까먹지 않을 것이다.

연구실에서 PL/SE 거장들의 이름들이 종종 나올 때가 있다. 사실, 나는 어떻게 이 분들의 이름을 다 꿰고 있는 것인가 궁금했었다. 나는 그 해답을 이번 학회에서 얻을 수 있었다.

용기를 얻다

학회에 참석하여 여러 발표를 들으며 앞으로의 연구 진행에 대해 깨우친 점이 많았다. 이 뿐만 아니라, 리셉션이나 디너 파티를 통해 다양한 사람들과 교류를 하며 영어에 대한 큰 자신감을 얻게 되었다. 이 자신감은 내가 회화를 잘한다는데서 오는 것이 아니라, 내가 회화를 못해도 상대는 들어주려고 한다는데서 오는 자신감이었다.

이와 관련된 특히나 기억에 남는 일화가 있다. 싱가포르 대학교의 한 교수님은 나에게 먼저 오셔서 말을 거셨었다. 나는 당황하면서 “영어를 잘 못하는데 괜찮냐?”라고 했었다. 그러자, 그 교수님은 나에게 못해도 괜찮다며 자신은 끝까지 들어줄 준비가 되어있다고 하였다. 이 한마디가 나의 긴장을 모두 누그러뜨려 주었고, 손짓 발짓 다해가며 열심히 대화를 했었다.

이 일을 겪고나서 영어공부를 더 열심히 해야겠다는 생각이 들었다. 물론, 지금 실력에서 멈춰있어도 상대방은 나에게 관심이 있다면 들어줄 것이다. 그러나, 상대방이 나에게 관심을 표한만큼 나도 그만큼의 관심을 돌려주고 싶다. 그러기 위해서 영어공부를 해야겠다고 생각이 든 것이다.

이전까지는 영어를 못하는 것이 부끄러워서 공부를 하였었다.
그러나, 이런 마음가짐으로 시작하면 늘 '어차피 모국어가 아닌데,
나는 할 수 없어'로 끝나버렸다. 지금은 달라졌다. 이번에는말로
인생의 큰 짐을 덜 수 있지 않을까 생각이 든다.

마지막으로

여러모로 머리로 깨어나고 열심히 교류도 시도했던 뜻 깊은
학회였다. 다만, 아쉬웠던 점은 웬만한 발표들이 이해하기 어렵고
지루하게 구성되었던 것이다. 논문만 붙으면 끝이라는 생각을 가진
발표들이 정말 많았다. 그러나, 나는 교류를 시도해보면서 위와 같은
생각은 정말 편협한 사고라는 것을 알 수 있었다. 물론, 학회의 목표
중 내 논문 자랑하기도 있다. 그러나, 멋있게 자랑함으로써 모두와
교류하는 장을 만드는 것이 진정한 학회의 의의라고 생각한다.

마지막으로, ICSE 2023 에 참석할 기회를 만들어 준 연구실 분들과
교수님께 감사의 말씀 올립니다.