

要求獲得における質疑応答プロセスの グラフデータベースを用いた支援

M2021SE004 今堀由唯

指導教員：佐伯元司

1 はじめに

ソフトウェア開発では要求仕様書の作成過程で要求分析者（以下、分析者）が顧客やユーザなどのステークホルダとの質疑応答を行うことでステークホルダの要求を集めて分析者が要求リストを作成していく。IPAは要求定義に求められる3つのことの1つに「要求仕様を「抜け」「漏れ」「あいまい」なくシステム開発につなげること」[1]を挙げている。

その際、未議論事項や未承諾事項などが残っていると、要求リストを正確かつ明確に作成することが不可能となる。また、分析者が未承諾事項などの質疑応答履歴をすべて把握しておくことは、システム等の規模が大きくなればなるほど困難であるという問題がある。

そのため、未承諾事項などの議論の内容を保持するシステムが必要である。これまで、発話行為論などを用いて質疑応答履歴を構造化し蓄積し活用しようとする研究[2][3]があるが、質疑応答から生じる要求リストの低品質化などの対処支援は十分に行われておらず、蓄積した履歴の有用な活用法が問題である。

本研究では質疑応答履歴を蓄積し、その履歴を活用することにより要求リストの作成を支援するシステムの実現を目的とする。

本論文では2章でアプローチ、3章で提案手法で使用されるグラフデータベースの設計、4章で品質低下の特徴を検出するクエリの設計について述べる。5章で事例適用し、6章で評価と考察を行い、7章でまとめと今後の課題を述べる。

2 アプローチ

要求リストを作成する際質疑応答プロセスでの話し合いの内容を要求リストにする方法が従来では取られてきた。本研究では従来の方法に質疑応答プロセスでの話し合いの内容を質疑応答履歴として蓄積しグラフデータベース[4]を作成する。グラフデータベースの検索や操作機能を使うことにより、議論の抜け漏れや未承諾事項の確認などが可能となり質向上が期待できる。図1にアプローチを示す。

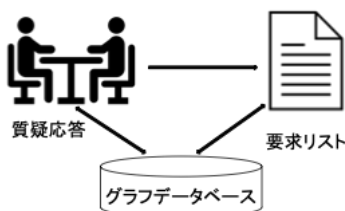


図1 アプローチ

また、質疑応答を重ねると要求リストも更新されていく。そのため、要求リストの更新の把握だけでなく版管理をする必要がある。版管理をすることで新たに要求リストに加わったものや削除された要求リストの把握が可能になる。

質疑応答履歴を構造化して格納するためにグラフデータベースであるNeo4j[5]を使用する。Excelなどの表形式での管理と比較し、グラフデータベースを使用することで動的な更新に柔軟に対応すること、グラフ表示による履歴の可視化などが可能となる。

3 グラフデータベース設計

グラフモデルは加藤らの研究[6]での質疑応答モデルを分析し定義した。この文献は図書館の業務を支援する情報システムで、要求分析者（以下、分析者）が図書館の利用者（以下、利用者）、図書館員と質疑応答を行い、初期要求リストを詳細化し、更新していくプロセスである。

グラフデータベースを使用することで解決したい問題を以下に示す。

- (1) 全体を把握している人がいない
- (2) 要求の実装漏れが発生する
- (3) ステークホルダとのコミュニケーション不足
- (4) 合意形成ができない、時間がかかる
- (5) ドキュメントに不整合が発生する
- (6) 質疑応答が収束しない
- (7) 完了判断ができない

これらの問題を解決するためのグラフモデルの定義を行い、それによってデータ形成を行い、グラフデータベースを構築していく[7]。

3.1 グラフモデル定義

グラフデータベースを設計するにあたり、質疑応答プロセスをクラスをノード、関連をエッジ、属性をプロパティと自然な対応がつくことなどの理由からオブジェクト指向を使ってモデル化した。グラフモデルを図2に示す。システムをメールと質疑応答の2つに分ける。メールのノードにはメールが1件ずつ格納されている。質疑応答のノードはそれぞれの話題番号で分かれており、図書館員や利用者、分析者などのステークホルダは要求のノードに対して質問、確認、回答の活動を行う。

3.2 データ形成

3.1節で示したグラフモデル定義に従ってグラフデータベースを構築する。図3に実際に使用したメールの一部を示す。

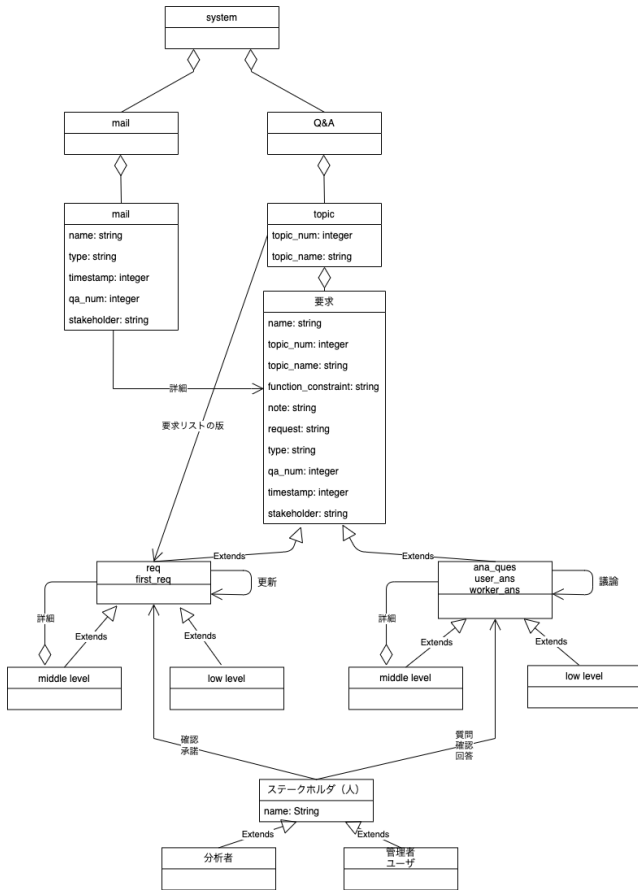


図2 グラフモデル

このメールは2020年05月26日15:40に分析者から送られた質疑応答番号が4のメールである。

<p>2020年05月26日15:40に 分析者から送られた質疑応答番号が4のメール</p> <p>2. 「図書館への資料の登録と削除」についての質問 これは 「貸出可資料を図書館システムに登録する」 「貸出可資料を図書館システムから削除する」 ということでしょうか。</p> <p style="text-align: center;">省略</p>
--

図3 メールの一部

このメールは分析者から確認のメールのためノードのラベルは ana_ques となる。「貸出可資料を図書館システムに登録する」を例に説明する。「2. 「図書館への資料の登録と削除」についての質問」とあることから name がメールの1つの要求である“貸出可資料を図書館システムに登録する”，topic_num が“2”，topic_name が“資料の貸出しと返却”となる。function_constraint が“機能”，request は function_constraint が“機能”のためプロパティを持たない。function_constraint が制約でも特定の要求の記述がない場合はプロパティを持たない。type は「図書館への資料の登録と削除」についての質問とあ

ることから“質問”となる。timestamp はメールの送信日時である12桁の数字“202005261540”，qa_num は質疑応答番号である“4”，stakeholder は分析者からのメールであるため“分析者”となる。

このメールをグラフデータベースに格納すると図4となる。これはNeo4jの画面キャプチャをとった図である。

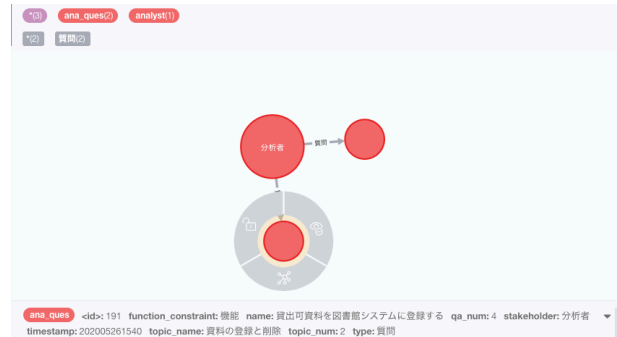


図4 ノードの繋がり

分析者のノードの他に、赤色のノードを選択すると、画面下部に選択したノードのプロパティが表示される。name が「貸出可資料を図書館システムに登録する」であるため分析者から質問のエッジで結ばれた「貸出可資料を図書館システムに登録する」という要求が作成されたことがわかる。もう1つのノードは「貸出可資料を図書館システムから削除する」である。ノードはプロパティの type が質問であるため分析者からグレーの質問のエッジで関係を持つ。

4 クエリ設計

3章で列挙した問題を解決するために、クエリを設計していく。設計したクエリは

- 話題ごとの質疑応答を確認するクエリ (1)
- 議論や要求の抜け漏れを抽出するクエリ (2)
- 未承諾事項を抽出するクエリ (2)(4)(6)(7)
- 話題の手戻を抽出するクエリ (6)
- ステークホルダの発言回数、量を確認するクエリ (3)
- 議論の進捗状況を確認するクエリ (4)(6)(7)
- 指定したワードを検出するクエリ (5)
- 要求リストの版管理をするクエリ (1)
- 要求リストから特定の要求を抽出するクエリ
- 話題番号が変更された要求リストを探すクエリ (5)
- 質疑応答の内容が含まれたメールを探すクエリ

である。3章で列挙した問題の解決に対応する数字を () で示している。例えば、話題ごとの質疑応答を確認するクエリは、3章の (1) の問題解決に役立つ。本論文では、質疑応答の抜け漏れを抽出する検索クエリ、要求リストの版管理をするクエリの説明を行う。

4.1 議論や要求の抜け漏れを抽出するクエリ

質疑応答をする際に議論に取り上げられなかったもの及び議論にはなったものの要求リストにならずに放置されている要求を抽出するためにこの検索クエリを作成し

た. この検索クエリから初期要求や途中で出てきた要求から要求リストに採用されなかったものを検出する. 初期要求の抜けを確認する際はラベルを first_req に議論になったものの要求リストに採用されなかったものを検出する際は ana_ques にすることで検出が可能になる. 検索クエリは

クエリ 1 抜け漏れを検出するクエリ

```
1 MATCH (n:ラベル)-[*]->(m)
2 where not (m:req)
3 RETURN n;
```

で検索する.

1行目の match 文で探したいラベルのノードとそのノードと繋がりがあがる全てのノードをエッジの向きを指定した状態で指定しそこから2行目で要求リストのラベルが付けられたノードがあるかどうか探し出し, 要求リストのラベルがないものを return 文で出力するクエリである.

4.2 要求リストの版管理をするクエリ

質疑応答を繰り返すことで要求リストは更新されていく. そのため要求リストの版管理ができる検索クエリを作成した. エッジに確認したい要求リストのバージョンを指定することで確認が可能となる. 検索クエリは

クエリ 2 要求リストの版管理

```
1 match (n)-[:要求リストのバージョン]->(m)
2 return n,m;
```

で検索する.

5 事例適用

加藤らの研究 [6] で使われた図書館システムを作成する際の質疑応答を使い, 3章, 4章をもとにグラフデータベースを構築する. 質疑応答が行われた期間は2020年5月25日から2020年8月3日までの96件のメールのうち話題1, 話題2を使用した. 全体のグラフを図5に示す.

中心にあるサイズの大きいノードがステークホルダを表し, 緑色のノードが要求リスト, 黄緑色のノードが初期要求, 水色のノードがメール, 紫色のノードが話題番号, 赤色のノードが分析者の質問, 確認の要求のうち要求リストにならなかったノード, オレンジ色のノードが利用者の回答, 黄色のノードが図書館員の回答を表している.

5.1 議論や要求の抜け漏れを抽出するクエリ

4.1節で作成した検索クエリのラベルを ana_ques に指定して検索すると, 図6のグラフを得る.

このグラフでは分析者から質問や確認のあった要求から議論を確認し要求リストとして採択されなかったもののグラフを出力する. 図中のノードを選択するとその情報が得られ, プロパティの値から漏れのあった「機能」や「制約」がわかる. 今回の検索では漏れに対する対処として議論になったものの要求リストに採用されなかったノード

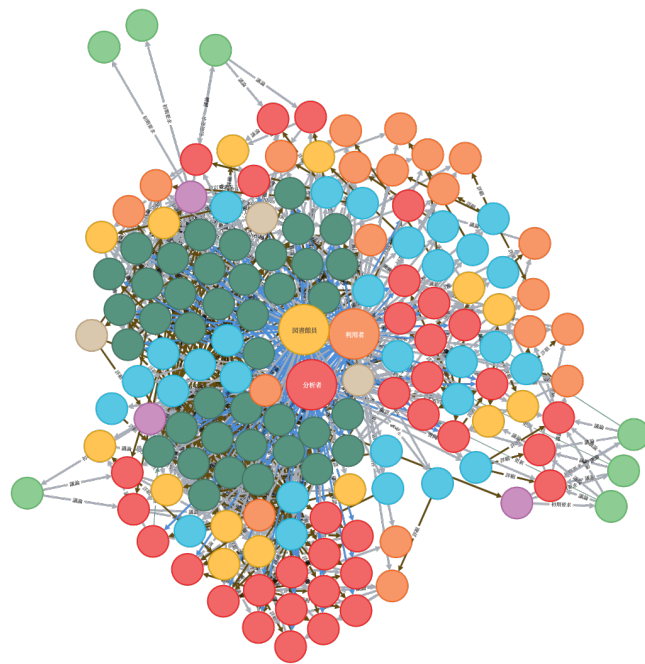


図5 図書館システムの質疑応答

ドを出力するグラフを作成したが, ラベルを first_req にすることで初期要求から要求リストにならなかった議論を探すため抜けに対する対処が可能になる.

5.2 要求リストの版管理をするクエリ

4.2節で作成した検索クエリのエッジを改訂要求リスト_20200608で検索すると, 図7のグラフを得る.

これは初期要求から次の改訂版の要求リストである. 2つのまとまりに分かれているのは話題が2つであるからである. まとまりの中心にある紫色のノードは話題番号のノードである. ノードをクリックすることにより要求の内容が表示される.

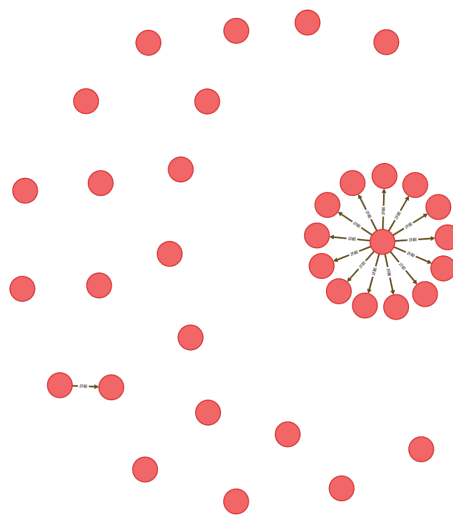


図6 要求リストから漏れた要求を表すノード

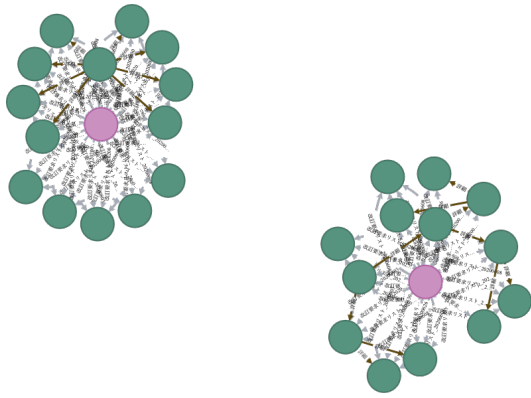


図 7 版管理

6 評価と考察

4, 5章で示した検索クエリと事例適用を使い評価を行う。

6.1 グラフデータベース

4章で作成したクエリは抽出したい情報以上の情報が出力されたクエリが3件あったが、全てのクエリで抽出したい要求の抽出が可能となった。抽出したい情報の絞り込みが上手く行かなかった原因として、自由記述で記入した内容を要求リストのノードにしているためプロパティの値が一般化されず、検索クエリでの絞り込みが上手くいかないといったことが挙げられる。グラフデータベースの構造上の問題と検索クエリの不備の双方に原因があるためこれらの改善は今後の課題とする。

6.2 表計算ソフトとの比較評価

3章で列挙した問題と要求リストを文書にまとめることの8点からExcelシートとグラフデータベースを比較し評価する。評価結果の一部を表1にまとめる。表中の○, △, ×は各々検出できた項目, 検出状況に不備がある項目, 検出できない項目である。

表1 表計算ソフトとの比較の評価

評価項目	グラフデータベース	Excel
全体の把握	○	△
合意形成	○	×
ドキュメントの不整合	△	△
文書記述	△	○

Excelは1枚のシートで情報を項目別に蓄積し、関係はリンクで繋ぐ。そのため、各種のリンクで繋がれた情報を俯瞰的にみることが難しい、平面上に配置された情報が基本のため階層構造をわかりやすく書くことが難しいなどの欠点がある。これらの理由よりグラフデータベースと比較し、質疑応答履歴の「全体の把握」や「合意形成」が取れているか否かの判断等が検索による把握が難しい。

また、「ドキュメントの不整合」はグラフデータベース、Excelともに特定のワードが含まれるかを検索すること

が可能になるため異なる話題番号で似たような要求がないかどうかの確認は可能である。しかし、文章が全体として矛盾しているか否かの判断は難しく、全てのドキュメント不整合に対する検出はできていない。

文書記述はグラフデータベースでは要求リストに記述すべき内容とそれらの内容の階層構造は表現できるが、要求リストの記述順を指定できない。Excelシートでは記述順序、章や節のような階層構造を保持した記述等が可能であるためグラフデータベースよりは有利であると判断した。

7 まとめと今後の課題

本研究では質疑応答履歴を蓄積し、その履歴を活用することにより要求リストの作成を支援するシステムの実現を目的とした。また、質疑応答履歴を構築する際のグラフモデルを必要な検索クエリが抽出可能なように定義し、グラフデータベースの設計、構築を行った。構築したグラフデータベースから質疑応答の特徴を抽出する検索クエリを作成し、それらを事例である図書館システムの質疑応答履歴に適用した。今後の課題を以下に示す。

- グラフデータベース全体の構築
- 検索クエリで表現できることの充実
今回作成した検索クエリでは情報を絞り込めないなどの課題がいくつか残った。それらを解決するための検索クエリの作成を行う。
- 実際の質疑応答への適用
- ユーザインタフェース
ユーザインタフェースを整えてグラフデータベースの構築、検索クエリの実行を容易にする。
- 自然言語処理技術などを用いた会話の分析
質疑応答の同じ議論の繰り返しや矛盾点の検出を自然言語処理技術等を用いて行う。

参考文献

- [1] ユーザのための要件定義ガイド 第2版 要件定義を成功に導く128の勘どころ, IPA, 2019.
- [2] Colin Potts, et al., Inquiry-Based Requirements Analysis, IEEE Software, 11(2), pp.21-31, 1994.
- [3] Motoshi Saeki, et al., Structuring utterance records of requirements elicitation meetings based on speech act theory, ICRE 1996, pp.21-30, 1996.
- [4] I. Robinson, et al., Graph Databases: New Opportunities for Connected Data, 2nd ed., O'Reilly, 2015.
- [5] Neo4j, <https://neo4j.com/>.
- [6] 加藤 潤三 他, 要求獲得のためのシソーラスの統合, 情報処理学会研究報告ソフトウェア工学 (SE), 2021-SE-208-4 8 ページ, 2021.
- [7] 今堀 由唯 他, 要求獲得における質疑応答履歴のグラフデータベースシステムの実現, FOSE2022, pp.123-128, 2022.