PROJECT.2

02

석사 연구 2

ALSI-TRANSFORMER: TRANSFORMER-BASED CODE COMMENT GENERATION WITH ALIGNED LEXICAL AND SYNTACTIC INFORMATION

2023 IEEE Access

ABOUT PROJECT

정부 과제에 참여하면서 진행한 연구입니다.

2023 IEEE Access

Abstract

Title: ALSI-Transformer: Transformer-Based Code Comment Generation with Aligned Lexical and Syntactic Information

Author: Youngmi Park, Ahjeong Park, Chulyun Kim

- 진행 기간: 2022년 3월 ~ 2022년 12월 / 참여 인원: 3명
- 정부과제명: 공정한 SW저작권 거래 및 유통 생태계 지원을 위한 저작권 응용 기술 개발
- 본인이 공헌한 점: 1차년도 실무 담당자, 관련연구 SOTA 모델 분석, 새로운 데이터 CAT 설계 및 구현, 소스코드와 CAT 데이터 정보 결합, 논문 진행

▶ [결과/성과]

1. 특허 출원

- 트랜스포머 기반의 자연어 주석 자동 생성 방법 및 장치
- 등록 심사중
- 출원일자: 2022년 12월 12일
- 발명자: 김철연, 박영미, 박아정

2. IEEE Access Accept

3. 멀티모달 태스크 동작원리 이해

4. 소스코드

- https://github.com/KIE-KID/ALSI-Transformer.git

ALSI-Transformer: Transformer-Based Code Comment Generation with Aligned Lexical and Syntactic Information

Youngmi Park, Ahjeong Park, Chulyun Kim

소프트웨어 프로젝트의 복잡성과 업데이트 빈도가 증가하면서 프로그램 이해의 중요성이 증가합니다. 이 때 좋은 주석은 프로그램 이해의 효율성 증가에 결정적인 역할을 합니다. 하지만 직접 주석을 작성하는 것은 시간이 많이 걸리고 품질을 보장하기 어려우며 기존 코드 주석은 관련 코드의 발전에 따라 계속해서 업데이트 돼야 합니다. **따라서 자동으로 고품질 주석을 생성하는 것은 매우 중요합니다.**

본 논문에서는 자동 주석 생성의 정확도를 향상시키기 위해 다음을 소개합니다. **Lexical 정보와 Syntactic 정보의 순서와 길이를 정렬하기 위한 새로운 구문 시퀀스인 CAT(Code-Aligned Type sequence)을 제안하고 그에 따른 신경망 모델인 ALSI-Transformer을 제안합니다.** ASLI-Transformer는 Transformer 기반 딥러닝 모델로, 함수 단위의 소스코드를 입력으로 넣었을 대 적절한 자연언어 주석을 출력으로 생성하는 것을 목표로 합니다. 특히, CAT과 Gate Network를 활용해 소스코드의 lexical, syntactic 정보를 결합합니다.

다양한 실험을 통해, 표준 기계 번역 메트릭을 사용하여 논문의 방법을 현재 Baselines와 비교했고 **이 방법이 코드 주석 생성에서 최첨단 성능**을 달성한다는 것을 보였습니다.

기여도

실험 설계	50
실험 진행	50
논문 작업	50

현재 상태

2023년 IEEE Access Accept

새로운 데이터 타입과 모델

기존 연구와 다른 방향으로 정제한 새로운 데이터 타입인 Code-Aligned Type sequence(CAT)을 이용한 ALSI-Transformer을 제안합니다.

맡은 파트

- 주석 생성 관련연구를 공부하고 SOTA 달성 모델을 분석
- 새로운 데이터 타입 CAT을 설계 및 구현
- Lexical, Syntatic 정보를 결합하기 위한 6가지 방법을 설계 및 실험하고 최종적으로 Gate Network의 성능이 좋음을 확인
- DeepCom, Hybrid-DeepCom, SeTransformer에 대한 비교실험을 주도
- 논문의 Introduction, Related Work, 본문에 모두 참여

ALSI-Transformer: Transformer-Based Code Comment Generation with Aligned Lexical and Syntactic Information

Youngmi Park, Ahjeong Park, Chulyun Kim

1) CAT(Code-Aligned Type sequence)

- 딥러닝 기반 자동 주석 생성 모델에 쓰이는 소스코드 구조 정보 데이터인 Abstract Syntax Tree(AST), Structure-Based Traversal(SBT) 등을 수집 후 기존 연구에서 활용하는 소스코드 데이터의 한계점을 분석하고 이를 보완하는 **새로운 데이터 타입**을 설계했습니다.
- 기존 연구와 다른 방향으로 정제한 새로운 데이터 타입인 Code-Aligned Type sequence(CAT)을 제안하고 이를 활용해 **소스코드의 의미적, 구조적 정보 모두 추출**하는 방법을 설계했습니다.
- CAT은 소스코드 정보 손실과 중복 및 불필요한 정보 포함 등의 기존 연구의 한계점을 보완하여 **정보 손실을 최소화**합니다.
- CAT은 소스코드에서 생성된 AST로부터 코**드 토큰과 타입 토큰이 순서에 따라 정렬되어 추출된 정보로 기존의 SBT 정보보다 더 나은 syntactic 정보**를 얻을 수 있습니다.

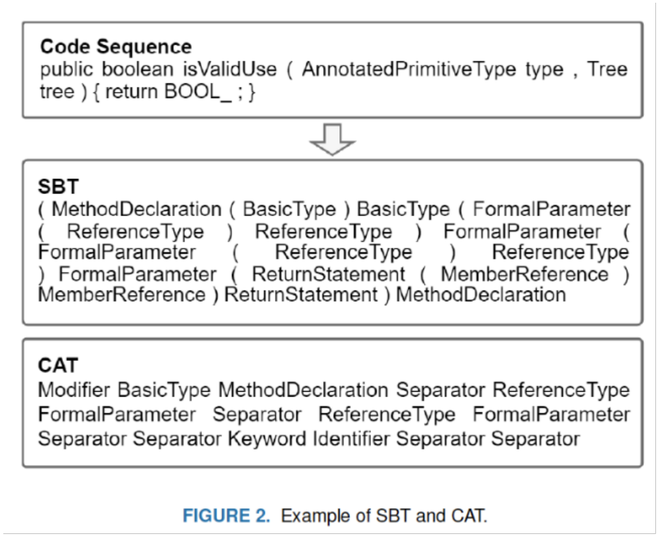

FIGURE 2. Example of SBT and CAT.

ALSI-Transformer: Transformer-Based Code Comment Generation with Aligned Lexical and Syntactic Information

Youngmi Park, Ahjeong Park, Chulyun Kim

2) ALSI-Transformer

- 소스코드 파일의 하위 단위인 **함수에 대해 자연언어 설명 생성 알고리즘**을 구현해 새로운 모델인 **ALSI-Transformer을 제안**
- ASLI-Transformer는 Transformer 기반 딥러닝 모델로 함수 단위의 소스코드를 입력으로 넣었을 대 적절한 자연언어 주석을 출력으로 생성하는 것을 목표로 함. 특히, **CAT과 Gate Network를 활용해 소스코드의 lexical, syntactic 정보를 결합했습니다.**

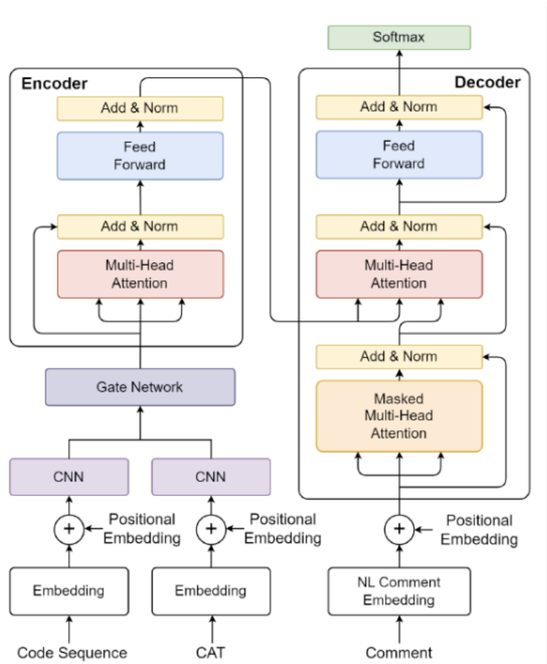

FIGURE 1. Structure of the ALSI-Transformer.

TABLE 8. Results of comparison between ALSI-Transformer (*Gate Network*) and five aggregation methods in terms of BLEU and METEOR.

Aggregation Method	BLEU	METEOR
Alternation	51.26	64.05
Separation	52.40	65.20
Addition	53.21	65.89
Average	50.75	63.46
Concatenation	51.41	64.23
Gate Network	53.80	66.11

- Lexical, Syntatic 2개의 정보를 결합하기 위핸 1개의 인코더를 사용했습니다.
- 2개의 정보를 결합하는 방법으로 Alternation, Separation, Addition, Average, Concatenation, Gate Network를 실험했고 **최종 결과 Gate Network의 방법이 높은 성능**을 보였습니다.
- 따라서 ALSI-Transformer는 Gate Network를 사용합니다.
- 자세한 결합 방법은 다음장에 준비되어 있습니다.

ALSI-Transformer: Transformer-Based Code Comment Generation with Aligned Lexical and Syntactic Information

Youngmi Park, Ahjeong Park, Chulyun Kim

2-1) ALSI-Transformer - Aggregation Methods

1) Alternation: 코드 시퀀스와 CAT이 번갈아 나오는 결합 방법 입니다.

2) Seperation: 코드 시퀀스 묶음과 CAT 묶음이 '<code>'라는 토큰으로 연결되는 결합 방법 입니다.

3) Addition: 코드 시퀀스 임베딩과 CAT 임베딩을 Add 한 결합 방법 입니다.

4) Average: 코드 시퀀스 임베딩과 CAT 임베딩을 Average 한 결합 방법 입니다.

5) Concatenation: 코드 시퀀스와 임베딩과 CAT 임베딩을 Concatenation한 결합 방법 입니다.

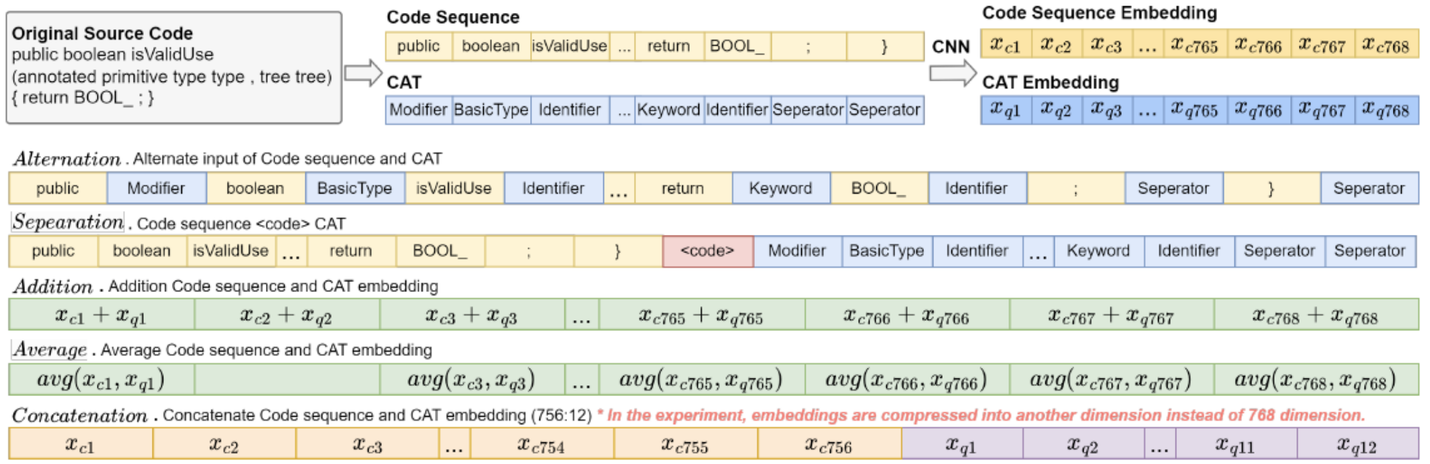

FIGURE 3. Example of five aggregation methods. Code sequence and CAT extracted from original source code used for *Alternation* and *Separation* methods. Code sequence embedding and CAT embedding are generated through the CNN layer. Two embeddings are used for *Addition* and *Avgerage*. *Concatenation* makes embedding in different ratios.

ALSI-Transformer: Transformer-Based Code Comment Generation with Aligned Lexical and Syntactic Information

Youngmi Park, Ahjeong Park, Chulyun Kim

Experiment Results

- 6개의 베이스라인과 비교하기 위해 **BLEU, N-BLEU, METHOR에 대해 성능 평가**를 진행했습니다.
- 논문의 방법이 주석 생성에서 **SOTA(2022년 7월 기준)을 달성함**을 확인했고 이 방법이 **Lexical 과 Syntatic 정보 결합이 중요**하다는 것을 입증했습니다.
- 기존의 방법보다 모델 사이즈, 파라미터 개수가 적고 학습시간이 효율적임을 확인했습니다.
- 또한 2개의 정보를 처리할 때 two encoder와 one encoder 중 one encoder의 성능이 좋음을 확인했습니다.

TABLE 5. BLEU, n-gram BLEU, and METEOR score for our model ALSI-Transformer compared with six baselines.

Models	BLEU	BLEU 1	BLEU 2	BLEU 3	BLEU 4	METEOR
Seq2Seq (attention) [9]	37.87	46.53	41.53	37.81	35.04	23.29
Transformer [10]	45.55	55.62	46.30	41.57	38.69	29.06
DeepCom [6]	20.26	32.88	21.91	18.93	17.35	31.72
Hybrid-DeepCom [11]	38.20	51.63	40.56	36.70	34.41	51.26
ComFormer [7]	42.99	55.31	47.57	41.72	36.26	59.12
SeTransformer [4]	49.00	62.78	51.91	47.47	44.62	62.99
ALSI-Transformer	53.80	57.26	56.29	52.49	50.03	66.11

TABLE 7. Comparison of the number of model parameters, training time, and model size between ALSI-Transformer and SeTransformer.

Models	# of Parameters	Training Time(s)	Model Size (GB)
ALSI-Transformer	146,939,910	174,279	8.3
SeTransformer [10]	167,308,038	265,140	10

TABLE 9. Comparison results between ALSI-Transformer and ALSI-Transformer (two-encoder).

Models	BLEU	METEOR	Model Size (GB)
ALSI-Transformer	53.80	66.11	8.3
ALSI-Transformer (two-encoder)	50.05	63.30	9.5