

Supporting Software Maintenance with Dynamically Generated Document Hierarchies

Katherine R. Dearstyne

Computer Science and Engineering
University of Notre Dame
Notre Dame, IN, USA
kdearsty@nd.edu

Alberto D. Rodriguez

Computer Science and Engineering
University of Notre Dame
Notre Dame, IN, USA
arodri39@nd.edu

Jane Cleland-Huang

Computer Science and Engineering
University of Notre Dame
Notre Dame, IN, USA
JaneClelandHuang@nd.edu

Proceedings of the 40th IEEE International Conference on Software Maintenance and Evolution (ICSME 2024)

背景

- ソフトウェアのドキュメンテーションは以下の
広範囲をサポート^[1~5]
 - 影響分析
 - 安全性評価
 - 変更分析
 - 新規開発者の参入
 - 高品質で多層的な構造のドキュメンテーションの
作成や維持は非常に時間がかかる作業^[6~8]
- ⇒ 利点が多くあっても適切なドキュメンテーション
が難しく欠如しがち^[9]

[1] : A. Forward and T. C. Lettbridge, "The relevance of software documentation, tools and technologies: A survey," in Proceedings of the 2002 ACM Symposium on Document Engineering, ser. DocEng '02, New York, NY, USA: Association for Computing Machinery, 2002, p. 26-33. [Online]. Available: <https://doi.org/10.1145/585058.585065>

[2] : T. Lettbridge, J. Singer, and A. Forward, "How software engineers use documentation: the state of the practice," IEEE Software, vol. 20, no. 6, pp. 35-39, 2003.

[3] : T. Roehm, R. Tarks, R. Koschke, and W. Maalej, "How do professional developers comprehend software?" in 2012 34th International Conference on Software Engineering (ICSE), 2012, pp. 255-265.

[4] : O. Gotel and A. Finkelstein, "Contribution structures (requirements artifacts)," in Second IEEE International Symposium on Requirements Engineering, March 27-29, 1995, York, England, UK: IEEE Computer Society, 1995, pp. 100-107. [Online]. Available: <https://doi.org/10.1109/RE.1995.512550>

[5] : S. Marw, J. Steghofer, and M. Sarron, "Software traceability in the automotive domain: Challenges and solutions," in Software Engineering and Software Management, SE/SWM 2019, Stuttgart, Germany, February 18-22, 2019, ser. LM, S. Becker, L. Bogicevic, G. Herzum, and S. Wagner, Eds., vol. P-292, GI, 2019, pp. 61-62. [Online]. Available: <https://doi.org/10.3840/se2019-14>

[6] : J. Cleland-Huang, O. Gotel, J. H. Hayes, P. Ma'edec, and A. Zisman, "Software traceability: trends and future directions," in Proceedings of the 2014 Future of Software Engineering, FOSE 2014, Hyderabad, India, May 31 - June 7, 2014, I. D. Hebl and M. B. Dwyer, Eds. ACM, 2014, pp. 55-69. [Online]. Available: <https://doi.org/10.1145/2593882.2593891>

[7] : A. D. Lucia, A. Marcus, R. Oliveira, and D. Poshvanyk, "Information retrieval methods for automated traceability recovery," in Software and Systems Traceability, Springer, 2012, pp. 71-98.

[8] : M. Rath, J. Rendall, J. L. Guo, J. Cleland-Huang, and P. Ma'edec, "Traceability in the wild: automatically augmenting incomplete trace links," in Proceedings of the 40th International Conference on Software Engineering, 2018, pp. 834-845.

[9] : B. Dagenais and M. P. Rohlfed, "Creating and evolving developer documentation: Understanding the decisions of open source contributors," in Proceedings of the Eighteenth ACM SIGSOFT International Symposium on Foundations of Software Engineering, ser. FSE '10, New York, NY, USA: Association for Computing Machinery, 2010, p. 127-136. [Online]. Available: <https://doi.org/10.1145/1882201.1882312>

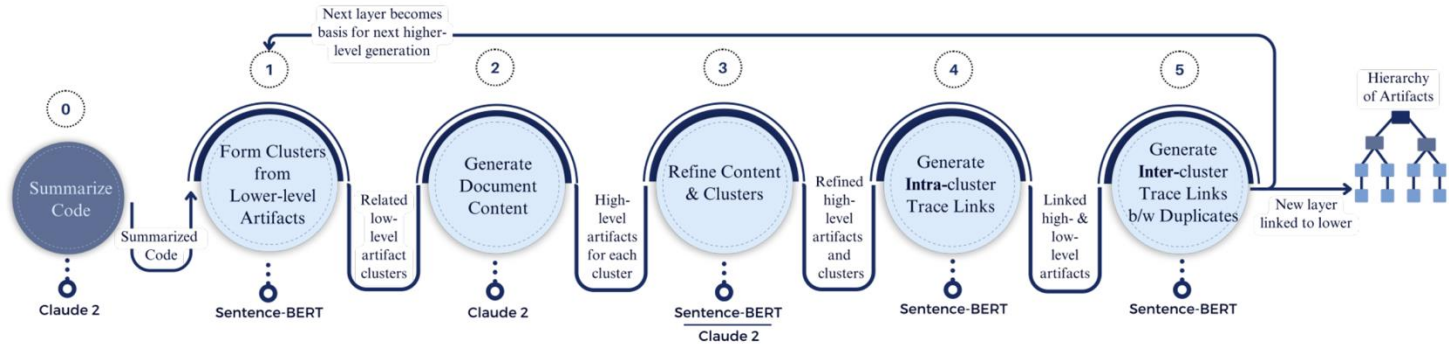
関連研究

- コードレベルのドキュメント自動生成
 - API仕様の自動化^[10~18]や詳細なコードレベルでのドキュメント生成^[19]
- ソースコード理解による説明の生成^[20~23]
 - クラスやメソッド, パラメータレベルのコード要約
- ユビキタス・ソフトウェア・トレーサビリティの追求^[24~28]
 - LLMによる自動トレース・リンク生成

提案手法

- LLMを用いて多層階層のドキュメントを自動生成するパイプライン HGEN
 - システム全体の概要レベルからコードレベル, またサブシステムなど階層構造で表したドキュメントを生成
 - 階層同士を接続するトレースリンクも同時に生成
- ソフトウェア保守活動を効果的に支援

HGEN のプロセス：概要



0. 最初の入力のコードを処理する場合、コードを自然言語の要約に変換
1. 1階層下の説明を受け取り機能や特徴でクラスター化
2. 各クラスターに対して機能など自然言語の説明を生成
3. 生成された重複する情報が含まれる説明を整理
4. 生成された説明と1階層下の入力を動的にリンク
5. さらに上位層が必要な場合、現在の説明を次の入力とし、すべての階層が生成されるまで繰り返す

HGEN のプロセス：ステージ0

- 最初の入力のコードを処理する場合は，コードから自然言語の要約を作成
 - Anthropic Claud 2.0モデルで以下のステップにより作成
 1. コードがユーザーに提供する機能の概要
 2. コードがどのように機能をユーザに提供するかの説明を加えた洗練された要約

HGEN のプロセス：ステージ1

- 生成文を受け取り機能や特徴でクラスター化
 1. 生成文をSentence-BERTで埋め込み表現に変換
 2. 5つのクラスタリングアルゴリズムでクラスタ生成
 3. 1つまたは5つ以上のクラスタを除外
 4. 類似度やサイズから各クラスタに重要度スコアを計算
 5. 重要度スコアの高いものから順にランク付け
 6. 類似度が1.5標準偏差以上離れた生成部を除外
 7. 重要度ランクが高い順に重複がないか検討
 8. クラスタに属さない生成文の再検討

HGEN のプロセス：ステージ2

- ステージ1 のクラスタを基に新しいドキュメント層を生成
 1. 生成するドキュメント形式_(要件, エピックなど)の指定
 2. クラスタごとに生成する新しいドキュメント数を決定
 3. Claudで各クラスタから指定の形式に沿ったドキュメントを生成

HGEN のプロセス：ステージ3

- 重複する情報が含まれる生成文を整理
 1. 新しい生成文が別のクラスタの内容と重複するか確認
 2. 類似度が高い生成文同士を再クラスタ化
 3. 重複がある場合類似した生成文を再生成

HGEN のプロセス：ステージ4

- 生成文と1階層下の入力を動的にリンク
 1. 各クラスタ内で階層が上下の生成文間にリンクを作成
 2. 階層が上下の生成文に埋め込みベクトルを生成し、
コサイン類似度を計算
 3. 各クラスタ内で類似度を正規化し、最高が1になるよう
スケーリング
 4. 類似度が0.8未満のペアのリンクを削除

HGEN のプロセス：ステージ5

- 異なるクラスタ間で重複している生成文同士が関連していればトレースリンクを作成
 - 全生成文間のコサイン類似度を計算し，平均から2標準偏差以上高いペアをマーク
 - 各重複ペアについてトレースリンクを作成
 - 類似度が高い2つの生成文が同じ下の階層の生成文にリンクしている場合は片方を削除
- さらに上位層が必要な場合は現在の生成文を次の入力とし，全ての階層が生成されるまで繰り返す

データセット

- 異なるドメイン，言語，開発プロセス，ドキュメント構造を持つ3つのプロジェクト

Dronology * [32]

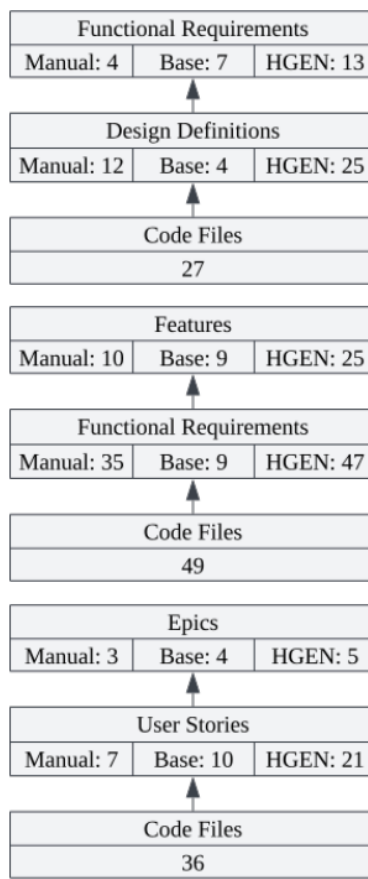
Dronology is an Open-Source small Unmanned Aerial System (sUAS) written in Java. It provides a platform for controlling and coordinating multiple sUAS to support search-and-rescue, surveillance, and scientific data collection missions.

SAFA [33], [34]

SAFA is a software documentation management platform that leverages live traceability to build a knowledge graph and support change impact analysis. Our industry collaborators provided access to closed-source client-side source code.

Jack of Clubs *

Jack of Clubs is a re-creation of Ace of Spades, a voxel-based first-person shooter game. The creator gave us access to the manually created user stories and epics, and we are releasing them to the public as part of this paper.



- Dronology
423のJavaファイル, 211の設計定義, 99の要件
- SAFA
242のVueファイル, 262の機能要件, 101の機
- Jack of Clubs
36のC++ファイル, 7のユーザーストーリー, 3のエピック

RQ

- プロジェクトごとに1人ドキュメントの作成経験があるシニア開発者が専門家として評価に参加
- RQ1. 成果物の品質：個々のモデルが生成した成果物は専門家が生産した成果物の品質と比べてどうか
- RQ2. カバー範囲：元のドキュメントに現れる概念は生成されたドキュメントでどの程度カバーされているか
- RQ3. 関係：モデルが生成したドキュメントについて、適切な親子関係をどの程度効果的に構築しているか

比較対象

- 人手によって作成されたドキュメント

Group	Metric	Metric Description
Language	Readability	How easily a user can understand the information provided.
	Appropriateness	Language appropriateness with respect to the artifact's technical level.
Content	Conciseness	How brief, yet clear the system is described.
	Importance	Considers the importance of the information provided.
Effectiveness	Usefulness	Considers how useful this documentation is to the project expert.
	Helpfulness	Consider how helpful the documentation is for understanding its children.

- Claude 2.0で単純なアプローチにより作成されたドキュメント
 1. コードの要約から各層のドキュメントを作成
 2. 異なる層の生成文の間でコサイン類似度が0.7以上を基準に直接リンクを設定

RQ1 : 成果物の品質

<i>Human vs. Baseline</i>			
Metric	Human Mean	Baseline Mean	Corrected P-value
Readability	3.90	4.44	0.002445
Appropriateness	3.70	4.00	0.214476
Conciseness	4.30	4.44	0.596927
Importance	4.10	4.35	0.191984
Usefulness	3.51	3.67	0.794294
Helpfulness	3.82	3.88	0.990378

<i>Human vs. HGEN</i>			
Metric	Human Mean	HGEN Mean	Corrected P-value
Readability	3.90	4.38	0.000104
Appropriateness	3.70	4.16	0.001493
Conciseness	4.30	4.32	0.618787
Importance	4.10	4.33	0.095525
Usefulness	3.51	4.14	7.09e-06
Helpfulness	3.82	4.21	0.002445

<i>Baseline vs. HGEN</i>			
Metric	Baseline Mean	HGEN Mean	Corrected P-value
Readability	4.44	4.38	0.990378
Appropriateness	4.00	4.16	0.462604
Conciseness	4.44	4.32	0.990378
Importance	4.35	4.33	0.990378
Usefulness	3.67	4.14	0.020646
Helpfulness	3.88	4.21	0.172422

- 1～5のリッカード尺度で評価
 - スコア分布が正規分布でないためMann-Whitney U検定($p < 0.05$)
 - 18回のテストを行う際に誤差を抑えるためHolm-Sidak法でp値を調整
- LLMは人間と同じかそれ以上の品質のドキュメントを作成できる
- ベースラインと違いHGENは階層構造を構築できる

RQ2 : カバー範囲

Project	ID	Baseline		HGEN	
		% Covered	Covered by	% Covered	Covered by
Dronology	E1	6.3%	9.1%	87.5%	28.9%
SAFA	E2	37.8%	38.9%	84.4%	43.1%
JOC	E3	50.0%	35.7%	100%	38.5%

- 各専門家に人手のドキュメントにある概念を特定しLLMが生成したドキュメントの網羅性を評価
 - 全3プロジェクトにおいて網羅率が顕著に増加
 - 概念を含む生成文の割合(Coverd by)も増加

RQ3 : 関係

Project	Approach	mAP	Precision	Recall	# Orphans
Dronology	Baseline	84.5%	47.2%	89.5%	17
	HGEN	94.0%	56.3%	93.4%	0
SAFA	Baseline	91.9%	49.2%	100%	28
	HGEN	94.5%	54.3%	98.4%	9
JOC	Baseline	95.5%	67.3%	74.5%	11
	HGEN	96.7%	81.4%	80.2%	1

- 生成されたドキュメント階層を可視化し正確さを確認
- 基準の正解を専門家が作成，生成されたリンクが一致するかを評価
 - HGENが高いPrecisionとRecallを，特にmAPは全プロジェクトで示した
 - ベースラインよりもHGENの方がリンクがない生成文が大幅に少ない
 - ✓ HGENは階層内で概念をより包括的に捉えソフトウェアの高レベルな理解を促進する

実際の企業プロジェクトへの適用

- 7企業の9プロジェクト(2つはOS)でHGENをテスト
 - HGENの生成ドキュメントをもとに企業と面談し内容を分析
-
- ドキュメントの品質：多くがポジティブな評価
 - 期待される用途：複雑なシステムの理解支援_{など}
 - 改善の提案：外部ライブラリの情報追加_{など}

まとめと所感

まとめ

- ソフトウェアメンテナンスにおいてソースコードからドキュメントを作成し負担軽減を目指す
- 人手によるドキュメントと遜色なく，ベースラインと比べて階層構造が網羅されたドキュメントを作成できた

所感

- HGENのステップの不明点が多かった
- プロジェクトごとに1人の専門家による評価なのが信頼に足るのか疑問

補足資料

- [10] : “Home.” [Online]. Available: <https://www.openapis.org/>
- [11] : D. van Heesch, “Doxygen, a tool for generating documentation from annotated source code (ver.1.9.7),” May 2023. [Online]. Available: <https://www.doxygen.nl/>
- [12] : “Welcome — Sphinx documentation.” [Online]. Available: <https://www.sphinx-doc.org/en/master/>
- [13] : “API Documentation & Design Tools for Teams | Swagger.” [Online]. Available: <https://swagger.io/>
- [14] : D. Kramer, “Api documentation from source code comments: A case study of javadoc,” in Proceedings of the 17th Annual International Conference on Computer Documentation, ser. SIGDOC ’99. New York, NY, USA: Association for Computing Machinery, 1999, p. 147–153. [Online]. Available: <https://doi.org/10.1145/318372.318577>
- [15] : “Django.” [Online]. Available: <https://www.djangoproject.com/>
- [16] : “Spring Boot.” [Online]. Available: <https://spring.io/projects/spring-boot>
- [17] : “FastAPI.” [Online]. Available: <https://fastapi.tiangolo.com/>
- [18] : “Node.js.” [Online]. Available: <https://nodejs.org/en>
- [19] : S. S. Dvivedi, V. Vijay, S. L. R. Pujari, S. Lodh, and D. Kumar, “A comparative analysis of large language models for code documentation generation,” 2023.
- [20] : G. Sridhara, E. Hill, D. Muppaneni, L. Pollock, and K. Vijay-Shanker, “Towards automatically generating summary comments for java methods,” in Proceedings of the 25th IEEE/ACM International Conference on Automated Software Engineering, ser. ASE ’10. New York, NY, USA: Association for Computing Machinery, 2010, p. 43–52. [Online]. Available: <https://doi.org/10.1145/1858996.1859006>
- [21] : G. Sridhara, L. Pollock, and K. Vijay-Shanker, “Generating parameter comments and integrating with method summaries,” in 2011 IEEE 19th International Conference on Program Comprehension, 2011, pp. 71–80.
- [22] : P. W. McBurney and C. McMillan, “Automatic source code summarization of context for java methods,” IEEE Transactions on Software Engineering, vol. 42, no. 2, pp. 103–119, 2016.
- [23] : —, “Automatic documentation generation via source code summarization of method context,” in Proceedings of the 22nd International Conference on Program Comprehension, ser. ICPC 2014. New York, NY, USA: Association for Computing Machinery, Jun. 2014, p. 279–290. [Online]. Available: <https://dl.acm.org/doi/10.1145/2597008.2597149>
- [24] : Software and Systems Traceability. London: Springer London, 2012. [Online]. Available: <http://link.springer.com/10.1007/978-1-4471-2239-5>
- [25] : G. Antoniol, J. Cleland-Huang, J. H. Hayes, and M. Vierhauser, “Grand challenges of traceability: The next ten years,” arXiv:1710.03129 [cs], Oct. 2017, arXiv: 1710.03129. [Online]. Available: <http://arxiv.org/abs/1710.03129>
- [26] : A. D. Lucia, A. Marcus, R. Oliveto, and D. Poshyanyk, “Information retrieval methods for automated traceability recovery,” in Software and Systems Traceability. Springer, 2012, pp. 71–98.
- [27] : M. Rath, J. Rendall, J. L. Guo, J. Cleland-Huang, and P. Mäder, “Traceability in the wild: automatically augmenting incomplete trace links,” in Proceedings of the 40th International Conference on Software Engineering, 2018, pp. 834–845.
- [28] : M. Rahimi and J. Cleland-Huang, “Evolving software trace links between requirements and source code,” in Proceedings of the 10th International Workshop on Software and Systems Traceability, SST@ICSE 2019, Montreal, QC, Canada, May 27, 2019, J. Steghöfer and N. Niu, Eds. IEEE / ACM, 2019, p. 12. [Online]. Available: <https://doi.org/10.1109/SST.2019.00012>