

生成AIを用いた図の記述状況可視化に関する研究

工学院大学 安藤勇輝 指導教員:位野木万里

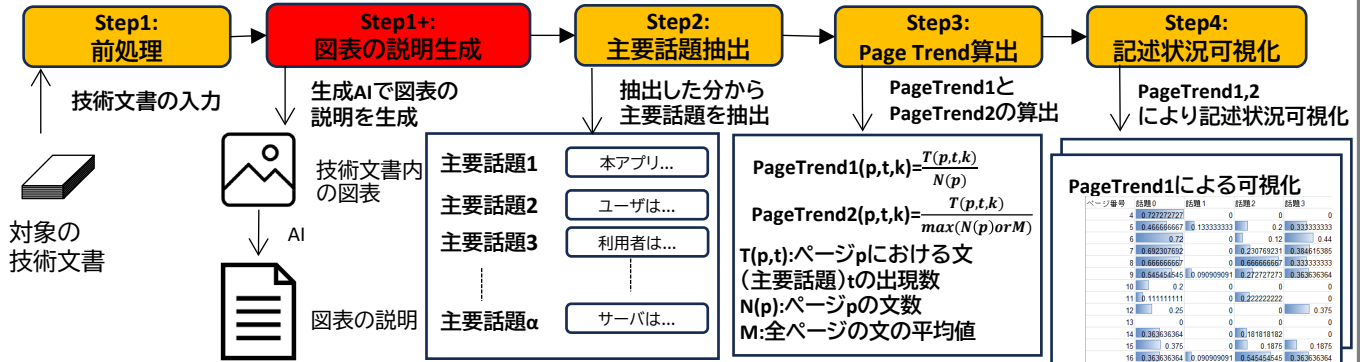
背景

- 記述状況可視化ツール[1]は文書理解支援において一定の効果有
- 従来の記述状況可視化ツール[1]では、図表は対象外という課題

課題へのアプローチ

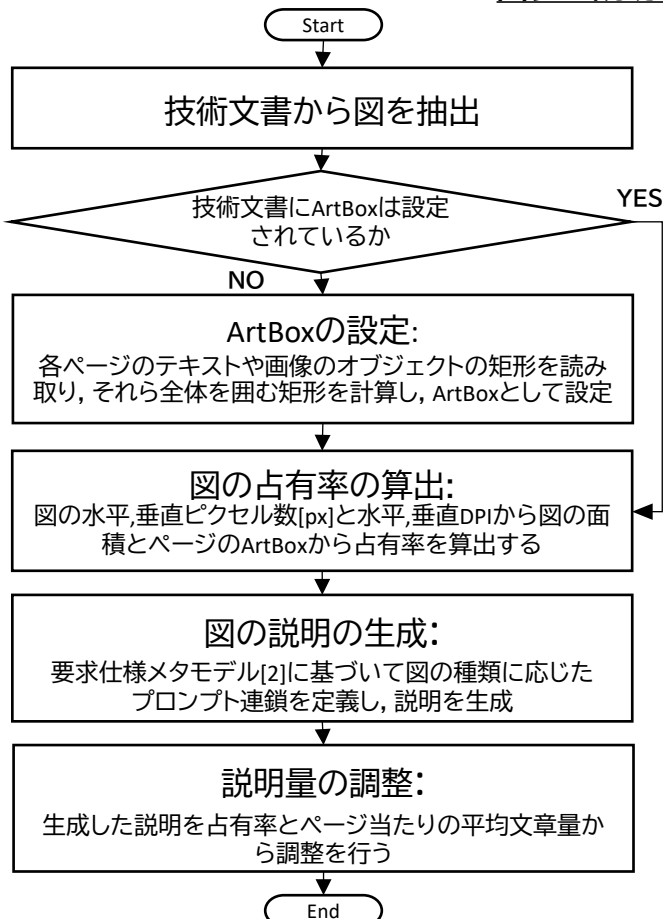
- 図の種類に応じたプロンプト連鎖を設計し、図の説明を生成する
- 生成した説明を図表の占有率に基づき説明量を調整し、記述状況可視化

図を含めたPage Trendによる記述状況可視化ツール



(Step1-Step4は関連論文[1]を踏襲。)

図表の説明生成のフロー



作成ツールについて

- 環境: Google Colaboratory
- ・従来ツールの前処理に組み込む形で作成
 - ・PDF操作にはpythonライブラリPyMuPDFを使用
 - ・占有率は具体的には以下の式で求める

$$\text{Occupy}[\%] = \frac{\text{ImageArea}[\text{in}^2]}{\text{ArtboxArea}[\text{in}^2]} \times 100$$

業務フロー図の説明生成の例

1. 業務フロー図の知識と図を与える
2. 業務内容を特定し、関連情報(関係者や機能)の説明を指示する
3. 登場する各処理の説明を指示する。
4. 入出力情報の説明を指示する。

期待効果

占有率を考慮した説明量に調整することで、技術文書の記述状況を画像を含めた上でより正確に可視化可能効果的な例

- ・図表のみのページが含まる文書
- ・同一話題についての図表だがページをまたぐ文書

今後の課題

- ・作成したツールの適用評価の実施
- ・対応可能な図の種類の拡大
- ・画像以外の形式で保存されている図への対応

[1] 中村雄太郎,長岡武志,北川貴之,位野木万里,本位田真一, Page Trendによる記述状況の可視化を用いた要求仕様書の理解手法の提案とtf-idfによる記述状況の比較評価,日本ソフトウェア科学会「コンピュータソフトウェア」,2024,Vol. 41,No.3, pp.115-121

[2] 北川貴之, 橋本憲幸, 吉田和樹, 位野木万里, "要求定義における暗黙知の形式知化手法", 日本ソフトウェア科学会 コンピュータ ソフトウェア27巻山3号p93-98,2010/08/07

イノベーション創出の場としてのハッカソン： 要求工学・デザイン思考に基づくプロセスモデルの提案

戸上晃希, 上原悠, 位野木万里

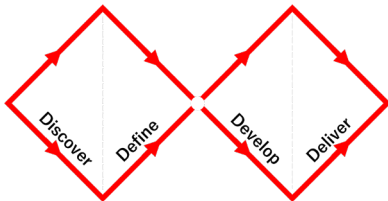
背景

- DXの社会実装に向けて常に新しいアイデアが求められている
- ハッカソンは企業が参加者からアイデアを得てイノベーションに貢献する手法として有効
- ハッカソンでの開発方法は検討の余地が残る
- 要求工学などの体系的な方法論を適切に取り入れることが困難

解決策へのアプローチ

- ハッカソンの事例を分析
 - 課題の再確認
 - 開発プロセスの知見を抽出
- 事例に基づいたプロセスモデルの提案
 - 初級技術者でも質の高いハッカソンが可能
 - プロトタイプの実向上によるイノベーション促進

ダブルダイヤモンド

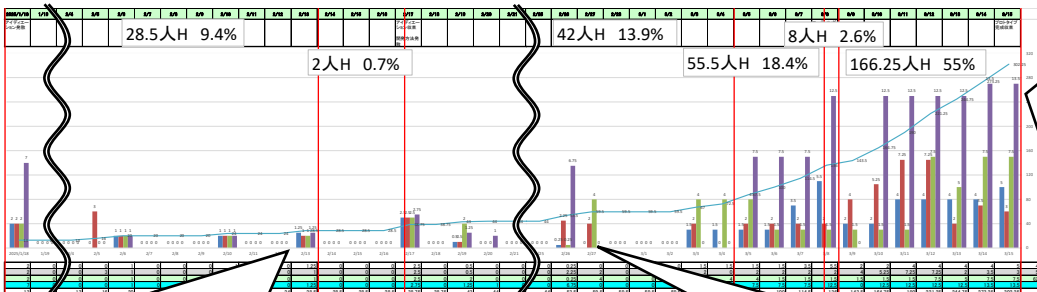


デザイン/イノベーションプロセスのモデル^[1]
発散・収束の繰り返りで課題解決を図る

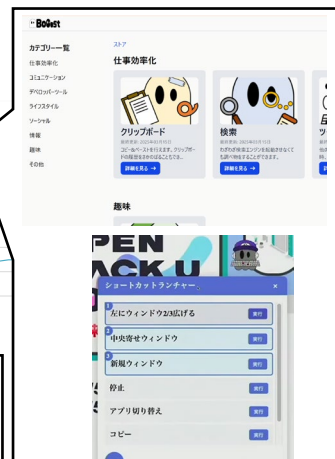
事例分析

- 分析方法
チャットログとメンバーへのインタビューによる振り返り形式で工数を集計・比較
審査員の評価を根拠としてプロトタイプの優れている点について追跡
- ハッカソン事例の情報
Hack U 2025 OSAKA 最優秀賞のプロトタイプとその開発プロセス
リードタイム: 2025年1月18日(土)~2025年3月15日(土)
チームメンバー: 当時工学院大学情報学部3年生4人
審査基準: 新規性・技術性・発展性・再現性の4軸
審査員2名が180秒のプレゼンと自由観覧形式の展示会を評価
制約等: 3/3~3/15のみコーディング可能

縦軸工数・横軸リードタイムの工数一覧表



分析結果



コーディング

メイン/サブ機能開発担当と
専用サイト開発担当に分担
テストは各開発担当の裁量で
逐次実施

アイディエーション

チームメンバー全員でアイデア出しを実施
出たアイデアを技術的実現可能性で整理・投票で決定

要求分析・設計

AsIsのCJMにより要求分析を実施
UI以外の設計は開発担当ごとに実施

	リードタイム	工数(人H)	工数割合
発散(アイデア出し)	01/18~02/13	28.50	9.4%
収束(アイデア決定)	02/14~02/17	2.00	0.7%
発散(要求分析・設計)	02/17~03/04	42.00	13.9%
収束(コーディング)	03/05~03/08	55.50	18.4%
発散(機能再検討)	03/08~03/08	8.00	2.6%
収束(コーディング)	03/09~03/15	166.25	55.0%

10.1%

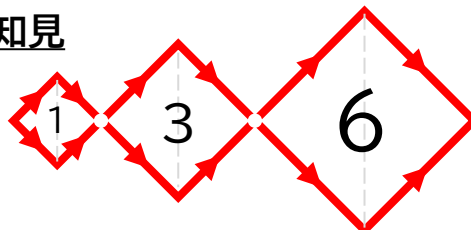
32.3%

57.6%

- 開発プロセスにおける発散・収束の節目を確認した結果「アイディエーションの発散収束」「要求分析・設計の発散と開発(コーディング)の収束」「機能の再検討の発散と開発(コーディング)の収束」であると判明
- 各発散収束間の工数を比較すると工数比率が1:3:6
- 審査員に高評価を得た機能はコーディング途中の機能再検討により発案・実装された

得られた知見

- 開発の途中で反復的に要求検討を行うことが重要
- 工数比率1:3:6が進めると最終的なコーディングに工数をかけつつ反復が可能



今後の展望

- 知見(仮説)に基づき他の事例を同様の形で分析
- 1:3:6が有効な条件の特定
- 開発プロセスで作成すべき中間成果物の特定・具体化

ドメイン特化型メタモデルを用いたコンテンツマッチング手法の提案:カードゲームの分析と比較

南須原大吾, 安藤勇輝, 田中恵美, 横岡歩, 陳家輝, 位野木万里

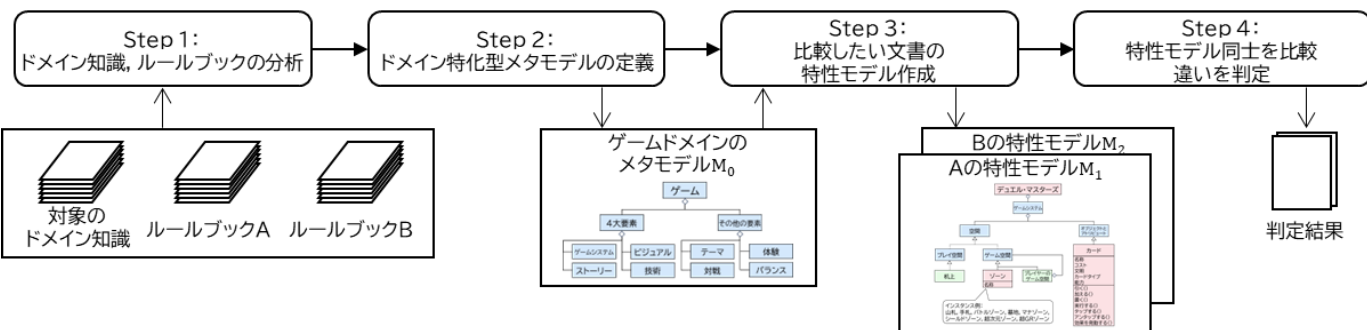
背景

- Digital Transformationの実現が求められており, ユーザの真の要求に応えるために様々な手法やツールが考案
- 既存の要約手法では, 概要は知ることではできてもドメイン固有の特徴をピンポイントで抽出・比較することが困難

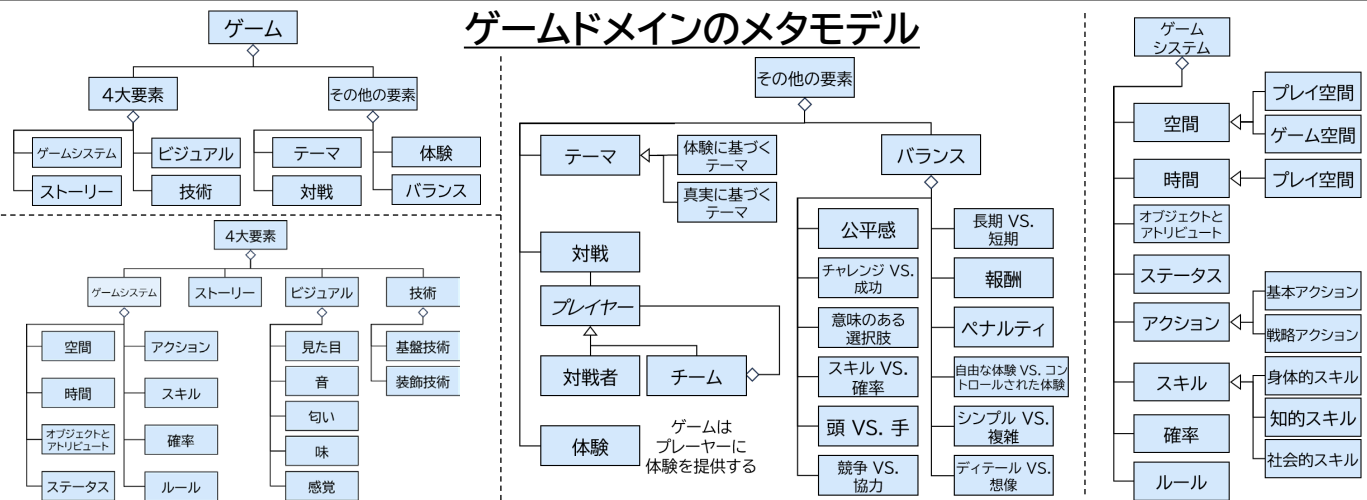
アプローチ

- ドメイン特化型メタモデルを用いたコンテンツマッチング手法を提案
 - 既存の要約手法における課題の再確認
 - ドメイン特化型メタモデルの作成
- ゲームドメインをケーススタディとして実施

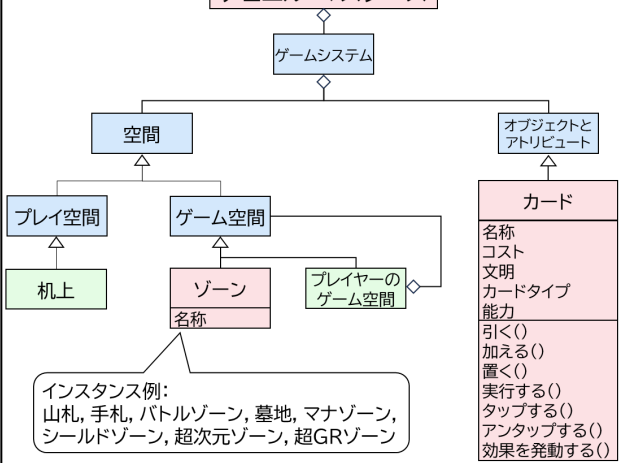
ドメイン特化型メタモデルを用いたコンテンツマッチング手法



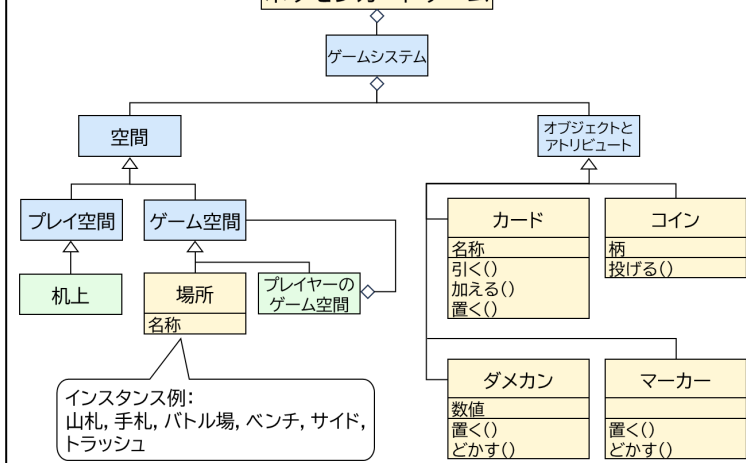
ゲームドメインのメタモデル



デュエル・マスターズ



ポケモンカードゲーム



モデル化によってゲーム空間の構造やカードのサブクラスの種類を明らかにし, 種類ごとのふるまいを比較することでカードゲームごとの特性を把握するきっかけにできる。

デザイン思考に基づくアイデア創出手法の研究: 発想法の組み合わせとCustomer Journey Mappingを用いた アイデア創出手法の提案

上原悠, 戸上晃希, 位野木万里

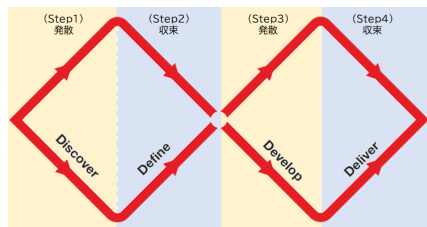
背景

- ・技術の進展や社会的価値の多様化によるアイデア創出機会の増加
- ・慣れた思考に捉われたアイデア創出
- ・グループでアイデア創出を行う際の停滞

解決策へのアプローチ

- ・発想法を組み合わせ、短時間で視点を変えながらアイデアを創出
- ・アイデアを広げるだけでなく、深めるためにCJMを作成

ダブルダイヤモンドプロセス



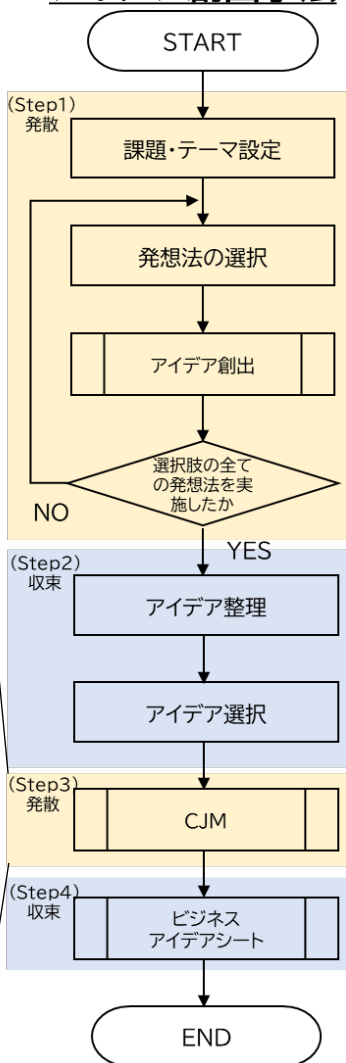
- ・デザイン思考のプロセスモデルの一つ[1]
⇒本研究では発散と収束を2回繰り返すプロセスを経てアイデアを単に発想するだけではなく、チーム内で共通の認識を持つ機会を提供

Customer Journey Map

CJMテーマ	発想段階	課題・テーマ設定	発想段階	課題・テーマ設定	発想段階	課題・テーマ設定	発想段階	課題・テーマ設定	発想段階	課題・テーマ設定
Phase	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定
Action	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定
Opponent	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定
Thinking	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定
Feeling	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定
Insight	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定
その他	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定	課題・テーマ設定

- ・ある人の体験を時間軸に沿って視覚化したもの[2]
⇒本研究では、創出したアイデアのAs-Is, To-Beを記述し、アイデアの課題点やさらなる改善点の洗い出しを実施

アイデア創出手法



発想法

- ・アイデア創出を行う人の学習コストが低くなるよう一般的な発想法を採用

ブレインストーミング

- ・ブレインストーミングの4原則(質より量に集中する, 他の意見やアイデアを非難しない, 自由にアイデアを出す, アイデアを結合・発展させる)を守りアイデア創出
⇒アイデアの素を生み出すことを実施

SCAMPER

Substitute(代用する)	Put to other uses (結合する)
Combine(結合する)	
Adapt(適応させる)	Eliminate(削除する)
Modify(修正する)	Reverse(逆転する)

- ・7つの項目に従ってアイデアを派生
⇒発想の固定化を防ぎ、強制的な発想の転換を実施

エクスカージョン



- ・全く別のものから着想
⇒普段の思考の枠を超えて、アイデア創出を実施

ケーススタディ

- ・参加者: 大学生19名(グループA, B, C)
- ・題材: 持続可能な社会とwell-beingを目指して、未来のまちの豊かさを再定義
- ・方法: グループA 3種類の発想法とCJM
グループB 3種類の発想法
グループC CJMのみ



結果

- ・グループA: いろいろな観点を取り入れながら、途切れることなくアイデアを創出
- ・グループB: 3種類の発想法から創出したアイデアが、同数程度ずつビジネスアイデアへと具体化
- ・グループC: CJMで抽出した課題に対する解決策の数を考える時間が多くありその数は課題の数の約2倍

今後の課題

- ・アイデアの中身の広がり, 多様な視点で考えることができるかについて分析
- ・生成AIの活用による, 発想の支援

文書比較による要求文書の獲得補助 ～2つの文書の違いが可視化できるのか？～

横岡歩, 南須原大吾, 安藤勇輝, 田中恵美, 陳家輝, 位野木万里

背景

- 対象システムが大規模化するに伴い要求仕様書の記述量が増加し複雑化
- 技術文書の理解補助のため要約技術やテキストマイニングの活用されているが、テーマの理解はできても背景や意味など間違った理解を招く可能性
- 記述状況マトリクスを用いて文書比較を行うことで共通性可変性の比較や特定がしやすい点が有効[1]

課題

- 要約技術やテキストマイニング手法では出現頻度の高い表現を把握することが可能だが重要箇所を見逃すリスク有
- 仕様書におけるシナリオでは、同じ機能でも対象となるアクターが異なる場合があるが、要約技術やコンテンツマッチング手法を用いて違いを可視化するのが困難

解決策へのアプローチ

- 記述状況マトリクスによるコンテンツマッチングは共通性や可変性の比較や特定がしやすい点が効果的[1]
- 物語などの文書では、特に視点による分析は重要
- 文書を構成する各文章の視点また話法に注目し、数値を共通しているの結果に加え文書比較

本実験における視点とは

本実験において、文書内の各文章に該当する登場人物をその章の視点として設定することを指す。
視点を分析することで人物に関する重要な情報を際立たせ、解釈上の問題を解消できる[2].

話法とは

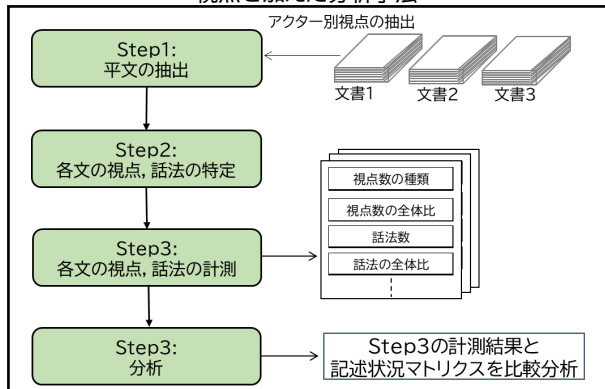
直接話法:登場人物の言葉をそのまま引用し、「」で囲む形で表現する手法。
間接話法:登場人物の言葉を自分の言葉で再構成し、引用せずに表現する手法。
その他:直接話法と間接話法に該当しない文。

視点を加えたコンテンツマッチング手法～多視点で対象文書をコンテンツマッチング～

Page Trendによるコンテンツマッチング手法



視点を加えた分析手法



ケーススタディ

3種類の桃太郎の文書

芥川版	芥川龍之介の作成した鬼を善とした桃太郎
石塚版	桃太郎が鬼ヶ島に向かう船の上での物語
楠山版	楠山正雄によって再話された桃太郎

記述状況マトリクスでは、芥川版と楠山版は共通している

「Page Trendによるコンテンツマッチング手法」の結果と比較

各文書のアクター別視点の種類

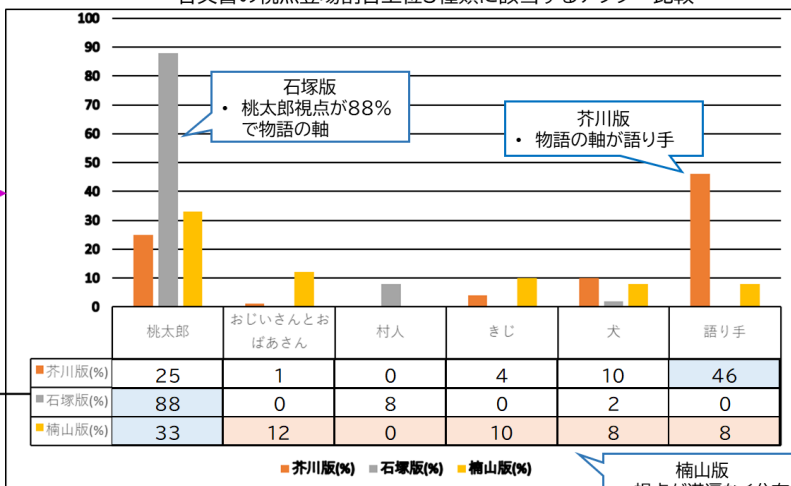
	芥川	楠山	石塚
登場人物数	7	11	5

話法別の各文書占有率

	間接話法	直接話法	その他	計
芥川版(%)	6(6)	23(21)	78(73)	100(100)
石塚版(%)	30(24)	0	93(76)	123(100)
楠山版(%)	0	63(53)	57(47)	120(100)

石塚版
話法 その他0%
視点 語り手0%
から一人称文書である

各文書の視点登場割合上位3種類に該当するアクター比較



今後の課題

- 物語においては視点や話法が明確に存在するが、仕様書やシナリオなどの文書においては手法の詳細化が必要
- 物語以外のシステム開発のシナリオ等の事例を対象に新たにケーススタディを実施

要求仕様書の自動生成エコシステムに向けた 生成AIを活用した複数仕様書間の共通差分抽出手法の提案

横井新大, 高杉柚月, 岡戸隆郁, 島川遼太郎, 陳家輝, 位野木万里

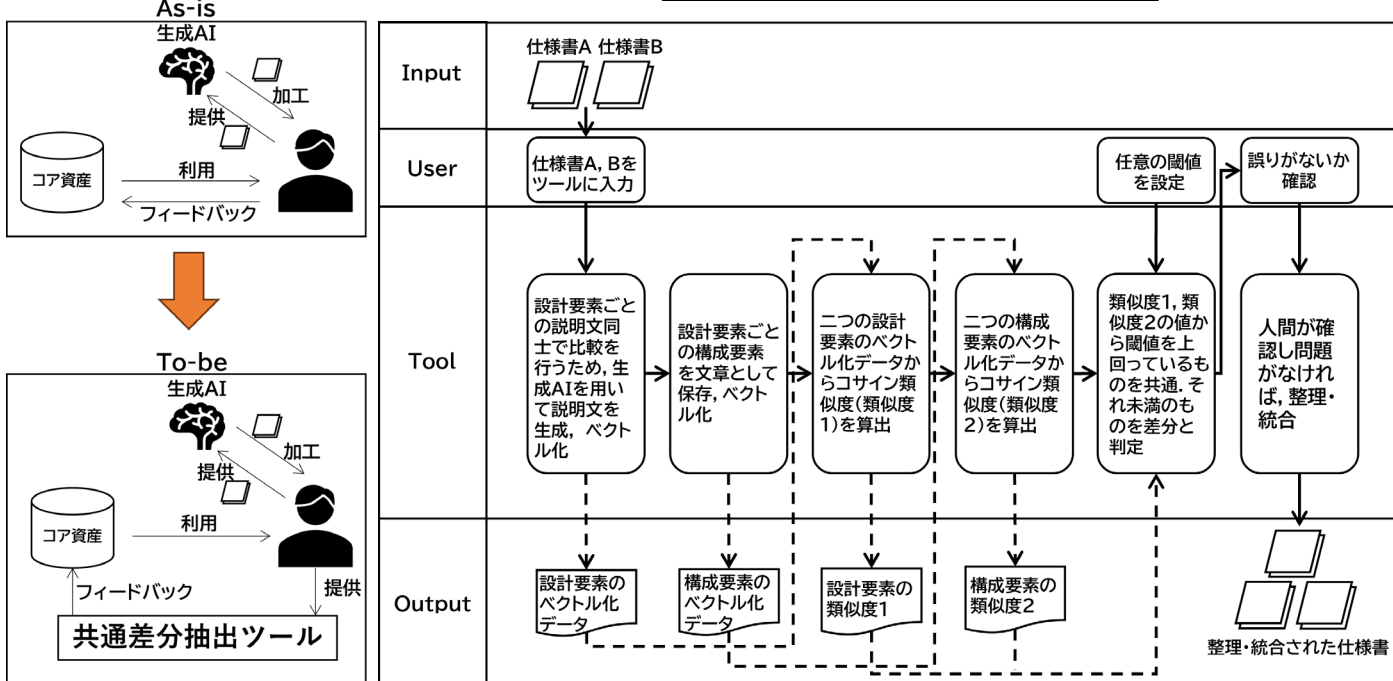
背景

- 生成AIを活用してソフトウェア要求仕様書を自動生成する取り組みが増加[1][2][3]
- 複数回の生成により, 何が共通で何が可変か把握が困難となる課題
- 共通部分と差分部分を整理・統合し, コア資産を効率的に再利用する仕組みが必要

仕様書の共通・差分抽出アプローチ

- 生成AIによって得られた複数の仕様書を対象に, 共通部分と差分部分を自動的に抽出する仕組み
- 名称の説明と構成要素の類似度に基づき共通・差分を抽出, 利用者は結果をもとに統合して目的に応じた仕様書を効率的に生成

エコシステムに向けた共通差分抽出



設計要素間の類似性を2つの類似度算出結果を用いて判定

$$S_1 = \frac{\vec{v}_e(t_A) \cdot \vec{v}_e(t_B)}{\|\vec{v}_e(t_A)\| \|\vec{v}_e(t_B)\|}$$
$$t_A = EP(DE(A, table, i))$$
$$t_B = EP(DE(B, table, i))$$
$$S_2 = \frac{\vec{v}_e(u_A) \cdot \vec{v}_e(u_B)}{\|\vec{v}_e(u_A)\| \|\vec{v}_e(u_B)\|}$$
$$u_A = EL(DE(A, table, i))$$
$$u_B = EL(DE(B, table, i))$$

$DE(A, table, i)$. Aシステムのi番目のtable名
 $EP(S)$. 設計要素Sの説明文, $S = DE(A, table, i)$
 $EL(S)$. 設計要素Sを構成する項目をカンマ区切りでつなげた項目リスト,
 $S = DE(A, table, i)$

$\vec{v}(t)$. 文tの文書ベクトル
 t_A, u_A . システムAの文
 t_B, u_B . システムBの文

S_1 は類似度1を算出
 S_2 は類似度2を算出

類似度1では意味的な類似度を判定
→異なる名称で同じ意味を判別
類似度2では構成要素的な類似度を判定
→項目の構成が似ている仕様を判別

共通・差分の判断例



今後の取り組み

- ・閾値を変えた場合の適用評価
- ・具体事例を変更し, 適用評価

[1]中野丈, 島川遼太郎, 杉村康気, 宅間健生, 村野遼, 奥田博隆, 位野木万里, 要求工学の知見をプロンプト設計に組み込んだ生成AIを活用した要求定義支援手法, 情報処理学会, 研究報告ソフトウェア工学(SE), Vol.2024-5E-216, No.9, pp.1-7, 2024.

[2]島川遼太郎, AI for Requirements Engineering に関する研究: プロンプト連鎖による要求定義手法の提案, 工学院大学, 2024

[3]宅間健生, 生成AIによるプロンプト連鎖の自動実行による要求定義支援ツールの提案, 工学院大学, 2024

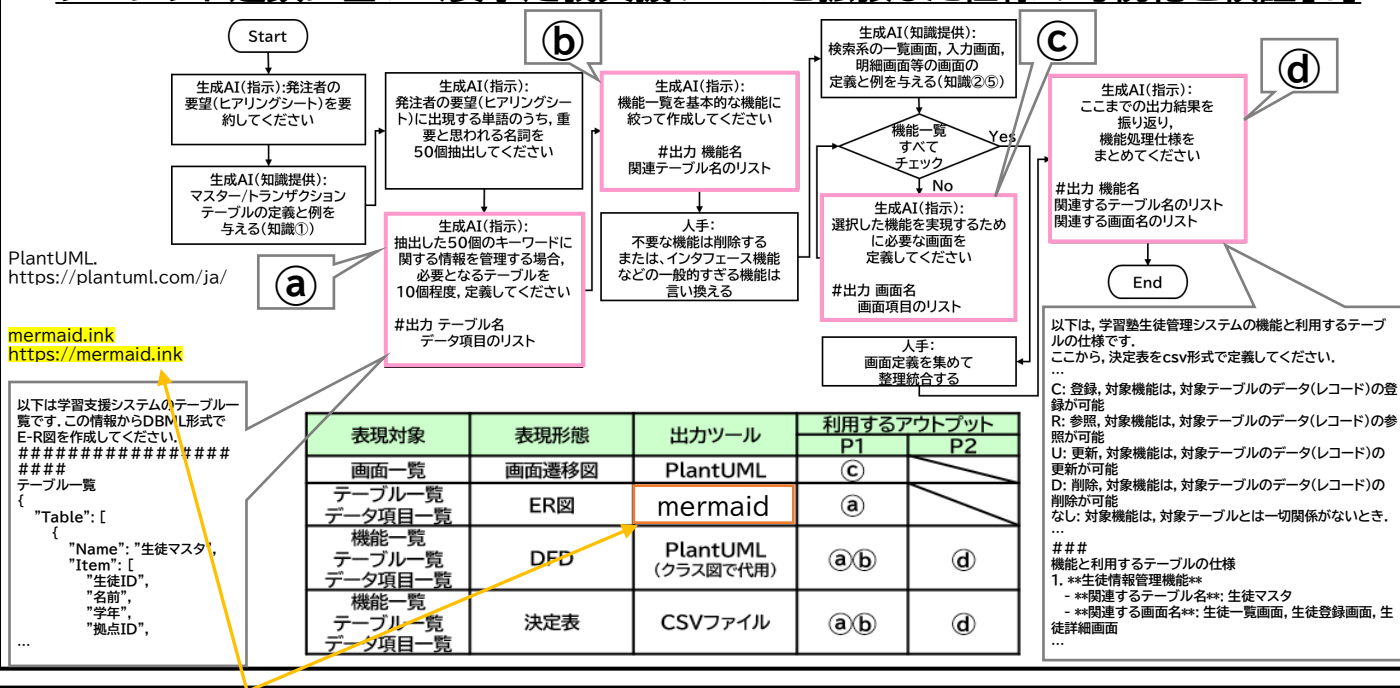
背景

- ソフトウェア工学に対するLLMを含む生成AIを活用する事例が増加している。その一つにソフトウェア要求仕様書を自動生成する取り組み
- LLMを用いてテキストからモデルを生成するT2M(Text-to-Model)を行う試み

アプローチ

- 要求仕様書から得られたテキストをLLMを用いてPlantUMLとMermaidのソースに変換
- PlantUMLとMermaidのソースを画像化

プロンプト連鎖に基づく要求定義支援ツールを拡張した仕様の可視化と検証[1]

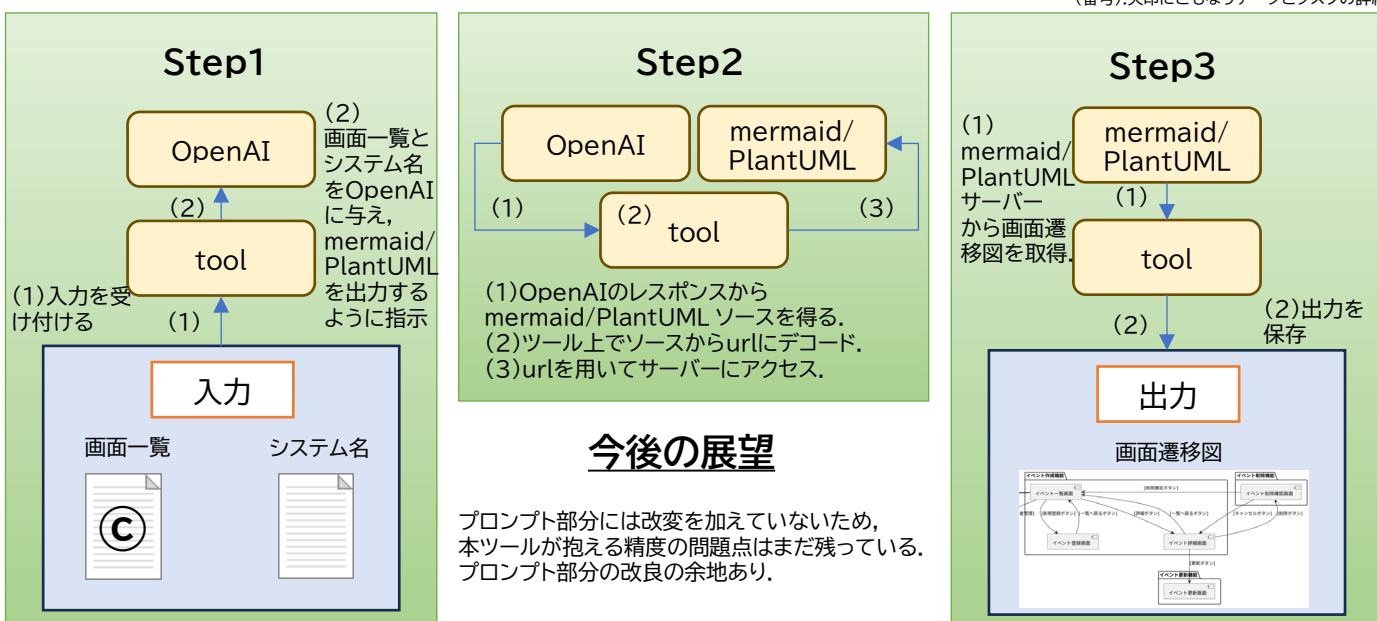


画面遷移図生成の例

dbdiagramから
mermaidに変更

ツールを拡張して仕様の可視化部分を自動化

step1 to 3:段階ごとのツールの動作
矢印:オブジェクトとのデータのやり取り
(番号):矢印にともなうデータとタスクの詳細



プロンプト連鎖による要求定義手法の妥当性検証

高杉柚月, 横井新大, 島川遼太郎, 陳家輝, 位野木万里

背景・課題

- プロンプトエンジニアリング手法と要求工学の知識融合させたプロンプト連鎖による要求定義手法・ツールを提案[1]
- 評価事例が少なく汎用性や実用性の検証不足
- 精度向上だけでなく人を支援するツールとして活用方法の整理が課題

解決策へのアプローチ

- 有識者の意見を踏まえて要求定義支援ツールの評価視点を整理し最適なパターンや活用方法を検討
- プロンプト連鎖による要求定義手法を用いて新たにSOR型システムを対象として適用評価

要求定義支援ツール

8種類のプロンプト連鎖のパターン(S1~S8)を提示
知識・Self-consistencyの有無, プロンプト設計のアプローチ(POA または DOA)の違いに基づいて構成
加えてS0のZero-shot 生成による知識を一切与えないパターン

パターン	パターン名	与える知識
S1	POA知識なし	知識①②
S2	POA知識あり	知識①②③④
S3	DOA知識あり	知識①②
S4	DOA知識あり	知識①②③④
S5	POA知識なし(Scあり)	知識①②⑤
S6	POA知識あり(Scあり)	知識①②③④⑤
S7	DOA知識なし(Scあり)	知識①②⑤
S8	DOA知識あり(Scあり)	知識①②③④⑤

「POA(Process-Oriented Approach:機能中心アプローチ)」
機能, テーブル, 画面の順に設計要素を仕様化
「DOA(Data-Oriented Approach:データ中心アプローチ)」
テーブル, 機能, 画面の順に設計要素を仕様化

【与える知識】

知識① テーブルの種類として、マスタテーブルと
トランザクションテーブルの存在の示唆
知識② SOR型情報システムの画面種類(登録画面・一括登録画面・
一覧画面・詳細画面・表示画面など)の示唆
知識③ SOR型画面に備える典型的な画面項目(登録ボタン・
検索ボタンなど)の示唆
知識④ 画面種類別に、知識③で述べた画面項目との対応の示唆。
知識⑤ SOR 型の情報システムを対象にしてSelf-Consistencyを
実践するためのノウハウ

適用評価

【目的】

要求定義支援ツールを適用して生成した仕様書の
品質を評価してツールを改善

【評価方法】

対象案件 : イベント開催及び参加者管理を行う
イベント募集管理システム
入力 : モデル, 知識, システム概要書
仕様生成 : DOA・POAツールを使用
生成パターン: S0(Zero-shot)~S8の9パターン
生成数 : 各パターン3回ずつ生成(計27件)
評価対象 : 各パターンの生成項目数上位2位(計18件)

【評価結果の評価】

理由 : 実際の開発現場では明確な正解が存在
しないため正解のない仕様を評価すること
でツールの活用方法を明確化
評価者 : 著者本人および有識者
観点 : 以下の表に示した4つの評価項目
4つの評価項目とその説明について[2]

評価項目	内容
Correctness(正確性)	生成結果が正しいか?
Completeness(網羅性)	生成結果に漏れがないか?
Consistency(一貫性)	設計要素間に矛盾はないか?
Non-redundancy(非冗長性)	生成結果に冗長性はないか?

評価結果

【正確性】

- イベント告知更新機能や
参加者情報ファイル出力機能など全体的に過剰な細分化

【網羅性】

- 基本的な機能やテーブルは揃っている
- 申し込み状況カレンダー照会が存在しないなど画面に
関しては不足

【一貫性】

- テーブルには存在するが対応する画面が不足

【非冗長性】

- 不要機能や重複要素が多く全体的に整理が不十分で
冗長性が高い

【ツールの活用方法】

- DOAでは主要テーブルは揃っており不足項目も少ない
ため仕様に必要なテーブルやデータ構造の正確さを確認
- 生成結果に大きな差異がないため機能の和集合を
取ることで全体機能の把握が容易
- S7では48画面, S0では8画面とパターンによって
生成数が異なり多くの画面がある場合は不要画面の
削除や追加の検討が容易

今後の展望

- QCDを加速するために人とAIが協働する具体的な
プロセスを明確化

価値要素を活用した類似性に基づくユーザ要求のコンテンツマッチング手法の提案

田中恵美, 安藤勇輝, 南須原大吾, 横岡歩, 位野木万里

背景

- DX進展に伴う要求仕様書の複雑化と初級技術者による仕様書理解の困難
→発注者と提案者間での認識齟齬確認のため要約技術を用いた手法が提案^[1]

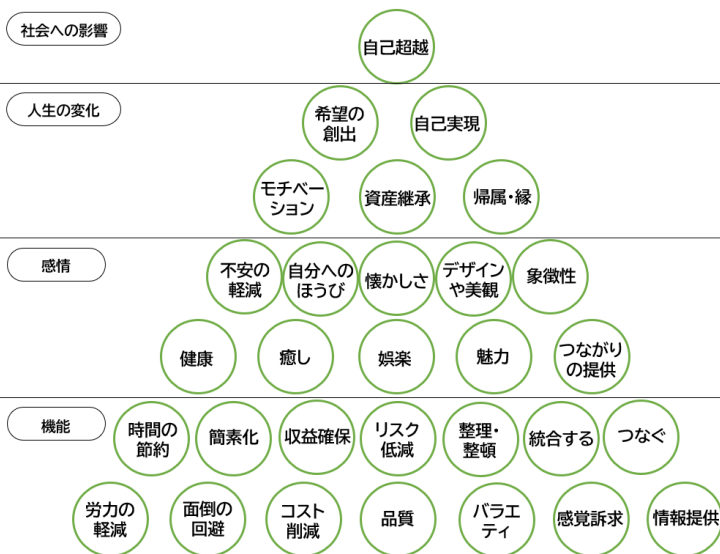
課題

- 既存の要約技術は文面情報に偏り発注者が本当に重視する価値や意図などの背景的情報の欠落の可能性
→発注者の真の要求に合致するかの判断が困難

解決策へのアプローチ

- 発注者と提案者の価値の違いを定量的に可視化する手法が必要
→ビジネス領域でも活用される *Almquist* の30の価値要素^[2]に着目
→要求理解における価値観把握への応用可能性

価値要素のピラミッド^[2]



価値要素を活用した類似性に基づくユーザ要求のコンテンツマッチング手法

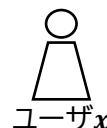
マッチ度の計算と定義

- 比較対象: 文書 a, b
- $\vec{v}(x)$ はユーザ x の価値観ベクトル.
- $\vec{v}(a)$ は文書 a , $\vec{v}(b)$ は文書 b の価値ベクトル.
- $\vec{v}(x), \vec{v}(a), \vec{v}(b) = \begin{bmatrix} v_1 \\ \vdots \\ v_i \\ \vdots \\ v_{30} \end{bmatrix}, i \in \mathbb{N}, 1 \leq i \leq 30, 1 \leq v_i \leq 5$
 v_i は, *Almquist* の i 番目の価値要素における *value*

Step1:

Almquist の30の価値要素それぞれに対しての重要度を1~5で選択しユーザ x の価値観ベクトルを作成

$$\vec{v}(x) = \begin{bmatrix} v_1 \\ \vdots \\ v_{30} \end{bmatrix}$$



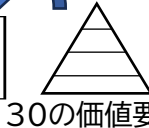
Step2:

同様に文書 a, b に対して30の価値要素に基づく評価を行い比較する文書に含まれる価値ベクトルを作成

$$\vec{v}(a) = \begin{bmatrix} v_1 \\ \vdots \\ v_{30} \end{bmatrix}$$



$$\vec{v}(b) = \begin{bmatrix} v_1 \\ \vdots \\ v_{30} \end{bmatrix}$$



Step3:

ユーザ x と文書 a , ユーザ x と文書 b のマッチ度を算出

マッチ度を MR とおく.

$$MR(Who, Doc) = \frac{\vec{v}(Who) \cdot \vec{v}(Doc)}{\|\vec{v}(Who)\| \|\vec{v}(Doc)\|}$$

Who : ユーザ x

Doc : 比較する文書, ここでは $Doc = \text{文書 } a \text{ or 文書 } b$

Step4:

マッチ度の高い方がユーザ x の価値観にマッチしていると判定

$$Max(MR(x, a), MR(x, b))$$

今後の展望

- 複数の実験協力者で適用評価実験を行い, 提案手法の有効性と結果の信頼性を検証する.

[1] Rodrigues, M., Pinto, F., Content Matching for City Improvement, Proc. of the International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME 2023), 2023.

[2] Almquist, E., Senior, J., & Bloch, N. The elements of value: Measuring—and delivering—what consumers really want. Harvard Business Review, 42(3), 56–65, 2017.

記述状況可視化技術に基づく 要求仕様書の日中技術者間理解支援手法の提案

工学院大学 陳家輝, 安藤勇輝, 南須原大吾, 田中恵美, 横岡歩, 位野木万里

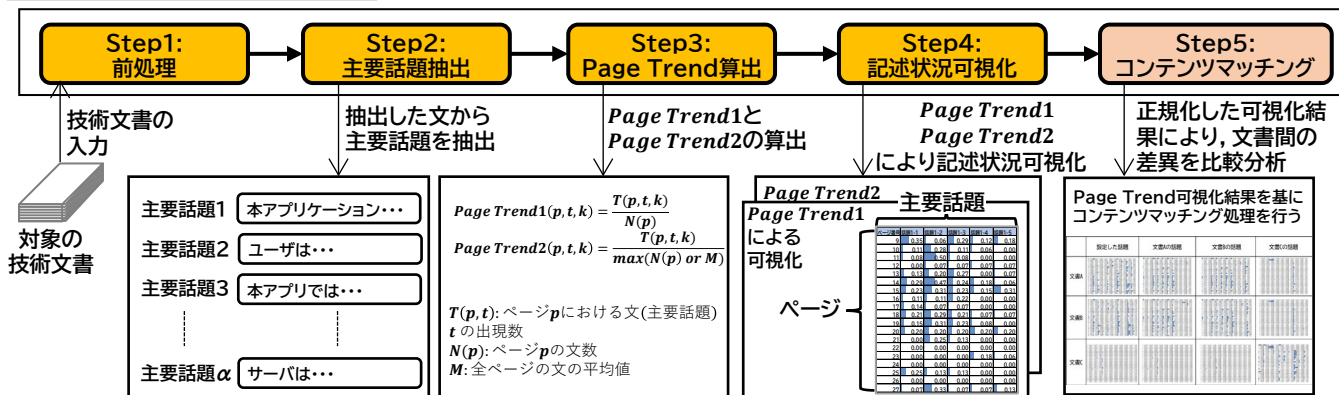
課題と背景

- グローバル化に伴い、日中両国の企業間で情報システム協同開発が増加し、言語・文化差による技術文書の理解不一致が課題
- 技術文書間の内容構成や情報差を分析・可視化し、読者が文書の関連性や違いを直感的に把握できるよう支援する情報提供が有効
- AIによる文書分析は進展しているが文書間の関係性と差異を明確化する理解支援手法は未成熟

解決策へのアプローチ

- 文書理解支援の高度化に向け、文書構造の特徴と内容差異を多面的に把握する可視化アプローチを導入
- Page Trendによる記述状況可視化ツール[1]におけるページ単位の重要話題分布の提示と文書全体の流れの把握支援
- コンテンツマッチング手法により、年度や種類の異なる文書間の内容対応を明確化し、可視化に基づく理解支援効果を検証

記述状況可視化ツール^[1]



比較検証: Page Trend可視化結果を基にした文書間分析

$Page\ Trend_{norm}(p', t, k)$: 正規化後のPage Trend値, α : 正規化後のページ数

$\vec{v}(x, t)$: 話題ベクトル, $\cos(x, y, t)$: 文書間の類似度

$s(p')$: $e(p')$: 正規化後ページ p' に対応する元ページ区間の開始と終了位置(起点ページ・終点ページ)

$\Delta(p')$: 区間に含まれる元ページの総数(ページ数の差)

・コンテンツマッチング処理に基づく複数文書間Page Trendマトリクス

	設定した話題	文書Aの話題	文書Bの話題	文書Cの話題
文書A				
文書B				
文書C				

・正規化後の話題ベクトルに基づく文書間cos類似度一覧表

	話題1	話題2	話題3	話題…	話題14	話題15	話題16	話題…	話題27	話題28	話題…	話題39	話題40	閾値は0.5以上場合
文書AとBのcos類似度	0.483	0.745	0.616		0.682	0.722	0.773		0.714	0.483		0.000	0.000	29
文書AとCのcos類似度	0.000	0.262	0.097		0.323	0.000	0.000		0.000	0.000		0.000	0.000	2
文書BとCのcos類似度	0.000	0.400	0.193		0.490	0.000	0.000		0.000	0.000		0.000	0.000	1

・ Step5: コンテンツマッチング

正規化処理: 次元のダウンサイジング

対象文書間でページ数が異なるため、最小ページ数の文書を基準として全Page Trend2を再スケーリングし、各文書をページ区間ごとに平均化して正規化されたPage Trend_{norm}を算出する。

$$Page\ Trend_{norm}(p', t, k) = \frac{1}{\Delta(p')} \sum_{p=s(p')}^{e(p')} Page\ Trend2(p, t, k)$$

話題別ベクトル空間の構築

各文書の正規化済みPage Trend_{norm}値をページ方向に並べ、話題ごとにベクトルとして構築する。これにより、文書内における話題分布の傾向を表現する。

※ x, y, zはそれぞれ文書A, 文書B, 文書Cを表す

$$\vec{v}(x, t) = \begin{pmatrix} PageTrend_{norm}(1, t, x) \\ PageTrend_{norm}(2, t, x) \\ \vdots \\ PageTrend_{norm}(\alpha, t, x) \end{pmatrix}$$

文書A・B・C間のコサイン類似度の算出

正規化後の話題ベクトルのcos類似度により文書間の傾向を比較する。合計40話題の中cos ≥ 0.5 を閾値とし、該当数で類似性を判断。

$$\cos(x, y, t) = \frac{(\vec{v}(x, t) \cdot \vec{v}(y, t))}{(\|\vec{v}(x, t)\| \|\vec{v}(y, t)\|)}$$

文書Aと文書Bは高い類似度を持つことが分かる

まとめ

- コンテンツマッチングを用いた、Page Trend可視化により、文書構造や情報対応の特徴を把握しやすくなり、理解促進に寄与すると期待

今後の課題

- マッチング結果を定量的に評価する
- 今後は、中国語技術文書の適用評価実験を追加し、手法の検討を継続

要求工学の知見をプロンプト連鎖に融合した生成AIによる要求定義支援手法の提案と適用評価

島川遼太郎¹, 宅間健生¹, 奥田博隆², 位野木万里¹

背景

- DX化の需要が増加
- 技術者が情報システムの**要求仕様書**を作成する機会が増大
- 要求仕様書の**自動生成**は業務効率化に有効
- プロンプトエンジニアリング**手法が提案
- 生成AIによるペルソナやシナリオの自動生成は取り組まれているが要求仕様書に特化したプロンプトは未成熟
- 技術者が効率的に要求仕様書を作成
- プロンプトエンジニアリング**と**要求工学**の知見を融合し、要求仕様書を**自動生成**

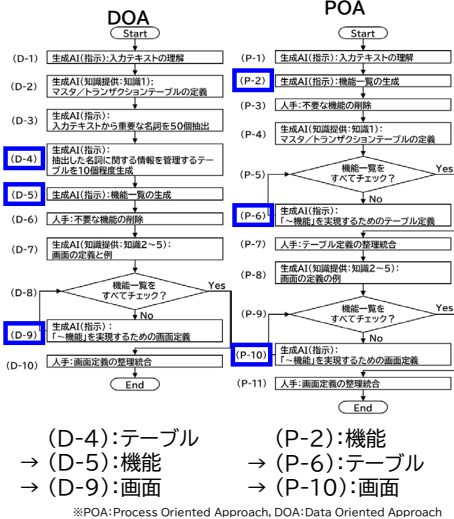
課題

- プロンプトエンジニアリングと要求工学の知見を融合する方法は複数存在
 - (1)DOA or POA
 - (2)Self-Consistency実施の有無
 - (3)設計要素の種類に関する知識の提供の有無
- 上記の組み合わせによる性質や違いが不明

解決策へのアプローチ

- 実事例で、複数のパターンで要求定義を行うことで、プロンプト連鎖の違いを明らかにする

プロンプト連鎖



知識

- 要求定義に関連する知識として定義
- メタモデルに基づいて形式化
- 入力として生成AIに提供

知識1: マスタランザクションテーブルの存在

知識2: SoR型情報システムに存在する、登録、一覧、詳細等の画面の種類

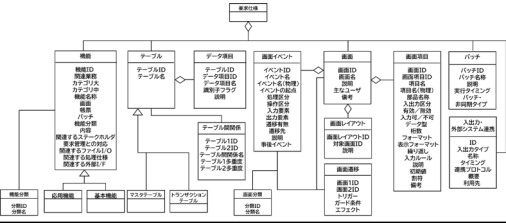
知識3: 画面に用いられる、登録/検索ボタン等の画面項目の種類

知識4: 画面の種類毎に、知識3の画面項目との対応関係

Self-consistency

- 導出過程を含む例を学習させて、生成を指示し、一貫した解を選択する方法
- 機能、画面、テーブル間の整合性を維持しながら定義を導出する過程の例
- 知識5として生成AIに付与

メタモデル



プロンプト連鎖のパターン

パターン	プロセス	Self-consistency	与える知識
S0	Zero-shot	Zero-shot	Zero-shot
S1	POA	行わない	知識1,2
S2	POA	行わない	知識1,2,3,4
S3	DOA	行わない	知識1,2
S4	DOA	行わない	知識1,2,3,4
S5	POA	行う	知識1,2,5
S6	POA	行う	知識1,2,3,4,5
S7	DOA	行う	知識1,2,5
S8	DOA	行う	知識1,2,3,4,5

適用評価

対象となる事例: 学習塾生徒支援システムのシステム要望書

システム概要	文字数: 2,863文字
- 会社概要 - システム化の目的 - 顧客の業務内容(概要) - ステークホルダ - システムが対象とする業務 - 業務の補足 - システムに対する要望 - システム化の方向性	ページ数: 119ページ 正確仕様書
- 機能一覧 (機能15個) - データモデル (テーブル9個) - 画面仕様 - 画面一覧 (画面31個) - 画面遷移図 - 画面項目定義 - 画面イベント定義 - 画面イメージ - バッチ・非同期処理仕様 - ファイル入出力・外部システム連携仕様	機能一覧、テーブル、データ項目、画面一覧、画面項目の要求仕様書を作成 S0からS8の各パターンを6回ずつ生成

実行環境:
 ・pythonのopenAIのライブラリでプロンプト連鎖を行うツールを作成
 ・ChatGPT: gpt-4o
 ・temperature: 0.4

評価結果

評価結果(機能、テーブル、画面、データ項目、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、テーブル)

項目	生成数	再現率	適合率	F値
1 全体	71	0.5859	0.502	0.514
2 S0	28	0.436	0.662	0.500
3 S1	80	0.547	0.404	0.431
4 S2	75	0.567	0.430	0.458
5 S3	94	0.671	0.522	0.471
6 S4	70	0.586	0.417	0.450
7 S5	70	0.650	0.423	0.443
8 S6	68	0.652	0.556	0.597
9 S7	70	0.652	0.556	0.597
10 S8	69	0.638	0.520	0.537

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	適合率	F値
1 全体	652	0.4659	0.4319	0.418
2 S0	139	0.341	0.640	0.394
3 S1	80	0.435	0.221	0.343
4 S2	449	0.462	0.360	0.377
5 S3	495	0.500	0.428	0.448
6 S4	444	0.631	0.478	0.493
7 S5	537	0.464	0.347	0.364
8 S6	535	0.460	0.348	0.369
9 S7	504	0.610	0.476	0.492
10 S8	495	0.589	0.474	0.486

評価結果(機能、画面項目)

項目	生成数	再現率	
----	-----	-----	--