大規模言語モデルを用いたソフトウェア仕様書の用語チェック

金井 健一郎[1] 安部 夏樹[1] 加羽澤 優[1] 内出 隼人[1] 斉藤 辰彦[1]

[1]三菱電機株式会社 情報技術総合研究所
{Kanai.Kenichiro@dy, Abe.Natsuki@dn, Kabasawa.Yu@ak, Uchide.Hayato@dy,
Saito.Tatsuhiko@db}.mitsubishielectric.co.jp

概要

本研究では，ソフトウェア仕様書の品質向上を目的に，大規模言語モデル(Large Language Model, LLM)を活用した自動レビュー手法を提案する．本手法ではRetrieval-Augmented Generation (RAG)を用いて，レビュー対象に関連する情報を取得し，レビュー観点毎に自動レビューを実行する．用語統一をレビュー観点とした実験を行った結果，性能に課題が残るものの，正しく指摘できる例もあり，知見が得られた．また，レビューの性能向上にはプロンプト内の表の形式を統一させることやレビュー対象に関連する情報を必要な情報に絞ることが重要であることが示唆された．しかし，他にも性能が低い要因があると考えられ,要因の特定は今後の課題である．

1 はじめに

ソフトウェア(以下, S/W)開発プロセスの一つであるウォーターフォール型開発[i]では，各フェーズで埋め込まれた不具合が後のフェーズに流出すると，手戻りが発生し，費用面／工程面で大きな損失につながる．そのため，埋め込まれた不具合はそのフェーズで発見することが原則である．

各フェーズで埋め込まれた不具合を発見するための作業はレビューと呼ばれる．レビューは設計ドキュメントなどの成果物を関係者で確認し，不具合やその他不備を発見するとともに，各フェーズにおける成果物の品質を評価する作業である．そのため，網羅的な観点で一定の作業品質でレビューを実施することが，各フェーズにおける成果物の品質向上のために重要である．

設計ドキュメントの中でもS/W仕様書は, S/W設計フェーズの成果物であり，S/Wを規定するドキュメントである.したがって,S/W仕様書の品質はS/Wの品質に大きく影響を与えるため，S/W仕様書のレビューも重要である．しかし，S/W仕様書のレビューでは複数の文書を参照する必要があり，コストや難易度が高い．参照する文書は例えばシステム仕様書やインタフェース仕様書,規約など多岐にわたる．

そこで本研究では，S/W仕様書のレビューを一定の品質で行い，S/W品質の向上を目指す，LLMを用いた自動レビューの第一歩として，RAG [1]により複数の文書から抽出した情報をLLMに入力し,S/W仕様書の用語をチェックする手法を提案する．

2 関連研究

近年，LLM はテキストの評価タスク[2]，[3]に応用されており，S/W開発におけるドキュメントレビューへの応用可能性も示唆されている．

表形式で書かれた S/W 仕様書のレビューの自動化を行った研究 [4]では，LLMに表を正確に認識させる手法が検討されている．この研究によれば，表の見出しと値を区別できる形式であるMarkdown形式やJSON形式にすること，及び，表形式の中に自然文が多い場合はMarkdown形式，記号表現が多い場合はJSON形式でLLMに入力することで，LLMによる仕様書内の不具合検出率が向上することが示されている．また，この研究の中ではレビューの観点を整理し，プロンプト内に具体的にチェックすべき項目を与えることも提案している．

[i] ウォーターフォール型開発：ソフトウェア全体をシステム設計，ソフトウェア設計，プログラム設計，コーディング，単体試験などのように徐々に細部を設計し，製作・試験を進めていく開発手法.ここに書いたシステム設計などをフェーズと呼ぶ.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

表 1　レビュー観点例

副特性分類	レビュー観点
体裁	用語が定義され統一されているか.
機能完全性	システム機能の中に，ソフトウェア機能にて言及されていない機能がないか.
機能正確性	ソフトウェア機能が使用するデータの精度はシステム機能の規定する精度を満たしているか. ソフトウェア機能が実行する処理内容はシステム機能とあっているか.

S/W 仕様書のレビューへの適用を前提とした情報検索手法を検討した研究 [5]では，LLM を用いた情報抽出について研究されている. この研究によれば，質問に関連する情報を検索するに際し，text-embedding-ada-002 による埋込ベクトルのコサイン類似度を用いて検索された情報をさらに GPT-3.5-turbo-16k を用いて取捨選択することで recall@1 が大きく向上することが示されている.

さらに， LLM に図を認識させる手法についての研究 [6]では，段階的に Chain-of-Thought（CoT） [7]を用いて認識させる手法を提案している．この手法では，図を PlantUML に変換するために，図の認識過程をノードの認識，ノードの形状認識等のいくつかのタスクに分割し，CoT を用いて段階的にこれらのタスクを GPT-4o に解かせることで精度の向上を図っている.

3 提案手法

3.1 レビュー観点

本研究では，先行研究 [4]にならいレビュー観点の整理から実施した．レビュー観点は，システム／ソフトウェア製品品質 [8]に規定される品質副特性の中から S/W 仕様書で考慮すべき項目に絞った項目と体裁や用語の統一といった項目に対して，レビューをする際に確認する具体的な観点を設定した．そのうえで，S/W 仕様書において自動でレビューをしてほしい観点について社内の要望を確認し，優先度をつけた．優先度が高いレビュー観点の例を表 1 に記載する.

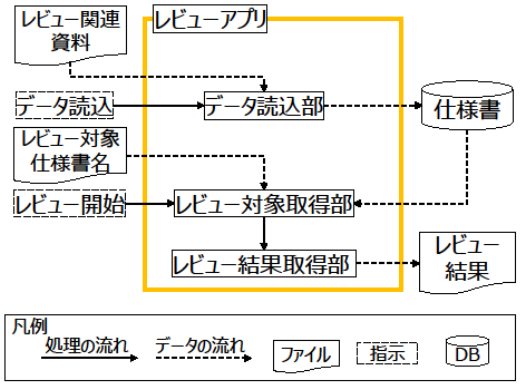

図 1 アプリケーション構成

3.2 手法

仕様書内にある記述のレビューをするには複数の仕様書にある関連する記述を参照し，レビュー対象の記述との整合性を確認する必要がある．本研究では，レビュー対象の記述に関連する記述の取得を RAG で実現した. 本研究で使用したアプリケーションの構成を図 1 に記載する．図 1 の通り，本アプリケーションは次の部位からなる.

データ読込部　データ読込指示により，レビュー対象の仕様書を含むレビュー関連仕様書を読み込み，後述のチャンク分割を施し仕様書 DB へ格納する部位．以下，読込部とする.

レビュー対象取得部　レビュー開始指示により，指定されたレビュー対象仕様書名に合致する仕様書のチャンクをすべて仕様書 DB から取得し，すべてのレビュー対象のチャンク内の文を検索クエリとして，Azure AI Search のハイブリッド検索を用いて関連する記述を取得する部位．以下，対象取得部とする.

レビュー結果取得部　すべてのレビュー対象のチャンクについて，レビュー観点並びに，対象取得部が取得したレビュー対象の記述とレビュー対象に関連する記述をプロンプトに設定し, LLM にレビューを指示し, LLM の出力結果をファイルに保存する部位．　以下，結果取得部とする.

なお, 表については PDF 内の表はテキストとしてそのまま読み込み，その他ファイル内にある表は TSV 形式で LLM に入力することとした．また，図については，PDF は文字列として認識できる部分はテキストとして LLM に入力し，その他のファイル内にある図は無視した．S/W 仕様書には様々な図が

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

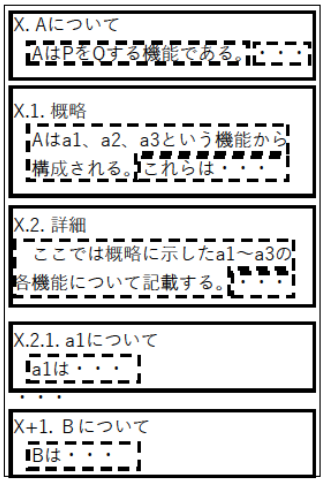

図 2 チャンク分割イメージ図　全体が仕様書のあるページを表しており, X 章及び X+1 章が記載されている. 実線の四角が章／節構造単位のチャンクであり, 破線の四角が文単位のチャンクを表す.

存在するため，図はレビューをする上で重要な要素である．図をレビューできるようにすることは今後の重要な課題である．

4 評価実験

4.1 実験設定

本研究では下記の実験設定でレビューを実行した．実験時に指定した項目及びその内容は下記である．

レビュー対象仕様書　読込部が読み込むレビュー対象とする仕様書．本実験では S/W 仕様書（104 頁）を章／節構造単位にチャンク分割した 73 個のチャンクをレビュー対象とする．62 個の不備を含む．チャンク分割は図 2 も参照のこと．

関連する仕様書　読込部が読み込むレビュー関連仕様書．本実験では，システムの構造を規定するアーキテクチャ仕様書，システムの機能を規定するシステム仕様書，システム内で保存するデータを規定するデータ仕様書，システム内の装置間のインタフェースを規定するインタフェース仕様書の 4 冊を使用した．各仕様書の頁数，チャンク数を表 2 に記載する．なお，チャンク分割については，PDF ファイルは固定長(512 トークン単位，125 トークンオーバーラップ)，その他は章／節構造単位とした．チャンク分割については図 2 も参照のこと．

埋め込みモデル　対象取得部が関連する記述を取得する際のベクトル検索に使用する埋め込みモデル．本実験では text-embedding-3-large を使用した．

表 2　関連する仕様書の概略

仕様書	頁数	チャンク数	ファイル形式
アーキテクチャ仕様書	47	125	PDF
システム仕様書	104	230	PDF
データ仕様書	46	66	その他
インタフェース仕様書	43	98	その他

表 3　検索性能とレビュー性能 ()内の数字は再現率，適合率を算出した実数値．検索精度はチャンク数，レビュー精度は不備数とする．

	再現率
検索	0.56 (19/34)
レビュー	0.19 (12/62)

レビュー観点　結果取得部がプロンプトに設定するレビュー観点．本実験では表 1 内の「体裁」をレビュー観点とした．

使用 LLM　結果取得部がレビューを実行する LLM を指す．本実験では GPT-4 32k を使用した．

4.1.1 評価尺度

評価指標として関連する記述の検索性能とレビュー性能を下記の通り算出した．

検索性能　レビュー対象の仕様書の一部の小節を対象に，その小節に関連する記述を人の目で探したものを正解とし，この正解と RAG の結果検索された関連する記述を比較することで算出する．

レビュー性能　LLM の出力結果の妥当性を人の目で確認するとともに，他に指摘すべき不備がないかを確認し，正解指摘，不正解指摘，未発見不備を数え上げることで算出する．

4.2 定量評価

4.1.1 に記載の評価尺度による評価結果を表 3 に示す．検索における再現率は 0.56 となっているが，レビューにおける再現率が 0.19 にとどまった．これから検索, レビューともに精度が低いことがわかる．

4.2.1 指摘可能な情報を含む場合の評価

検索結果に正解を含む場合のレビュー性能を分析した結果が表 4 である．表 4 から，特にレビュー対象の表と関連する記述の表を比較して指摘すべき不備の再現率が悪く不備を発見できていないことがわ

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

表 4　検索結果が正しい場合のレビュー指摘再現率 ()内の数字は分母が発見すべき不備数，分子が発見できた不備数を表す．

		レビュー対象	
		表	本文
関連する記述	表	0.00 (0/33)	－ (0/0)
	本文	－ (0/0)	1.00 (6/6)
	表+本文	－ (0/0)	1.00 (2/2)

かる．これは，PDF ファイルとその他のファイルで LLM に入力する表の形式が異なるため，LLM が表を比較できなくなっているためと考えられる．

4.2.1 関連記述の内容とレビュー精度

関連する記述の内容，特に表の形式がレビュー対象との表の形式に一致しているか否かのレビュー性能への影響を明確にするため，追加実験を行った．

追加の実験では，レビュー対象と関連記述の表同士を比較して発見すべき不備 33 個の中から無作為に 16 個の不備を選定した．そして，このチャンクを対象に，関連する記述を下記の 4 パターンで変えて，レビュー性能を確認した．結果を表 5 に示す．

パターン①　メインの実験から変更なし．

パターン②　表の形式をレビュー対象の表の形式と一致させる．

パターン③　表の形式は変更せずに，内容を不備の発見に必要な記述に絞る．

パターン④　表の形式をレビュー対象の表の形式に一致させるとともに，内容を不備の発見に必要な記述に絞る．

この結果は，表の形式を一致させただけではレビュー精度は上がらず，関連する記述の内容を指摘に必要な情報のみに絞ることも重要ではあることを示唆する．しかし，パターン④であっても不備の再現率は低いため，他の要因を調査する必要がある．これは今後の課題である．

4.3 定性評価

不正解であった指摘内容では，そこまで細かく指摘する必要がないと感じる指摘や，レビュー対象の表が理解できていない指摘が多かった．正解例も含めた具体的な内容を Appendix B に記載する．

表 5 関連する記述の内容とレビュー指摘再現率

パターン	再現率
①変更なし	0.06
②表の形式一致	0.03
③必要な記述に絞る	0.18
④表の形式一致かつ必要な記載に絞る	0.53

5 課題

課題は大きく 4 つある．

1 つは PDF 内の表とその他ファイルの表の形式を一致させる手法の検討である．PDF の表読み込み機能は Python の OSS にもあるが，全形式のファイルで同一の TSV 形式のテキストにする手法など検討が必要である．OCR を用いた表の読み込みも検討が必要である．

2 つ目は関連する記述の検索精度の向上である．これには，RAG におけるインデックス化，クエリ調整，検索，Reranking などの各手法を改善する必要がある．さらに，Agent を用いてレビューに必要な情報を LLM に取得させる手法も考えられる．これらの手法を検討し，関連する記述の再現率，適合率の向上を図る．

3 つ目は関連する記述の長さと表の形式以外のレビュー性能へ影響調査である．これはレビュー対象チャンクの長さなども影響があると考えられるため，今後実験を進める．

最後は図の読み込みである．S/W 仕様書には様々な角度から見た S/W 構造が図として記載されている．図をレビューできれば S/W の構造上の不備を発見でき，S/W 仕様書の品質を向上することができる．図の読み込みは参考文献[6]にて提案されている手法などをもとに検討を進める．

これら課題の検討を進め，LLM による S/W 仕様書のレビュー精度の向上を目指す．

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

参考文献

1. Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems (NeurIPS 2020)*, 2020.

2. Zheng, Lianmin, Chiang, Wei-Lin, Sheng, Ying, Zhuang, Siyuan, Wu, Zhanghao, Zhuang, Yonghao, Lin, Zi, Li, Zhuohan, Li, Dacheng, Xing, Eric P., Zhang, Hao, Gonzalez, Joseph E., Stoica, Ion. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems (NeurIPS 2023)*, 2023.

3. Liang, Weixin, Zhang, Yuhui, Cao, Hancheng, Wang, Binglu, Ding, Daisy, Yang, Xinyu, Vodrahalli, Kailas, He, Siyu, Smith, Daniel, Yin, Yian, McFarland, Daniel, Zou, James. Can large language models provide useful feedback on research papers? A large-scale empirical analysis. arXiv, 2023. https://arxiv.org/abs/2310.01783.

4. 福田貴三郎，中川尊雄，宮崎桂輔，徳本晋．大規模言語モデルを活用したソフトウェア設計自動レビュー手法の検討．研究報告ソフトウェア工学（SE），2023-SE-215(7), 1-8 (2023-11-23)，2188-8825.

5. 宮崎 桂輔．ソフトウェア設計書群を外部知識源とする RAG のための 検索性能改善の検討．研究報告自然言語処理（NL），2024-NL-259(18),1-7 (2024-03-03) , 2188-8779

6. 福田貴三郎，徳本晋，藤本博昭，小田嶋成幸．大規模言語モデルを活用したダイアグラムを含むソフトウェア設計書の自動レビュー手法の検討．ソフトウェアエンジニアリングシンポジウム 2024 論文集, 2024, 319-320（2024-09-10）

7. Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. 2023. https://arxiv.org/abs/2201.11903

8. International Organization for Standardization and International Electrotechnical Commission. 2011. ISO/IEC 25010: Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models. Geneva: ISO/IEC.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).

Appendix A プロンプト

今回使用したプロンプトは下記の通りである.

「あなたはとても優秀なソフトウェア設計のレビュアーであり、あなたの指摘は一貫しており、正確である。次のレビュー対象はソフトウェア仕様書であり、上位仕様となる関連記述と比較し、次の観点でレビューせよ。指摘はレビュー対象に関する指摘とし、関連記述に対する指摘はしてはならない。
指摘がある場合は、どの記述がどの記述と矛盾しているのかを明確に記載せよ。もしレビュー指摘がない場合は、指摘なしとのみ答えよ。

###レビュー観点:
{観点が入る}

###レビュー対象:
{レビュー対象が入る}

###関連記述:
{関連記述が入る}」

Appendix B レビュー結果の具体例

本研究で LLM が出力したレビュー結果の抜粋をレビュー対象の記述と関連記述とともに記載する.

正解指摘例

レビュー対象（全 56 文字）:「6.XXX データ管理基盤アプリの処理¥n 本章では、XXX データ管理基盤アプリを構成する(略)」

関連記述（全 679 文字）:「1. 概要¥n 本仕様書は、YYY-XXX データ管理基盤(ZZZ)の開発における...(中略)...本仕様書では、YYY 機能開発(XXX データ管理基盤)の開発における...(中略)...YYY 機能開発(XXX データ管理基盤)に対して...(中略)...XXX データ管理基盤...(略)」

レビュー出力:「###指摘:
レビュー対象の"XXX データ管理基盤アプリ"と関連記述の"YYY-XXX データ管理基盤(ZZZ)"、"YYY 機能開発(XXX データ管理基盤)"の表現が一貫していない。これらが同じ機能やアプリケーションを指しているのか、それとも異なるものを指しているのかが明確でない。これにより混乱を招く可能性がある。この用語の使用法を統一するべきである。」

正解：レビュー対象内にある「XXX データ管理アプリ」という言葉が関連記述内になく，未定義，用語統一がされていない.

不正解指摘例　人からすると指摘しなくても良い

レビュー対象（全 167 文字）：「7. データ管理基盤アプリ ¥n 7.2. データ投入機能 ¥n7.2.2. 情報編集処理 ¥n ログイン画面より、情報登録の権限を有するユーザーでログインした後、情報作成メニューより、情報編集を選択することで起動する。情報編集処理を構成する画面は次の通りである。（略）」

関連記述（全 2060 文字）：「(略)...No ¥n 部品 ¥n 説明 ¥n...(中略)...¥n③ ¥n 情報編集ボタン ¥n 情報一覧画面を表示し、情報の登録操作を行う。 ...（略)」

レビュー出力:「レビュー対象の...（中略)...で「ログイン画面より、情報登録の権限を有するユーザーでログインした後、情報作成メニューより、情報編集を選択することで起動する。」と記述されていますが、関連記述(中略)を見ると、「情報編集ボタン」を押すことで「情報一覧画面」が表示され、そこから情報の登録操作を行うとされています。¥n¥n これらの記述を比較すると、「情報作成メニューから情報編集を選択する」か、「情報編集ボタンを押して情報一覧画面を表示する」かで使用法が統一されていない用語（「情報編集」）が存在しています。これは、ユーザーがどの操作を行えば「情報編集」にアクセスできるのか混乱を招く可能性がありますので、使用法を統一すべきです。」

正解：指摘なし

表の形式が異なることにより指摘できなかった例

レビュー対象（全 2256 文字）:「5. XXX データ管理基盤アプリの機能¥n5.1. 機能一覧¥n データ管理基盤アプリの機能一覧は次の通りである。表 7.2.1.1-1 データ管理アプリ機能一覧
"機　能" "機　能" "機　能" "開発時期年度" "開発時期年度" "機　能" "機　能" "機　能" "20XX" "20YY"
"データ閲覧" "表示" "データピン表示" "ー" "ー"...（略)」

関連記述（全 5513 文字）:「(略)...機能
開発時期
20XX 年度
20YY 年度
...（中略)...
情報ピン表示
ー
ー （略)」

レビュー出力:これに関する指摘なし.

正解：レビュー対象内にある「データピン表示」は関連記述内では「情報ピン表示」となっており，用語が統一されていない.

This work is licensed by the author(s) under CC BY 4.0
(https://creativecommons.org/licenses/by/4.0/).