

SOTitle: A Transformer-based Post Title Generation Approach for Stack Overflow

Ke Liu, Guang Yang, Xiang Chen, Chi Yu

2022 IEEE International Conference on Software Analysis,
Evolution and Reengineering (SANER)

発表者：井町 瑠貴

背景

- | Stack Overflowでは、開発者が自分の問題を投稿し、他のユーザに回答してもらうことができる
- | 開発者は、質の高い回答を求めているが、質の高い回答が得られない質問も存在する

背景

いくつかの先行研究によると、質問の質が低い理由として、タイトルが不適切であることがあげられている

また、知識不足などの理由から自分で適切なタイトルをつけることが難しい場合もある



タイトルを自動生成することが
求められている

先行研究

- | 先行研究では、コードスニペットを用いてタイトルの自動生成を行うCode2Queが提案されている
- | しかし、同じコードスニペットであっても、異なる質問である場合があり、投稿の問題説明は貴重な情報源である

本研究では、コードスニペットと問題説明の両方を入力として考慮することで、高品質なタイトルを生成する

データセット

- 人気のある4つの言語(Java, C#, Python, JavaScript)を対象とする
- 以下の3つのルールを設定

1. スコアが5以下
2. 受け入れられた回答を持つ
3. コードスニペットを持つ

TABLE I
STATISTICAL INFORMATION OF CORPUS SPLIT RESULTS

Language	Training	Validation	Testing
Java	60,000	3,959	5,000
C#	60,000	6,817	5,000
Python	60,000	7,742	5,000
JavaScript	60,000	5,780	5,000

提案手法

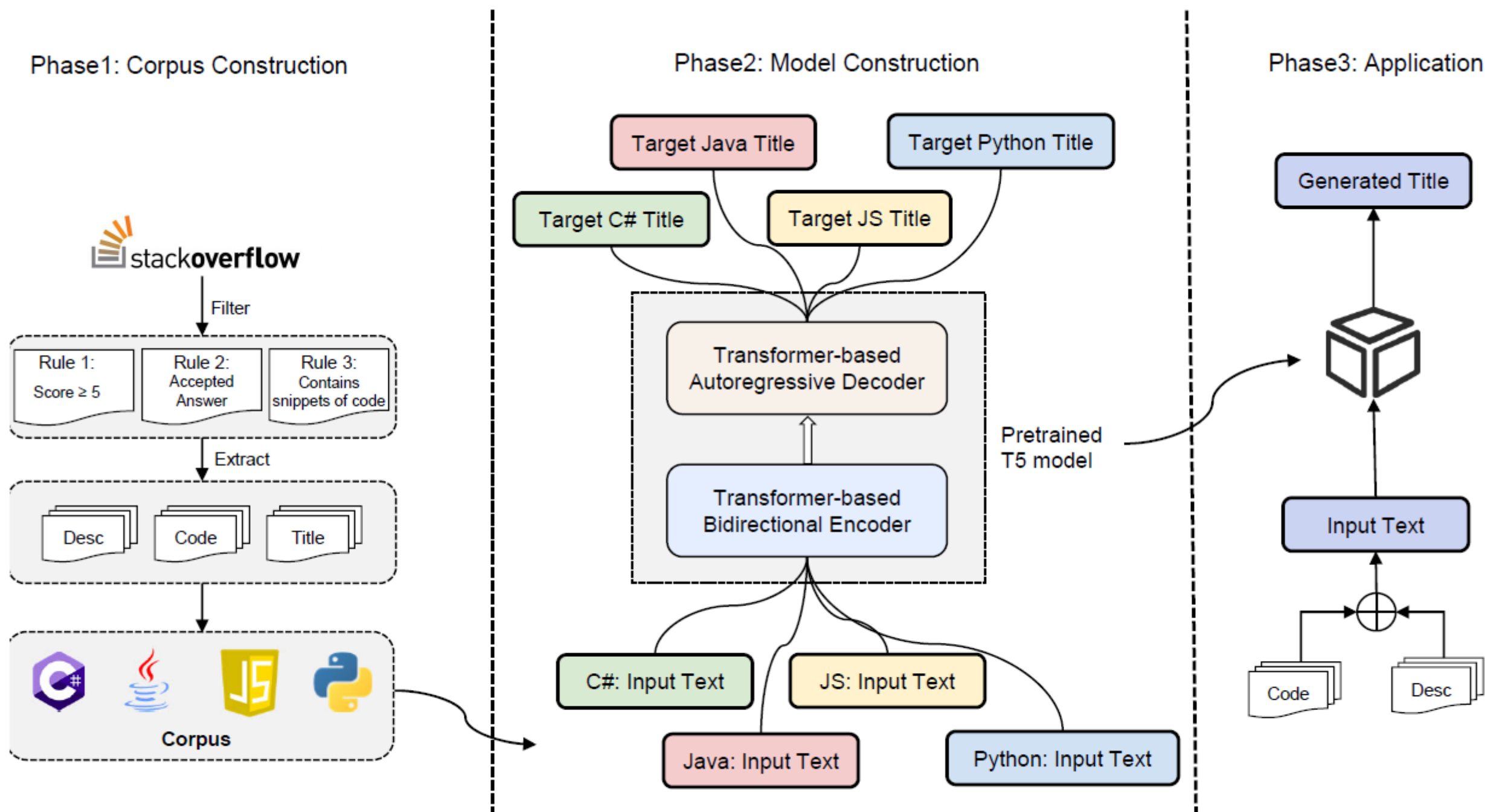
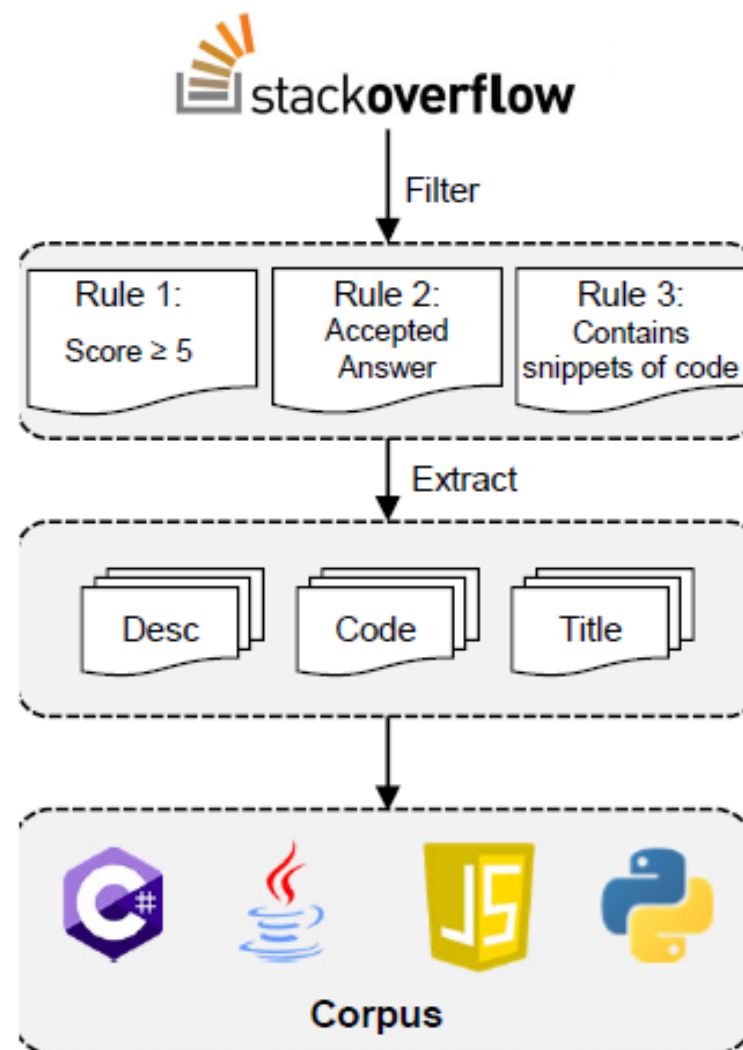


Fig. 2. Framework of our proposed approach SOTitle

提案手法 | コーパス構築

データセットに含まれる質問から、
問題説明(Desc), コードスニペット(code),
投稿のタイトル(Title)をトリプレットとして抽出



提案手法 | モデル構築

Multi-modal Input Modeling

SentencePiece法を採用

- テキストから文字や部分文字列に分解する
- 入力列をUnicodeで符号化された列とみなす

入力は以下のような形式

$$X = \text{prefix} \oplus X_{\text{desc}} \oplus \langle \text{code} \rangle \oplus X_{\text{code}}$$

Transformer-based Bidirectional Encoder

目的：Xの表現を学習する

- Self-attention層とfeed-forward neural networkからなる

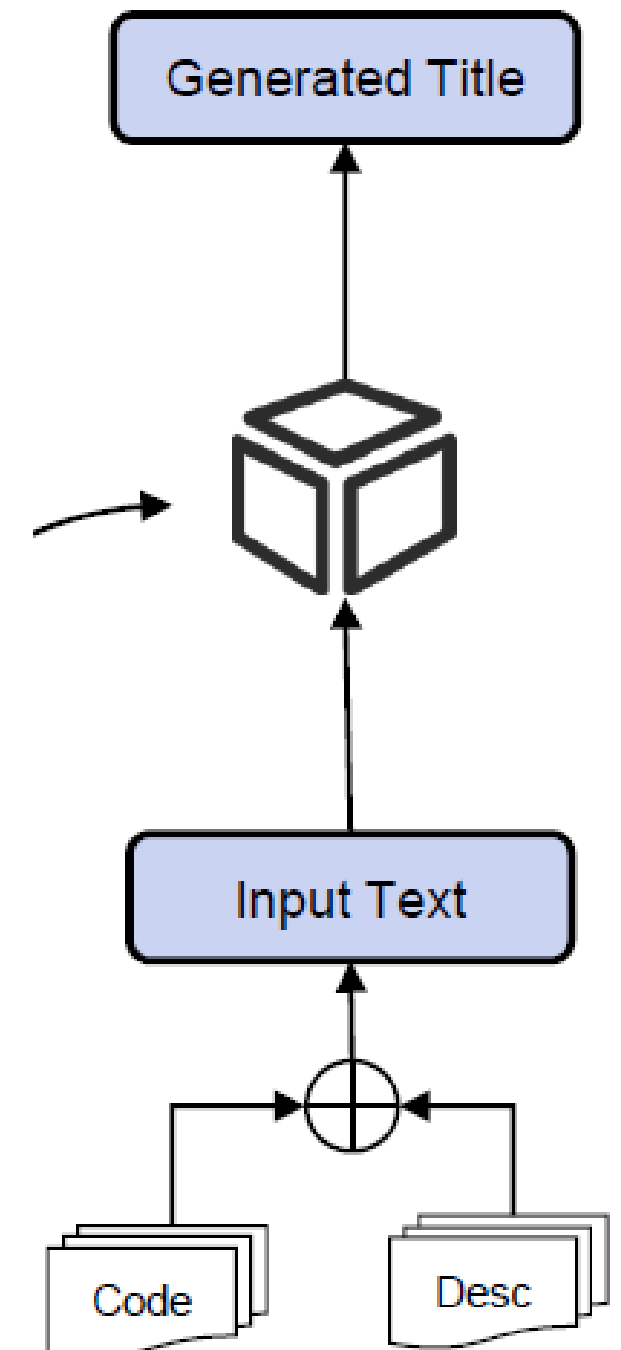
$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \quad \text{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2$$

提案手法 | モデル適用

Phase3: Application

新しい質問のコードスニペットと
問題説明をモデルに入力

モデルはビームサーチアルゴリズムに
より対応する投稿のタイトルを
自動的に生成する



実験設定 | リサーチクエストション

RQ1

SOTitleは自動評価において最新のベースラインを上回れるか

RQ2

入力モダリティの違いはSOTitleの性能にどのように影響するか

RQ3

SOTitleは、人間実験においてCode2Queを上回ることができるか

実験設定 | 性能指標

自動評価

Rouge-1, Rouge-2, Rouge-Lを使用する

人による評価

1. タイトルについて2~-1点で評価
2. タイトルについて3つの観点から評価

Problem Description: Assuming I have a column called df.Text which contains text (more that 1 sentence) and I want to use polyglot Detector to detect the language and store the value in a new column df['Text-Lang'] how do I ensure I also capture the other details like code and confidence ... Code Snippet: testEng ="This is English" lang = Detector(testEng) print(lang.language) ...	
Title 1: How to detect the language and store the value in a pandas dataframe ?	Title 2: Polyglot Detector Detector Detector Detector
Q1-1: What is the score of Title 1 ? A. EXCELLENT. B. GOOD. C. BORDERLINE. D. BAD.	Q2-1: What is the score of Title 2 ? A. EXCELLENT. B. GOOD. C. BORDERLINE. D. BAD.
Q2-1: What do you think of Title 1 ? (multiple choices) A. Relevance GOOD. B. Conciseness GOOD. C. Expressiveness GOOD.	Q2-2: What do you think of Title 2 ? (multiple choices) A. Relevance BAD. B. Conciseness BAD. C. Expressiveness BAD.

Fig. 4. A survey example used in our human study

実験設定 | ベースライン

以下の6つのベースラインを選択している

1. Code2Que

- LSTMベースの深層学習より，タイトルを自動生成

2. BM25

- 情報検索における順位付けの手法

3. Transformer

- 自然言語処理における理解や生成のタスクで用いられる

4. BART

- テキスト要約タスクにおいて良好な性能を発揮

5. NMT(Neural Machine Translation)

- コード変更からのコミットメッセージの生成に使われる

6. Hybrid-DeepCom

- コードコメント生成に使用されるニューラルモデル

結果分析 | RQ1

RQ1 : SOTitleは自動評価において最新のベースラインを上回れるか

SOTitleが最もよいパフォーマンスを示した



タイトル生成問題を，関連するタスクとして捉え，
補完し合うことで互いのパフォーマンスを
向上させることができると考えられる

TABLE III
COMPARISON RESULTS BETWEEN OUR PROPOSED APPROACH AND STATE-OF-THE-ART BASELINES

Approach	Java			C#			Python			JavaScript		
	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L
BM25	10.045	1.150	9.294	10.183	1.760	9.552	11.443	1.475	10.487	9.782	1.182	9.205
NMT	18.017	3.393	17.166	18.656	4.885	17.850	19.482	4.090	18.549	17.341	3.279	16.738
Transformer	12.689	1.957	12.332	12.252	2.046	12.115	14.125	2.236	13.602	13.153	2.082	12.751
BART	24.801	8.918	23.343	23.395	9.545	22.286	27.272	10.321	25.672	25.421	9.724	24.095
Hybrid-DeepCom	10.620	1.511	10.232	12.346	2.684	11.884	11.512	1.853	11.156	14.103	2.421	13.323
Code2Que	22.021	6.787	21.075	23.476	8.605	22.584	24.635	8.301	23.282	23.958	8.164	22.942
SOTitle	29.328	10.968	27.262	29.550	11.958	27.613	31.828	11.985	29.287	31.150	11.822	28.897

結果分析 | RQ2

RQ2 : 入力モダリティの違いはSOTitleの性能にどのように影響するか

- ・コードと問題説明を比較した場合, 問題説明から有用な情報が得られることがわかる
- ・コードと問題説明から得られる情報は補完的であり, どちらも考慮が必要である

TABLE IV
THE PERFORMANCE OF SOTITLE AND CODE2QUE WITH DIFFERENT COMBINATIONS OF INPUT MODES

Approach	Java			C#			Python			JavaScript		
	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L
Code2Que _{code}	14.157	2.576	13.590	14.895	3.388	13.602	12.705	2.184	12.015	12.648	2.109	12.112
SOTitle _{code}	15.064	2.644	14.255	15.445	3.525	14.255	13.534	2.236	12.801	13.467	2.295	12.952
Code2Que _{desc}	21.213	6.042	20.637	22.896	8.022	21.957	24.012	8.001	22.439	23.288	7.979	22.310
SOTitle _{desc}	24.586	9.101	23.083	24.812	10.063	23.439	26.966	10.408	25.226	26.294	10.176	24.922
Code2Que	22.021	6.787	21.075	23.476	8.605	22.584	24.635	8.301	23.282	23.958	8.164	22.942
SOTitle	29.328	10.968	27.262	29.550	11.958	27.613	31.828	11.985	29.287	31.150	11.822	28.897

向上

結果分析 | RQ3

RQ3 : SOTitleは人間実験において, Code2Queを上回ることができるか

- SOTitleは, Code2Queよりも高品質なタイトルを生成することができることがわかる
- コードスニペットが複雑な場合や, 問題の説明に有用な情報が欠けている場合, うまく生成できない

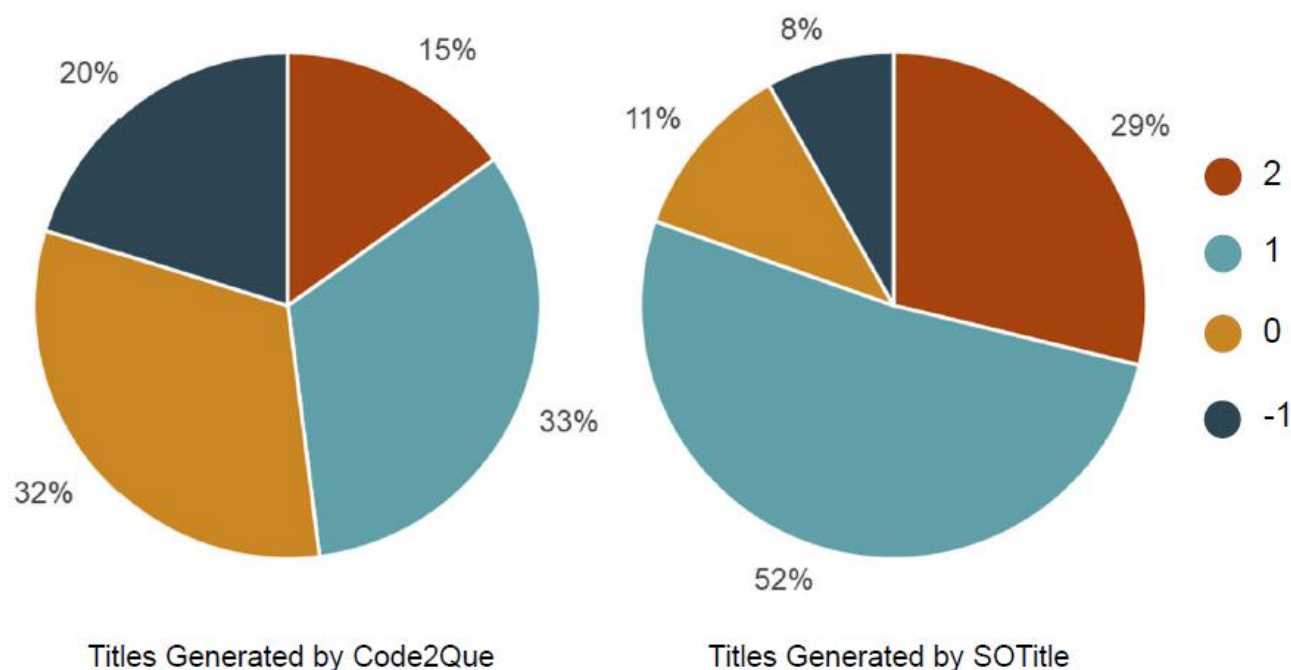


Fig. 7. Rating distribution of human votes and each sector presents the proportion of votes with the corresponding rating score.

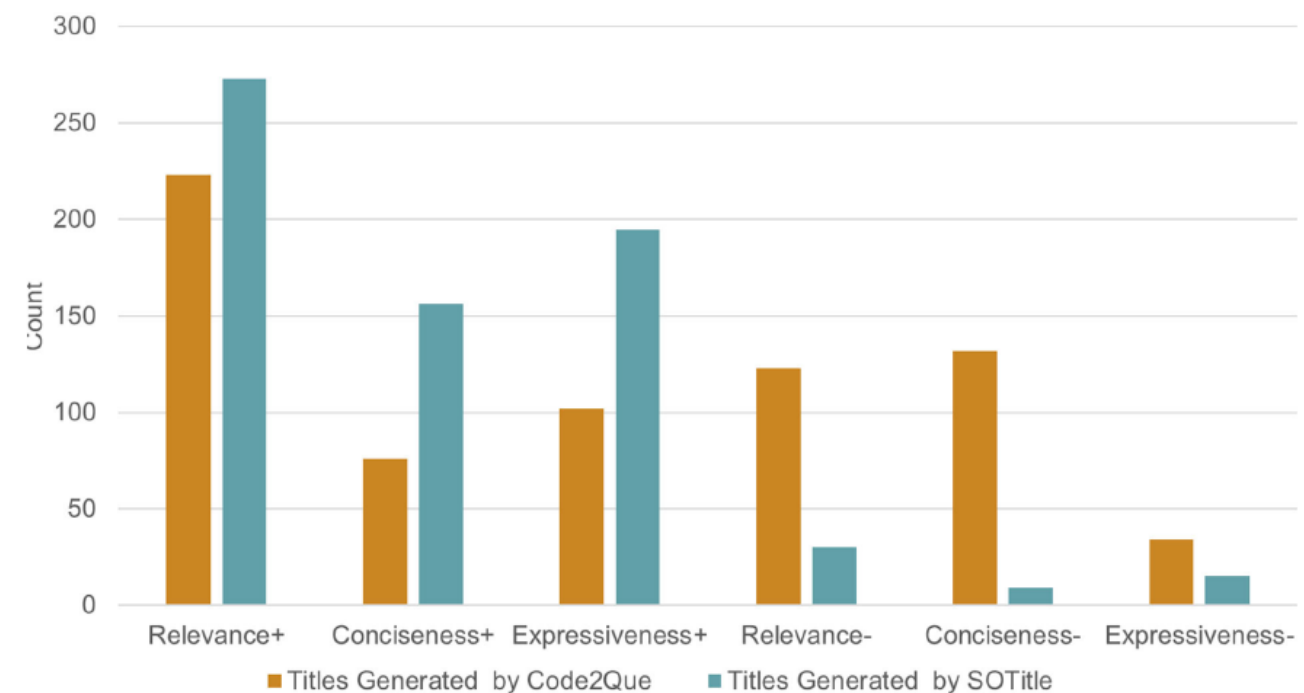


Fig. 8. Advantages (+) and disadvantages (-) distribution of human comments. Each bar presents the count of corresponding perspective.

結論

- | 質問投稿に含まれるコードスニペットと問題記述を活用し、タイトルを自動生成するSOTitleを提案した
- | 6つのベースラインをRougeの性能指標で比較し、SOTitleの競争力が示された
- | Code2QueとSOTitleを用いて人体実験を行い、SOTitleの有用性を検証した

所管

- 具体的なモデル構築の手法についてあまり理解できなかった
- 問題説明の記述能力とタイトル作成能力には、相関関係があるように感じるが、実際にどうなのか気になった

参考

| Transformerについて

<https://tips-memo.com/translation-jayalmmar-transformer>

<https://crystal-method.com/topics/transformer-2/>

| ROUGEについて

<https://qiita.com/icoxfog417/items/65faecbbe27d3c53d212>

補足資料

Rouge³

Rouge

自動生成された要約と，人間が作成した要約を比較し質を評価

Rouge-1, Rouge-2

それぞれ，自動生成要約と正解要約の間の1-gram, 2-gramの共起を評価

Rouge-L

「一致する最大のシーケンス」
(longest common subsequence : LCS)を評価

³<https://github.com/pltrdy/rouge>

結果分析 | RQ1の例

RQ1 : SOTitleは自動評価において最新の
ベースラインを上回れるか

生成したタイトルの例

Problem body :	Title :
I see that I can append to a list while iterating over it <pre>lst = [1] for i in lst: lst.append(i+1) print(i)</pre> Am I allowed to make use of this behavior? or is it discouraged? I note that the same can not be said for <code>set</code> <pre>lst = set([1]) for i in lst: lst.add(i+1) print(i)</pre> Error: size changed during iteration.	Ground truth: Is it correct to append to a list while iterating over it ? BM25: Iterating over partitions in Python NMT: How to append to a list ? Hybrid-DeepCom: Appending a list to a list comprehension in python Transformer: Python list comprehension BART: Python list append () Code2Que: How to append to a list in python ? SOTitle: Is it discouraged to append to a list while iterating over it ?

Fig. 6. The titles generated by SOTitle and baselines for a question post related to the Java programming language