

「改訂版ブルーム・タキソノミー」を利用した ソフトウェアドキュメンテーションの改善手法の提案

山川 陽亮[†] 金城 篤史^{††}
[†]ピクシブ株式会社 ^{††}沖縄工業高等専門学校

1 はじめに

ソフトウェア開発において、チーム内で熟練した開発者(以降、熟練者)はドキュメントを作成することがある。[1]

ソフトウェア開発におけるドキュメントには、会社等の組織の中で開発されるシステムに対するドキュメント、OSS (Open Source Software) 開発プロジェクトで開発されるシステムのドキュメント、ある組織が別の組織・個人に対して公開する開発者向け SDK (Software Development Kit) に対するドキュメントなど複数の状況・用途が考えられるが、今回は開発組織における熟練者が社内の他の開発者向けに知識の共有を目的として作成するドキュメントに焦点を当てる。

情報工学においてはドキュメントを作成しなければいけない状況が存在するが、ドキュメントを執筆することで開発者が実装等に使える時間は減ることになってしまう。またドキュメントは充足したという状況を測ることが難しい、ドキュメント化対象のシステムが多すぎる、などの問題も存在する。知識の共有を目的としてドキュメントが作成されるのであれば、熟練者への属人性が高いサブシステムに対してドキュメントが無いものを洗い出すことでドキュメント化対象を絞ること、優先度を付けることが可能なのではないかと考えた。

教育学における「知識の呪縛」(すでに理解した情報を知らないものとして想定することは難しいこと)[2] は情報工学においても起きていることなのではないかと考える。例えば、熟練者が執筆したドキュメントは他の開発者にとって十分な情報を満たしたものとならない(知識の呪縛による認知バイアスがかかるため)などが挙げられる。

ドキュメントが良いものであれば、ドキュメントが対象とする人の理解度が上がると考えられる。そこで本稿では、ドキュメント作成前にドキュメント対象者の理解度を計測し、これを元にドキュメントを作成する。

ただ理解度は個人の認知に基づく指標であるため、本稿では教育心理学にて知られている改訂版ブルーム・タキソノミーという枠組を使い、理解度の数値化

を行う。

2 理解度を計測する指標

理解度を個人の認知だけではなく、指標を元に多角的に評価するための枠組みは複数あるが、そのうちの1つであり教育学にて用いられることのある「改訂版ブルーム・タキソノミー」[3] に注目することにする。

ブルーム・タキソノミーでは、学習者の行動を認知的領域、情意的領域、精神運動的領域の3つに分類する。このうち認知的領域は、記憶、理解、応用、分析、評価、創造、の6段階に分けられる。

3 アンケートの実施と対象者

本稿ではブルーム・タキソノミーにおける6段階の認知段階を情報工学に適用し、認知レベルを理解度を示す指標として用いて活用することを目指す。適用の手法としてはブルーム・タキソノミーに基づいたアンケートを作成し、これをドキュメント利用者に回答してもらい各サブシステムに対する理解度を測定する。これをドキュメンテーションに役立てることを目的とする。

アンケートの対象者は、山川と同じ開発組織に属し、山川と同じく広告配信システムの開発・運用に携わる9人の開発者となる。またこの全員から回答を得られた。これらのアンケート回答者はブルーム・タキソノミーについての事前知識が与えられないままアンケートを回答している。アンケート回答者はブルーム・タキソノミーを元に作成されたアンケートを用いて自身自身の認識を回答しており、回答者とは別の評価者等がアンケートを記入することは今回想定していない。

4 アンケートの対象とするシステム

今回のアンケートでは筆者の所属する開発組織で開発・運用を行っている広告配信システムに含まれる14個のサブシステム^{*1}、6個のツール、3個のインフラに関する設問を設定した。

5 アンケートの内容

理解度を選択するための具体例がアンケート上に書いてあり、それをもとにアンケート回答者が1~6の理解度を選択する。

Applying the Revised Bloom's Taxonomy to Improve Software Documentation

[†]Yosuke Yamakawa

^{††}Atsushi KINJO

pixiv Inc. (†)

Okinawa National College of Technology (††)

^{*1} うちフロントエンド開発、バックエンド開発、データ分析など同じシステムでも別の技術領域が存在するものが5つ存在し、これらはシステム B(frontend) のように領域別の設問を設定している

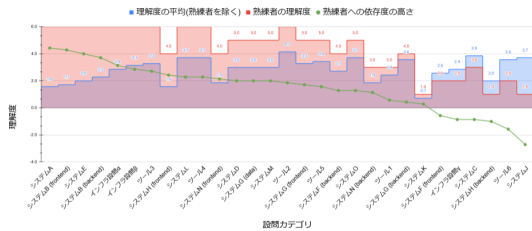


図 1: 理解度の平均と熟練者への依存度の高さ

アンケートの内容は以下の通りである

このサービスについてこれまでの質問を踏まえあなたの理解を教えてください^{*2*3}

(注釈)

- 1～6 のうち 6 に近いほど理想の状態だと考えてください
- 「認識・記憶できる」のうち例に挙げられている項目のすべてを満たさなくても「認識・記憶できる」に該当する場合はあります。下記文章はあくまで参考程度に考えてください

(理解度を選択するための具体例)^{*4}

レベル	説明と例
1. 認識・記憶できる	ツール上の情報を認識し利用する 例: このツールで特定の CPU 使用率のグラフを見つけられる など
2. 理解できる	情報を解釈し、説明や比較ができる 例: 各モジュールが何を担当しているか説明できる など
3. 応用できる	学んだ知識を具体的な状況で使える 例: 新しいパラメータを API に追加し、動作確認ができる など
4. 分析できる	情報を分解し、要素間の関係を理解できる 例: エラー発生原因をログやコードベースから特定できる など
5. 評価できる	批判的に情報を評価し、結論を導き出せる 例: 新しい要件を検討し、それが現在のアーキテクチャに適合するかを判断できる など
6. 創造できる	新しいアイデアや設計を作り出せる 例: 大規模な変更を伴う新しい機能を提案し、それを設計・実装できる など

6 結果

理解度は値が高いほど理解が深いものとし、‘熟練者の理解度 - 熟練者を除いた回答者の理解度平均’によって求めた数値を熟練者への依存度の高さとする。縦軸が hoge、横軸が hoge、棒グラフが hoge 結果を図 1 に示す。

7 考察

熟練した開発者がチームに 1 人いる状況を想定し、この開発者への依存度が高いようなシステムを洗い出すことにした。

熟練者の指標としては以下の視点を総合的に判断し、当てはまる上位 1 人を今回の考察の対象にした。またこれらの指標は OSS (Open Source Software) 開発プロセスに関する研究 [4] におけるコア開発者の指標を参考にした。

- アンケートの対象とする各サブシステムに直近 1 年間最も多くデプロイした開発者
- 新機能の実装やコードベースの管理に多く取り組んだ開発者
- 開発タスクの起票から本番環境へのデプロイ速度が早い開発者

属人性の高さとしてはシステム A やシステム B のフロントエンド領域などにおいて熟練者への依存度の高さが (TODO: ここに強調の語彙) だが、システム G のように差が 0.4 ポイントしか見られないようなものもある。またシステム K やシステム H のバックエンド領域など熟練者、平均ともに理解度の低いものも見受けられ、これは開発組織内で誰も詳細を知らないシステムとなっていると考えられる。

この結果を開発組織内に対し討論する機会を設けたところ以下のような意見が出た。

- 小規模なチームでは特定の人物へ依存しすぎることがありえる- 熟練者への依存度を下げたい という思いはあるが何から始めたらいいかわからない状況がある。 good First issue とかを用意してはどうか- 出てきた理解度の平均、熟練者への依存度共に大きな認識の相違はなさそう

Looker の理解度の平均が高いが、これはアドプラのエンジニアがみんな広告関連で分析のために Looker を使うためであり、それがアンケートの結果にも現れている

属人性が低いシステムの特徴 hoge 高いシステムの特徴 hoge

8 まとめ

・アンケートによって執筆すべきとされるシステムを洗い出すことが出来た・ただブルーム・タキソノミーの 6 分類を学習者に評価してもらうことによる同じ数値でも認識の相違がある・これは hona が書いたので hona が認識出来ていない問題はわからない

認識と相違ない数値を可視化できたのは意義がありそう

今後の展望を書く? - ドキュメントの足りてる具合を数値化- 何がわからないかを自由記述方式のボックスとかを用意することで具体的な FB 内容をヒアリン

^{*2} これより前にアンケートの回答を円滑にするためにシステムに対する設問を行っているが、本稿ではブルーム・タキソノミーを元に作成した設問のみ扱う

^{*3} アンケートの選択肢には「認識・記憶できる」の前の段階として「認識・記憶できない」を追加した

^{*4} アンケートでは以下の例に追加でもう 2 個ほど例示がある

グ- 定期的なアンケートとフィードバックによって改善を回す- 知識の断絶を防ぐためにこの手法を用いることが出来るかも- 理解度を FB したことによってドキュメンテーションが改善されるかどうかを比較する- 何らかの意味が現れていそうなデータが出て、それに意味付けをしたが実際に FB で改善されたのかという検証をしたい (この研究が最終的に到達しうる点)

参考文献

- [1] '小沢 暢, 松野 裕': ソフトウェア開発プロジェクトにおける引き継ぎプロセス及びドキュメント作成手順の提案と評価, 実践的 IT 教育シンポジウム rePiT 論文集, Vol.2023, No.0, pp.93-100, 2023.
- [2] '金田 茂裕': 教授者の課題知識と学習過程知識が教授学習法の望ましさを判断に及ぼす影響, 教育心理学研究, Vol.70, No.4, pp.333-346, 2022.
- [3] 'Anderson, Lorin W., Krathwohl, David R., Bloom, Benjamin Samuel': A taxonomy for learning, teaching, and assessing: a revision of Bloom's taxonomy of educational objectives, 2001.
- [4] 'Audris Mockus, Roy T. Fielding, James D. Herbsleb': Two case studies of open source software development, ACM Transactions on Software Engineering and Methodology, Vol.11, No.3, pp.309-346, 2002.
- [5] '宮原 史, 堤 盛人': 戦略的な人材育成の実現に向けた道路橋を維持管理する技術力の解明の試み ―ブルーム・タキシソミーの応用―, 土木学会論文集 F 4 (建設マネジメント), Vol.76, No.1, pp.14-28, 2020.