CodeBERTs를 활용한 코드 유사성 판단

2024.04.01

김태형, 정현기

Contents

1. 코드 유사성 판단

데이터 분석

- 두 개의 C++ 코드 쌍이 동일한 문제를 해결하는 코드인지 유사성을 판단하는 AI 알고리즘 개발
- 학습 데이터는 500개의 문제에 대한 c++ 코드임.
- 평가 데이터는 학습 데이터에 없는 다른 문제에 대한 코드 중에서 595,000개의 Pair 쌍으로 이루어진 데이터임.
- train_code 폴더를 기반으로 Negative Sampling 전략을 통해 Pair 쌍으로 이루어진 학습 데이터를 생성할 필요가 있음.

2. Preprocessing

Tokenizer

- Code Preprocessing
  - ① “#include”로 시작하는 행을 제거함.
  - ② 주석에 해당하는 행을 제거함. ex) /* Anotation Example … */ , // Anotation Example …
  - ③ 개행 문자를 제거함.
  - ④ 마지막으로 전처리 후 빈 행은 제거함.
- Tokenizer
  - Microsoft에서 다양한 프로그래밍 언어의 소스 코드를 기반으로 만든 Tokenizer를 사용함.
  - 모델에 따라 사전 학습된 tokenizer를 불러옴.
  - ① microsoft/graphcodebert-base
  - ② microsoft/unixcoder-base

2. Preprocessing

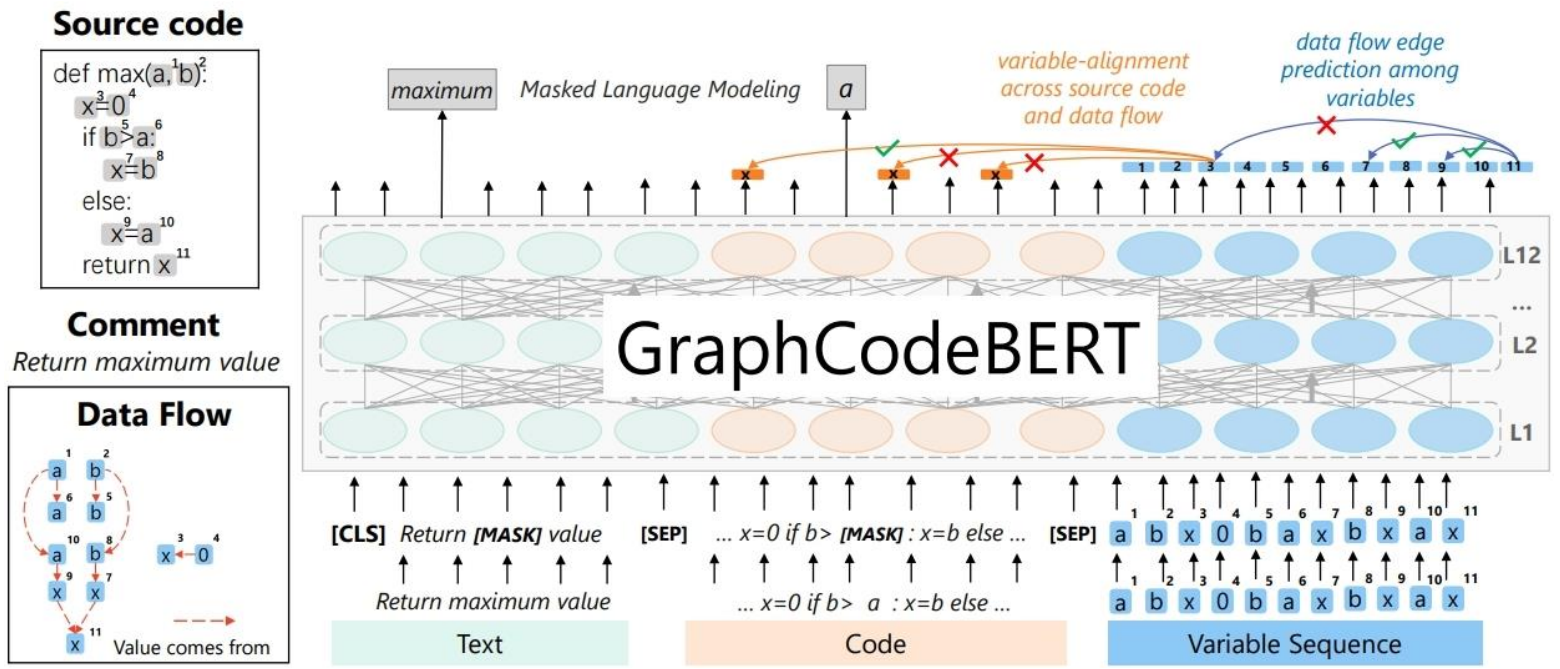
Negative Sampling

- BM25
  - 소스코드에 대해 훈련된 BM25는 입력한 쿼리와 소스코드 사이의 관련성을 평가하는 데 사용됨.
  - 즉, 입력으로 sort라는 쿼리가 들어갔을 때, "쿼리에 대해서 해당 소스코드가 내가 찾던 것과 얼마나 관련이 있는지"를 Score로 알려줌.
- Negative Sampling
  - Problem 번호를 기준으로 Positive Negative Pair를 생성함.

Code 1	Code 2	Similar
[#define, int, main, ~]	[int, while, k<10, ~]	1
[#define, if, b>a, ~]	[k = 2, k+t, return, ~]	0
…	…	…
[<iostream>, int, main, ~]	[std, cout, ~]	1

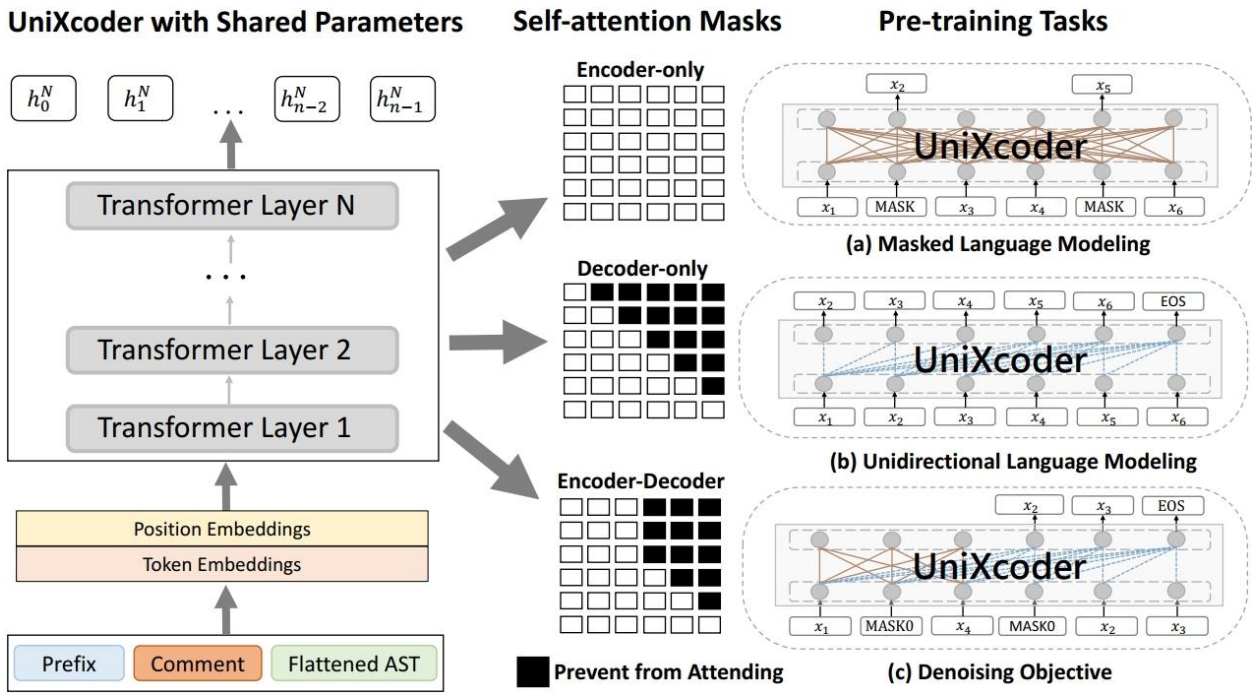
< 전처리 후 훈련 데이터셋 >

GraphCodeBERT Architecture



- GraphCodeBERT는 Data Flow를 사용하는 Transformer 기반의 프로그래밍 언어 사전 훈련 모델임.
- Graph-guided Masked Attention을 활용하여 code의 semantic-level structure를 학습함.
- MLM, code structure의 Edge Prediction, source code와 code structure 사이에 Node Alignment를 통해 사전 훈련을 진행함.
- Layer : 12, Hidden dimension : 768, Attention head : 12, Max sequence length : 512
- CodeSearchNet 데이터셋을 통해 사전 훈련되었으며, 이는 여섯 가지 프로그래밍 언어에 대해 문서 쌍을 포함한 2.3백만 개의 함수를 포함하고 있음.

UniXcoder Architecture



- UniXcoder는 Comment와 Abstract Syntax Tree(AST)의 cross-modal을 사용하는 Transformer 기반의 프로그래밍 언어 사전 훈련 모델임.
- AST의 구조적 정보를 유지하며 sequence 구조로 변환하는 one-to-one mapping function을 제안함.
- MLM, Unidirectional Language Modeling(ULM), Denoising Objective DeNoiSing(DNS)를 통해 사전 훈련을 진행함.
- Layer : 12, Hidden dimension : 768, Attention head : 12, Max sequence length : 1024
- C4 dataset 500만 개, unimodal CodeSearchNet 200만 개, multi-modal CodeSearchNet 100만 개의 데이터를 통해 사전 훈련됨.

5. 실험 결과 및 결론

Hyperparameter Setting

- 하이퍼파라미터는 다음과 같음.
- truncation_side : 'left'
- bm25 : 'bm25plus'
- GraphCodeBERT text_len : 512, UniXCoder text_len : 1024
- optimizer : 'adamw', learning_rate : 0.00003
- frac은 Negative Sampling 후 학습 데이터로 사용되는 random sampling의 비율로 0.01은 약 1M 개의 데이터임.

Code Pretraining Models	Private Accuracy	Public Accuracy
GraphCodeBERT (frac=0.01)	0.98859	0.98831
GraphCodeBERT (frac=0.02)	0.98909	0.98892
UniXcoder (frac=0.01)	0.98942	0.98911
GraphCodeBERT (frac=0.02) + UniXcoder (frac=0.01)	0.99111	0.99084

Thank you ☺

taehyeong93@korea.ac.kr