令和 7 年 9 月 4 日 (木)　　M2 堀本 隆誠 BGA24061

報告書

1　今週の進捗

- SRAG の調査 (未実装)

2　Structured RAG について

Structured Retrieval-Augmented Generation (SRAG) [1] は, 複数の文書や情報源に散らばった情報をもとに質問応答を行うための新しいフレームワークである. 従来の RAG は, 検索してきた非構造的なテキストをそのまま LLM に渡すため, 情報の集約や複雑な推論が苦手という課題があった.

SRAG はこの課題に対し, 情報を一度「テーブル」のような構造化データに整理してから LLM に処理させる, というアプローチを取る. 具体的には, まず質問内容を分析してどのような情報が必要かを判断し, テーブルの設計図 (スキーマ) を生成する. 次に, そのスキーマに従って関連文書から情報を抽出し, テーブルを埋めていく. 最後に, 完成したテーブルに対して問い合わせをすることで, より正確で信頼性の高い回答を生成することができる. この手法により, LLM は単なるテキストの海から答えを探すのではなく, 整理されたデータに基づいて分析や集計できるようになる.

図 1 は, SRAG の全体的な流れを示したものである. (a) では, 質問 (例:「カナダ人のチューリング賞受賞者は何人？」) から Knowledge Graph を使って関連情報を検索する. (b) は従来の RAG のアプローチであり, 情報が整理されていないため不完全な答えや誤った答えを返しがちであることを示している. 一方, (c) が SRAG のアプローチであり, 質問からテーブルのスキーマ (「名前」「国籍」など) を生成し, 情報を抽出してテーブルを完成させる. 最終的に, このテーブルに対して問い合わせることで,「6 人」という正確な答えを導き出している.

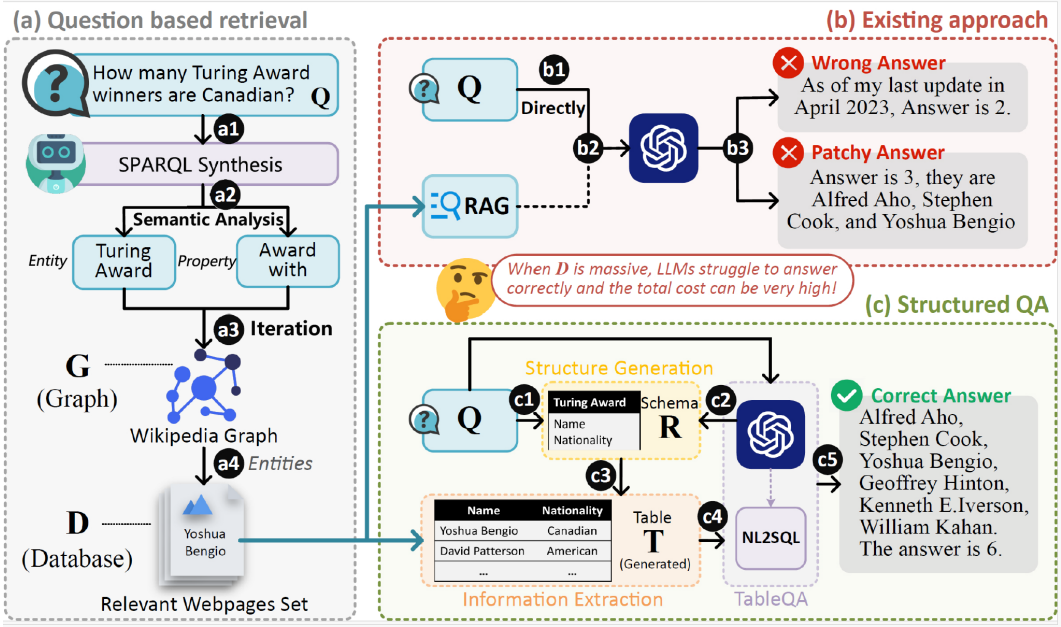

図 1: SRAG の全体像と従来手法との比較 (論文 Figure 1 より).

図 2 は, SRAG の処理をより具体的に 3 つのステップで示したものである. この論文では, スキーマ生成や SQL 生成といった高度なタスクには ‘GPT-4‘ (OpenAI) を, ドキュメントからの情報抽出のような比較的単純なタスクには ‘Mistral-7B‘ (Mistral AI, 7B parameters) のような軽量なモデルを使い分けることで, 効率と性能を両立させている.

- **Step 1: Documents Collection**: 質問のトピック (例: ACM Fellow) と必要な属性 (名前, 所属など) を特定し, Knowledge Graph (Wikipedia) から関連ドキュメントを収集する.
- **Step 2: Information Extraction**: 収集したドキュメントから必要な情報を抽出し, テーブル形式でデータベースに格納する.
- **Step 3: Question & Answer Generation**: ユーザーの質問を SQL クエリに変換し, Step 2 で作成したデータベースに問い合わせることで最終的な答えを得る.

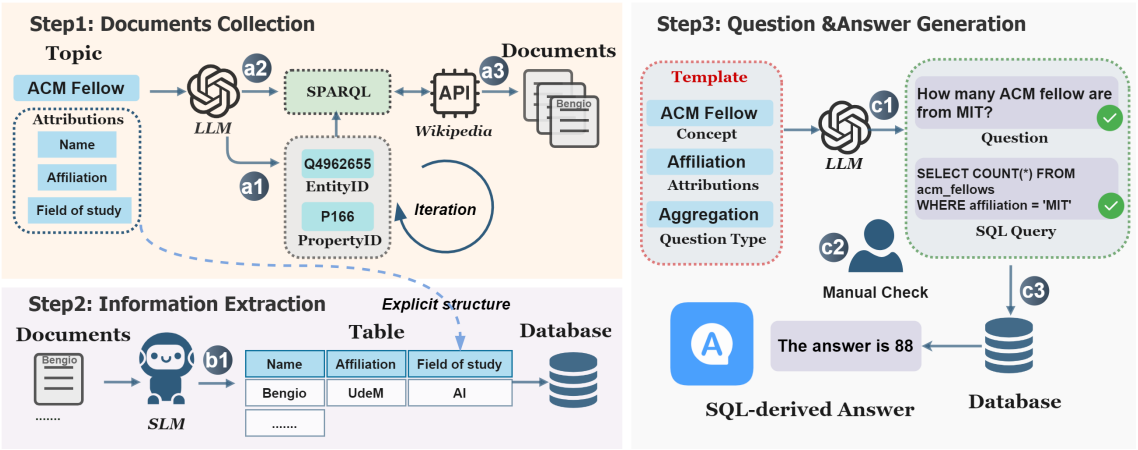

図 2: SRAG の具体的な処理ステップ.

3 今後の課題

- SRAG の実装

参考文献

[1] Teng Lin, Yizhang Zhu, Yuyu Luo, and Nan Tang. Srag: Structured retrieval-augmented generation for multi-entity question answering over wikipedia graph. 2025.