

AI-based Question Answering Assistance for Analyzing Natural- language Requirements

Saad Ezzini*, Sallam Abualhaija*, Chetan Arora^{‡§}, Mehrdad Sabetzadeh[†]

*SnT Centre for Security, Reliability and Trust, University of Luxembourg, Luxembourg

[‡]Deakin University, Geelong, Australia

[§]Monash University, Victoria, Australia

[†]School of Electrical Engineering and Computer Science, University of Ottawa, Canada

2023 IEEE/ACM 45th International Conference on
Software Engineering (ICSE), pp.??-??

発表：山村慎太郎

ソフトウェア要求仕様書（SRS）

システム開発において、顧客側が求める機能や品質などの仕様をまとめたドキュメント

—システム開発の初期段階で作成され、開発の指針となる

—自然言語（NL）で記述される

要求定義と要件定義

	内容	目線
要求定義	顧客の要望を明確化したもの	顧客寄り
要件定義	要望を実現するために備える機能	プログラマ寄り

SRS作成における課題

NLによる記述のため、曖昧さや不完全性、矛盾など内容に関する様々な問題が発生しうる

→SRSは品質保証の対象に

問題の修正やスクリーニングに時間がかかる

原因：

※スクリーニング：
問題の選別やふるい分け。

- ・手作業で行われる
- ・要件を正確に記述・解釈するための知識・情報の呼び出しが必要
- ・中には数十～数百ページにわたるSRSも存在する

課題：SRSの作成・品質保証には時間やコストがかかる

Open Domain QA (OpenQA)

質問に対する回答を、外部の文書から見つけ出して自動で出力する、自然言語処理(NLP)タスク

一般的なQAタスク

学習

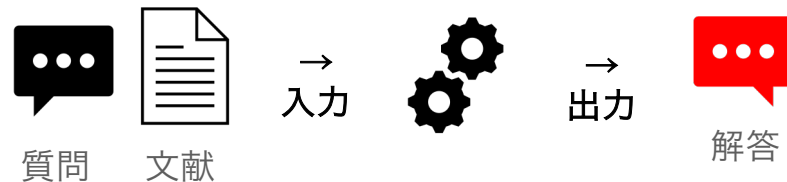


OpenQA

学習



回答



回答



外部のコーパス(Wikipediaなど)を”文献”
として自力で参照，回答を抽出
→幅広い質問に対応できる



外部コーパスの文書
(Wikipediaなど)

OpenQAを用いて、**SRS**の品質保証を支援

目的・アプローチ

目的

SRSの作成・品質保証の支援

アプローチ

OpenQAの手法を用いた”QAssist”を提案,
SRSの内容に関するQ&Aを自動で行えるように
→スクリーニング・修正の手助け

QAssist(OpenQA)による支援のイメージ

Text passages extracted from requirements specifications

[DR-13] The wet mass of the spacecraft shall not exceed 3004 kg.

...

[DR-27] The wet mass of the spacecraft shall be computed at a regular basis. **Incomplete requirement?**

...

[SCIR-19] The spacecraft shall carry a science camera system as part of the scientific payload and a separate navigation camera system.

[SCIR-20] The navigation camera system shall be used only for the detection of the comet nucleus.

...

Inconsistent requirements?

[MISS-29] The spacecraft shall use a navigation camera system for asteroid detection, approach navigation and close fly-by tracking of the asteroid.

Q1 What is the wet mass of a spacecraft?
Domain Knowledge

Q2 How often shall the wet mass be computed?

Q3 What is the function of navigation camera system?



Software
Requirements
Specification
(SRS)



Domain-
specific
Corpus

Text passages extracted from a domain-specific corpus

In aerospace engineering, mass ratio is a measure of the efficiency of a rocket.

It describes how much more massive the vehicle is with propellant than without; that is, the ratio of the rocket's wet mass (vehicle plus contents plus propellant) to its dry mass (vehicle plus contents).

Navigation is a field of study that focuses on the process of monitoring and controlling the movement of a craft or vehicle from one place to another. The field of navigation includes four general categories: land navigation, marine navigation, aeronautic navigation, and space navigation.

A navigation system is a computing system that aids in navigation. Navigation systems may be entirely on board the vehicle or vessel that the system is controlling (for example, on the ship's bridge) or located elsewhere, making use of radio or other signal transmission to control the vehicle or vessel.

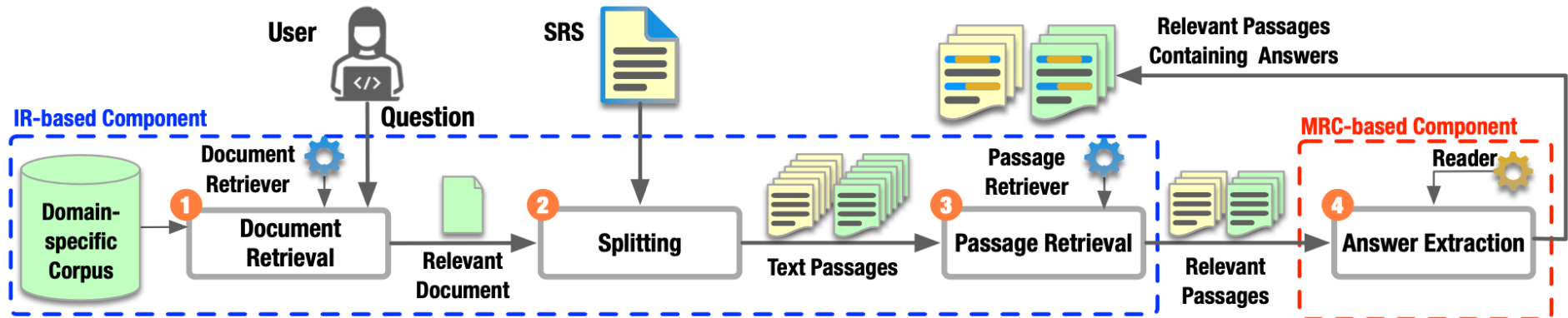
SRSから抽出された文

外部コーパスの文献から抽出された文

- ①SRSの内容に関する，疑問点などを質問
- ②QAssistがSRS・文献から答えになりそうな文・箇所を抽出
- ③抽出された文・箇所を見て，疑問の解消やSRSの問題点の発見に役立てる

例) Q1：外部コーパスにより，疑問の解消
Q2：SRSの不完全性の発見
Q3：SRS内の矛盾を発見

QAssistの概要①



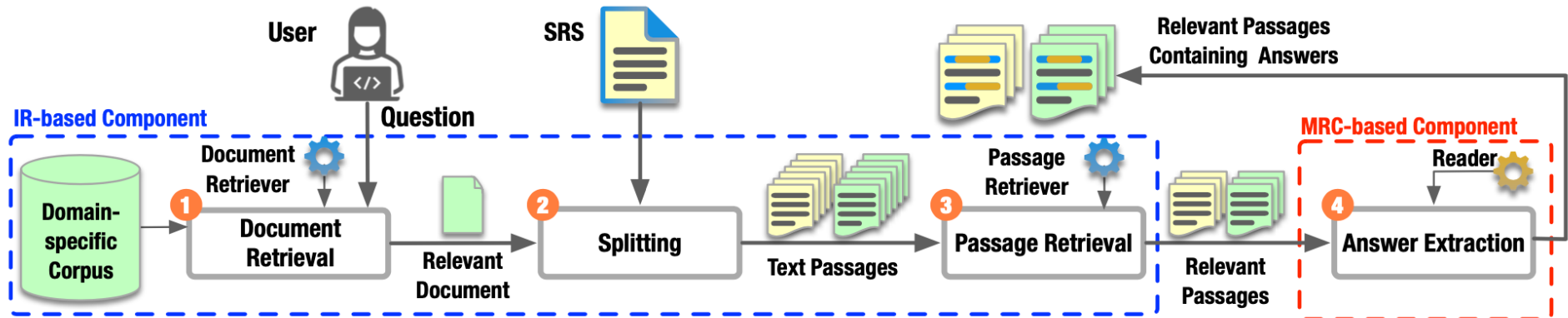
前半部分：質問文と関連度の高いパッセージ※の抽出

- ① 質問文を入力し， それに関連のある文献を外部コーパスから取得
- ② 取得した文献・SRSをそれぞれパッセージに分割
- ③ 分割した全てのパッセージの中から， 質問文に最も関連のある上位のパッセージを選択

主に， 既存のIR(情報検索)技術を利用

ex) TF-IDF, BM25 など

QAssistの概要②

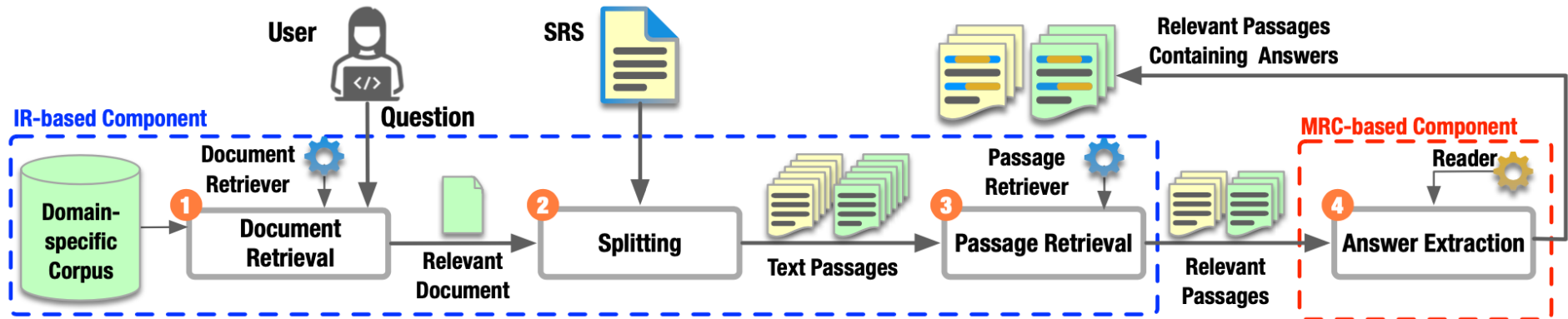


後半部分：回答となる部分の抽出

④ ③で取得したパッセージから，質問に対する回答となりそうな部分を抽出し，ハイライトで示す

主に，事前学習済み言語モデルが用いられる
ex) BERT, RoBERTa など

QAssistの性能検証実験



①③④の正確さを，使用モデルごとに評価

- ① 外部コーパスから質問に関連する文献を取得できるか
- ③ パッセージ集合から質問に関連するパッセージを取得できるか
- ④ 関連パッセージから回答箇所を抽出できるか

それぞれRecallなどの評価メトリクスで性能を評価

実験用データセット問題

既存手法(OpenQA)のデータセットは利用できない
既存手法(OpenQA)とQAssistとの相違点→必要な文献

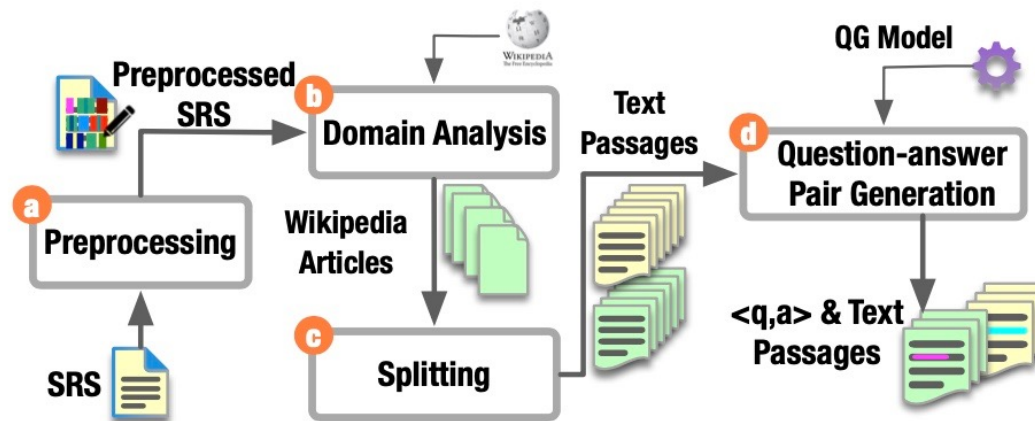
必要な文献

- ・ OpenQA : 外部コーパスの文献のみ
- ・ QAssist : 外部コーパスの文献とSRS

→③④検証実験には、SRSも含むデータセットが必要
(既存手法のデータセットでは不十分・利用できない)

実験用のデータセットを新たに作成
作成手法"REQestA"を提案

REQuestA



SRSとコーパスの文献から、<質問q, 回答a, パッセージ>のペア(正解データ)を生成・取得

質問生成に、事前学習済み言語モデル”T5”を用いる

- a. 入力されたSRSを前処理
- b. 入力SRSの分野を特定，文献を取得
- c. SRSと特定した分野の文献をパッセージに分割
- d. パッセージからランダムに回答部分を選択，T5モデルで対応する質問を生成

⇒ SRS, 外部文献を含むデータセット(正解データ)の完成

データセットと実験方法

データセット

Domain	$ \mathcal{T}_D $	$\langle q, a \rangle$		SRS	$ \mathcal{T}_S $	$\langle q, a \rangle$	
		auto	man			auto	man
Aerospace	42	45	53	#1	24	8	18
				#2	107	37	40
Defence	94	38	50	#3	11	5	4
				#4	71	19	26
Security	23	4	8	#5	18	15	13
				#6	4	2	2
Total	159	87	111	-	235	86	103

SRS：既存研究のものを6種類利用

T_D ：コーパス文献からのパッセージ数

T_S ：SRSからのパッセージ数

$\langle q, a \rangle$ ：質問・解答のセットの数

auto：REQuestAによる自動生成

man：人力で作成

実験方法

- ① QAssistに質問(q)とSRSを入力→回答させる
- ② 回答がデータセットの回答の正解データ(a)と一致するかを、
それぞれ評価

RQ

RQ1

前半部分(質問に関連する文献やパッセージの取得)において精度の高いモデルは？

RQ2

後半部分(パッセージから回答部分の抽出)において精度の高いモデルは？

RQ3

QAssistは実用的な時間で動作するか？

RQ1 (前半部分①③のモデル評価)

- ① 外部コーパスから質問に関連する文献を正確に取得できるか
→ Recall評価の結果, "BM25"が最も優れている
(計算効率などの関係)

Domain	$ \mathcal{D} ^\dagger$	TF-IDF	BM25	Dense	Reranking
Aerospace	1158	100	100	99.0	100
Defence	781	100	100	98.9	100
Security	50	100	100	91.7	100

$^\dagger |\mathcal{D}|$ is the number of articles in the corpus ($\mathcal{D}$) of Wikipedia articles.

- ③ パッセージ集合から質問に関連するパッセージを取得できるか
→ Recall・nDCG評価の結果, "Reranking"が最も優れている
(平均Recall: 90.1%(対文献), 96.5%(対SRS))

※nDCG (検索結果の, ランキングを考慮した関連性)

RQ2(後半部分④のモデル評価)

④ 関連ページから回答箇所を正確に抽出できるか

→一致度※によって、優れたモデルが異なった

- ・ 正解データと抽出箇所が**完全一致**

→"RoBERTa"がPrecision24.6%

- ・ 正解データと抽出箇所が**一部一致**

→"DistilBERT"がPrecision86.4%

- ・ 正解データと抽出箇所が**意味一致**

→"ALBERT"がPrecision84.2%

総合的に見れば, "RoBERTa"がbest

Model	Accuracy			F1
	<i>Exact</i>	<i>Partial</i>	<i>Semantic</i>	
ALBERT	24.3	79.1	84.2	64.6
q_S	31.7	78.0	86.4	67.6
q_D	17.2	80.2	82.1	61.7
BERT	21.4	70.6	82.9	63.1
q_S	28.3	70.4	83.8	66.4
q_D	14.8	70.8	82.1	59.9
DistilBERT	23.0	86.4	75.9	61.0
q_S	32.7	86.4	77.3	67.5
q_D	13.7	86.5	74.6	54.8
ELECTRA	21.1	81.3	81.0	60.1
q_S	31.2	82.1	80.6	65.0
q_D	11.5	80.5	81.4	55.4
MiniLM	23.3	73.3	82.4	63.4
q_S	32.4	73.6	82.6	66.1
q_D	14.6	73.0	82.2	60.9
RoBERTa	24.6	60.2	84.0	65.2
q_S	32.8	61.0	84.3	70.1
q_D	16.8	59.4	83.7	60.5

RQ3(QAssistは実用的な時間で動くか?)

Google Colab(無料プラン)で実行
選択IRモデル

①BM25, ③Reranking

結果

ALBERTのオーバーヘッド時間※を
除いた, 一質問あたりの平均実行時間
= 10.36秒

→十分実用的であると言える

※オーバーヘッド時間
Q&A以外の実行時間?

Model	Time
ALBERT	193.2
q_S	
q_D	32.2
BERT	
q_S	5.8
q_D	
DistilBERT	19.1
q_S	
q_D	5.0
ELECTRA	
q_S	11.6
q_D	
MiniLM	
q_S	
q_D	
RoBERTa	
q_S	
q_D	

まとめ

顧客の要望が記述される要求仕様書(SRS)は、品質保証に時間がかかるという問題を抱えている

SRSの品質保証をQ&A形式で手助けする**QAssist**を提案

実験用データセットREQuestAの作成

実験の末、QAssistの一定の実用性が見てとれた

所感

| 上流工程に関する論文をあまり読んでこなかった
→背景や手法を学べて良かった

| 精度は高そうだし，本当に使ってみたら役にたつ？
→実際に効率化できるか，という意味での実用性？

| “ソフトウェア”仕様書で“Aerospace”…？(データセット)

補足：データセッツ

③④：半分はREQuestAによる生成(人力チェック済み), もう半分は人力で作成

Td wiki文献のパッセージ分割数

Ts SRSのパッセージ分割数

Domain	$ \overline{\mathcal{T}_D} $	$\langle q, a \rangle$		SRS	$ \mathcal{T}_S $	$\langle q, a \rangle$	
		<i>auto</i>	<i>man</i>			<i>auto</i>	<i>man</i>
Aerospace	42	45	53	#1	24	8	18
				#2	107	37	40
Defence	94	38	50	#3	11	5	4
				#4	71	19	26
Security	23	4	8	#5	18	15	13
				#6	4	2	2
Total	159	87	111	-	235	86	103

補足RQ1 結果表

		Top-1		Top-3		Top-5		Top-10	
From $\mathcal{T}_S$		R	nDCG	R	nDCG	R	nDCG	R	nDCG
	(1)	60.3	60.3	78.6	70.9	84.3	73.3	89.8	75.1
	(2)	62.8	62.8	78.5	71.9	85.4	74.6	92.4	76.9
	(3)	52.9	52.9	81.2	70.3	86.5	72.5	88.5	73.2
	(4)	78.9	78.9	90.1	85.8	92.2	86.6	92.4	86.7
From $\mathcal{T}_D$		R	nDCG	R	nDCG	R	nDCG	R	nDCG
Aerospace	(1)	43.6	43.6	64.7	56.0	68.6	57.5	91.5	94.9
	(2)	50.5	50.5	83.0	70.1	91.7	73.6	95.0	74.7
	(3)	66.7	66.7	87.3	79.5	90.6	80.8	91.3	81.0
	(4)	75.1	75.1	95.0	87.3	95.0	87.3	95.0	87.3
Defence	(1)	41.9	41.9	62.1	54.1	66.7	55.9	86.3	62.2
	(2)	38.0	38.0	81.2	64.1	89.3	67.3	94.6	69.1
	(3)	77.2	77.2	89.8	84.7	91.2	85.2	92.5	85.6
	(4)	76.0	76.0	94.6	87.6	94.6	87.6	94.6	87.6
Security	(1)	33.4	33.4	70.0	53.8	70.0	53.8	80.0	57.4
	(2)	43.3	43.3	70.0	59.3	100	71.3	100	71.3
	(3)	63.3	63.3	100	85.2	100	85.2	100	85.2
	(4)	80.0	80.0	100	92.6	100	92.6	100	92.6

(1) TF-IDF, (2) BM25, (3) Dense, and (4) Reranking.