

ソフトウェア関連文書の品質改善

小川清, 齊藤直希
名古屋市工業研究所

(ogawa.kiyoshi,saito.naoki)@nmiri.city.nagoya.jp

August 13, 2012

Abstract

ソフトウェアと文書の作成, 編集および教育の取組みの経験を基に, ソフトウェア関連文書の品質改善を検討する。ソフトウェア関連文書の品質改善を行うにあたって, ソフトウェアの品質特性の枠組みに文書の特性を当て嵌めた。事例でソフトウェア品質副特性で分類し, 改善の方向を考える。

1 はじめに

ソフトウェア関連文書は, ソフトウェアを含むシステムの制約, 設計, 仕様, 利用, 保守, 廃棄に関わる文書とする。システムをハードウェア, ソフトウェアに分類すると, 文書はソフトウェアに分類できる。ハードウェア, ソフトウェア, サービスという3分類にすると, 文書はソフトウェアとサービスに分類するものがある。例えば, システム設計書はソフトウェアに分類できる。利用者手引きはサービスに分類できる。

本研究では, ソフトウェア関連文書をソフトウェアの品質指標にもとづいて分類することを検討する。検討対象は電子化する文書のみとする。手書き資料も, スキャナ, 写真などで電子化して管理することを前提とする。第二章で先行研究を確認し, 第三章でソフトウェア品質体系との関係を示し, 第四章で事例を検討する。最後にまとめと今後の課題を記す。

2 先行研究

2.1 日本語情報処理

日本語文書を電子的に作成するには, 日本語入力方式をはじめとする日本語情報処理 [1] 技術が課題であった。1990年代に, 電総研 (現在の産総研) の多言語処理 [2], NTTの日本語分析 [3], 文字鏡研究会における漢字の同定 [4] など急速に体系的になった。正確な文書を作成するための道具立てが揃って来ている。

2.1.1 機械翻訳

佐良木氏の提案による英日機械翻訳ソフトウェア [5] では, 複文の処理で意味を保存する方式を提案している。[6][7]

日本語処理の課題には, 日本語が膠着語であるため, 単語の分解が一意にできるとは限らないことがある。そのため, 英語から日本語への変換は簡単であるが, 日本語から英語への変換が難しい。機械翻訳では, 日英翻訳をするとき, 英日翻訳をして, 再現率を見て評価することがある。

2.2 ソフトウェア品質

ソフトウェア品質は, ISO/IEC 9126[8] での定義に基づき, ISO/IEC 25000[9] シリーズで体系化している。具体的には, ソフトウェア品質評価ガイドブック [10] で詳細の解説がある。ISO/IEC 9126 では, 7つの品質特性を定義し, Table 1 のように副特性を規定している。

Table 1: ソフトウェア品質

特性	副特性
F 機能性	F.1 合目的性, F.2 正確性 F.3 相互運用性, F.4 セキュリティ
R 信頼性	R.1 成熟性, R.2 障害許容性, R.3 回復性
U 使用性	U.1 理解性, U.2 習得性 U.3 運用性, U.4 魅力性
E 効率性	E.1 時間効率性, E.2 資源効率性
M 保守性	M.1 解析性, M.2 変更性 M.3 安定性, M.4 試験性
P 移植性	P.1 環境適応性, P.2 設置性 P.3 共存性, P.4 置換性
Q 利用時の品質	Q.1 有効性, Q.2 生産性 Q.3 安全性, Q.4 満足性

2.2.1 品質作り込みガイド

品質作り込みガイド [11] は, ソフトウェア品質評価ガイドブックで記載している評価をどのように使うかという視点を提供している。具体的には, 品質を輪郭 (profile) と

いう考え方にに基づき複数の指標の集まりとして表現している。システムの障害影響評価を行って、システムの目標を決めて行くという基本的な考え方は、多くのシステムに共通した枠組みとして利用できる。

文書とソフトウェアに関する事項は、「ドキュメントボリューム」と「ドキュメントバランスメトリクス」として、文書ごとに定義している。

「ドキュメントボリューム」は、参考値がなく、どのようなシステムの場合には、全体の記述量がどれくらいがよいかについては言及していない。「ドキュメントバランスメトリクス」では、要求仕様書は、全体の記述量、対象ユーザとその使い方に関する記述、動作環境に関する記述量、主な機能に関する記述量、安全に関する記述並びに非機能に関する記述量、システム全体構成に関する記述量、例外処理に関する記述量の7項目について、それぞれ参考値を示している。参考値は、百分率である。例外処理に関する記述量は、ものによっては全体の半分を超える場合もあり、具体的に何の例かがわからないと参考にならないかもしれない。

2.3 あいまいな日本語表現考

「あいまいな日本語」について阿部圭一氏の講演があり、あいまいな日本語表現考 [12] という資料をいただいた。あいまいであることが文書の最大の課題であろうか。

例えば、書籍としての「あいまいな日本の私」[13] と「美しい日本の私」[14] とを比較する。川端康成によるノーベル文学賞の答辞が「美しい日本の私」である。大江健三郎の「あいまいな日本の私」が、「美しい日本の私」に対する見解の表明である。大江は、美しいという積極的な表現に対して、曖昧という積極的な表現で何を示したかったのであろう。

文書において、「あいまい」なことと、「美しい」ことのどちらの視点が重要であろうか。

日本の美しさを国際的に紹介することは、日本文学の国際展開において重要であり、川端も大江もノーベル賞という十分な成果を上げている。「あいまいな日本の私」は、社会的、個人的な事情により、必ずしも明快に表現できない存在としての日本文化を提示しているのではないか。もし、曖昧性が日本の美しさの一要素であるのなら、あいまいであることを放棄することは、美しいことを放棄する可能性があり、品質を下げる可能性がある。

2.4 正確な文章の書き方

山本和彦氏による「正確な文章の書き方 [15]」という提言は、技術文書の基本的な課題を網羅している。ここでは文書作成の出発点として利用する。構成は次の通りである。
() 書きは本論文中で用いる略号である。

1. 伝えたいこと／あふれる思い (内容)
2. 正確さ／曖昧さの排除 (正確)
3. 豊かさ／軽やかさ (豊富)

4. バランス感覚／素直さ (均衡)
5. 内容の構成 (構成)
6. 思いきり／吟味する (吟味)

正確さ／曖昧さの排除において、「カタカナの乱用は避け、なるべく適切な日本語で置き換えましょう。」と提唱している。国立国語研究所における言い換えの提案 [16] と同じ文脈であり、プログラマだけでなく利用者にとって正確な文章は何かに着目している。JIS では、手引き [17] に類似の指摘がある。正確性に続いて、豊かさ／軽やかさ、バランス感覚／素直さを検討しているところが、美しさに繋がるものであると理解できる。内容の構成については、例えば国際規格の文章の書き方は、ISO/IEC Directives part2 [18] にある。JIS では、JIS Z 8301 [19] が内容校正を定義している。

3 品質分類の展開

3.1 品質特性と文書品質

品質特性を文書品質と対応づける。

相互運用性は用語の包含関係の矛盾の有無が大きな要素となる。障害許容性は文書そのものの障害を考える。運用性は文書の運用のし易さを定義する。試験性は試験記述の書き易さで定義する。互換性は異なる言語間（日本語と英語）で考えるとよい。利用時の品質は、サービスに分類する。例えば、正確な文書の書き方の6項目は、次のようなソフトウェア品質副特性との関係を考える。

Table 2: 文書品質とソフトウェア品質副特性

文書品質	副特性
内容	F.1 合目的性
正確	F.2 正確性
豊富	U.4 魅力性
均衡	U.1 理解性
構成	M.1 解析性
吟味	U.2 習得性

Table2 は、「正確な文章の書き方」の各要点は、異なる副特性と結びついている。また、文書作成上必要な、合目的性、正確性に留まらず、魅力性、理解性、解析性、習得性までの幅広い内容を含んでいると考えることができる。実践的な文書作成の指針として有効であることを想定できる。

西暦 2000 年問題 [20] のように、桁の制約による処理の不完全さを経験してもなおかつ二桁処理のままのものがあるなど文書の習慣がプログラムの合目的性と合わないこともある。

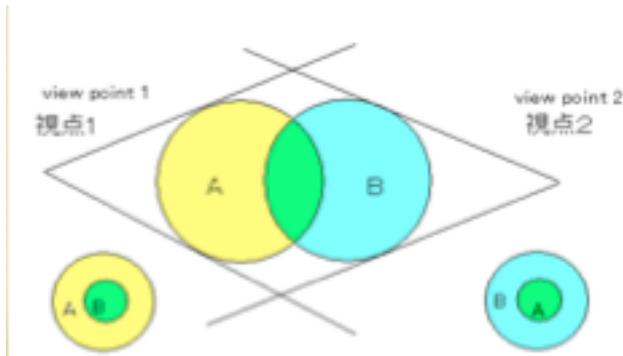


Figure 1: 視点

3.2 視点

立場によって意味が異なることと、意味があいまいであることを混同することがある。一つの言葉は、それぞれの立場では一つの意味を指しているも、立場の違いによって2つの意味を指すことがある。Figure1で、淡い方を1、濃い方を3、その中間を2とする。例えば、図のように、右から見れば中心がAで3。全体がBで2。左から見れば中心がBで3。全体がAで1。

右から見ればAが3、Bが2。左から見ればAが1、Bが3。右から見れば中心が3で全体が2。左からみえれば中心が3で全体は1。この事例では、中心が3であることだけが共通している。

このように、AとBとの関係では、左右で部分と全体の関係が逆転する。

これは、曖昧ではなく、正しいが、立場を消すと誤りになる。立場を消すと矛盾することも、立場を明記すれば曖昧でもなく正しい。

どちらの立場でも成り立つように構成する表現方法がある。限定子をつけて表現することによって、別のものとして記述する。

どちらかの立場だけで表現し、後で変換する方法もある。どちらかの立場で成り立つ場合には、どちらの立場を優先するかで2通りの方法がある。

図で示すと簡単なことを、言葉だけで説明するのが難しい例でもある。

文章においても、発話者の主客の立場によって認識と表現の関係の検討がある。[21][22][23]

図、論理式を含む論理言語、プログラミング言語で表現することによって、仕様を明確にするとよいのだろうか。

3.3 用語の木

用語の木 [24] は、概念の上下関係を示すものである。

2つの概念だけでも、用語の木を持つことが大事であることが分かる。実際のシステム設計では、立場の違いによって用語を整理することが大切である。

文章を理解する上で、用語の概念の上下関係が逆転する場合に、一面的な理解では、立場ごとに複数の用語の木を持つことが重要である。[25]

システム設計では、設計者と利用者ごとに用語の木を持つ。一つのシステムの中での略号でも、例えばRという記号が、RGBにおける赤と、RLにおける右という意味の2つの意味を持って表現していることがある。

3.4 図示

文書品質では、文字による品質だけでは十分でない。表のように文字を二次元に配置したものも有効である。図、写真などの文字ではないものの質をどのように測るかが重要である。

例えば、Uppaalのように図で入力して状態遷移の整合性を確認できれば、文字で入力するよりも理解性が高いと想定できる。

Alloy, SPINのように文字で入力しても、図示部分があると、入力したものについての理解性が高くなると想定できる。

このように、文字と図の相互変換性は、ソフトウェア品質の中では陽に規定していない。

図では、対称性をシステムの状態、設計の複雑さの判定の鍵として用いることを検討している。[27][28]

4 事例研究

文書の品質を考慮する上で、まず品質副特性を当て嵌める。品質副特性が同じものを集めることにより、文書の改善のきっかけ、方向性を分かり易くすることを検討する。

また、視点(立場)、用語の木、図示の3つの項目を考慮して文書改善事例を整理する。

4.1 責任校正

印刷業 [29] においては、印刷物の校正に内容が分からない人も分担することがある。責任校正では、内容的な校正よりも形式的校正が主となることがある。

責任校正を検討するのは、校正者が印刷側であり、執筆者ではないという関係にあるためである。ソフトウェア設計で言えば、設計者ではなく実装者が責任を持つことに類推できる。

責任校正が成立する背景について考察する。コンピュータによる原稿作成以前の時代にはいろいろな事象があった。例えば凸版(活版)印刷は、活字という一文字ごとに鉛で作った凸版の「活字」を縦に並べて、縦書きを実現するものである。活字の一面に、横に溝があるため、活字の向きを間違えることはあまりない。間違えて組み付けた後は、横は隣りの列の活字が並んでいるので、後からは見つけにくい。印刷で校正をする段になっても、意味の分かっている人は、意味を追いかけるので、文字が横向きになっても気が付かないことがある。

意味が分からない人間による校正は、横向いた文字を見つめることがある。そのため、意味が分からない人が校正する価値がある。

印刷屋が責任を持つ責任校正で、内容が分かっている人による校正と内容が分かっている人の校正があるとよい可能性を検討する。内容が分かっている人が校正で気が付くことについて記録する。

1. 句読点の抜け、重複。(F.2 正確性)

箇条書きのように、最後に読点を入れない場合を除いて、読点があることを確認する。句読点が重複して記載してあるものも確認する。

2. 横向き、逆さま (F.2 正確性)

文字が横向いていても、気が付かないことがある。左向き、右向き、逆向きの3種類ある。活字がないときに、文字とは違う向きにして井桁を表して、文字が今ないことを表すものがある。井桁のままになっているものも不具合である。

3. 字下げ (F.2 正確性) 文章の先頭を一文字、字下げしているのに、字下げしていない段落があったり、段落間に1行入れているときに、一行入っていないものを確認する。

4. 番号 (F.2 正確性) 引用、参考文献などで、引用の番号、順番と、参考文献一覧における番号、順番とが合っているかどうか。途中で追加があった場合に、どちらかが未修正の場合がある。

5. てにをは、ですます (U.1 理解性)

読んでいて、内容は分からなくても違和感を持つ場合がある。限定の「は」が2つあったり、目的の「を」が2つあったり。意味が分かる人でも意味が取りにくい場合がある。修正は著者に聞くこともある。

6. 固有名詞の綴り、スペル (U.1 理解性)

固有名詞には、複数の表現を取る場合がある。文章の中で複数の表現をとっていると分かりにくくなる。同じ名前が別の表現を取っていないかを確認する。ただし、似た名前の別のものがあるかどうかを、著者に確認してする。略号だけあって、フルスペルがないものも確認する。

責任校正においては、内容の校正が主眼ではない。そのため合目的性、魅力性は主たる対象にはなっていない。

4.2 L^AT_EX

名古屋市工業研究所で、印刷業界向けに L^AT_EX の研修 [30] を3年間行った。データベース検索技術者試験 傾向と対策 [31] を L^AT_EX で作成して入稿した経験と技術を、広く地元の印刷関係会社にも伝達するためである。

L^AT_EX のようなオープンソースの技術の伝搬や訓練を受けるところは他になく、印刷業界を中心に受講があった。ソフトウェア会社の利用手引き作成担当の受講もあった。

L^AT_EX による編集の利点として、文字で入力して二次元に組版する点がある。特に、数式の二次元配置は美しく、

理解性を増す。L^AT_EX は、著者が入力しても、印刷担当者が入力してもよい。

美しい文書を作成するためには、いくつかの規則を決めておくといよい。名古屋市工業研究所で技術文書の研修で手引きの原案として利用してきた規則は次の通りである。規則に品質副特性をあてはめ、類似のものを並べ直した。

1. 文字の大きさは5種類 (論理階層が5階層くらい) (U.1 理解性)
2. 文字の種類 (font) は、アルファベット、日本語でそれぞれ3種類 (U.1 理解性)
3. 略号にはフルスペルをつける (U.1 理解性)
4. 赤は危険または注意喚起のみにする。(会社のロゴなどに赤を使っている場合は注意喚起の一種と看做す。) (U.1 理解性)
5. 意味の塊ごとに余白があり、視覚的に意味の違いが区分できる。(U.1 理解性)
6. 要点は箇条書きにする。(U.1 理解性)
7. 出典 (参考文献, URL) を明記する。(U.2 習得性)
8. 日本語は斜体を用いない (斜体用字形を除く) (U.4 魅力性)
9. L^AT_EX 以外で作ったものを L^AT_EX にしてみても、段組みの違いから、より美しい状態を理解して、どちらの道具でもより美しく作る。(U.4 魅力性)

日本語は斜体を用いないのは、車体用字形でないと美しくない為である。美しい斜体を設計すれば魅力的になる可能性はある。

L^AT_EX では、章、図表の番号付けは自動生成であるため、正確性は確保できる。しかし、章、節の深さの指定は人手である。参考文献の順番は、登場順に自動変更してくれない処理系を利用している。

L^AT_EX では、理解性と魅力性を高めるのに役立つことが分かる。

L^AT_EX, 索引を作成し、用語の出現頻度、分布を確認することができる。文章の句点が多すぎたり、少なすぎたりするものを測って、文章全体で均一の取れたものにする必要がある。

4.3 プログラミング教育

プログラミング教育において、10種類くらい文献を参考に、有益なものを入力する方法を取った [32][26]。繰り返しいいものを入力することによって、要点を身につける。書道であれば、写経のような方法である。教材などで公開しているソースコードを、コンパイルしてみると、さまざまなエラーがでることがある。原因を分類して、対策を立てて行くと、体系的な理解に進む場合がある。[33]

プログラミング教育において「プログラムは詩のように」と、文法を教えずに、題材としてある関数などを組み

合わせて書くことを試す。書いたプログラムでシンタックスエラーがでたら、文法書を参考に直すことを繰り返す。

このどちらも母語教育方式といい、文法を教えるのではなく、繰り返し言葉を使うことによって言語を習得していく方式である。

この方法でよく遭遇する事象を、品質副特性に当て嵌めてみる。

1. 文法誤り (F.2 正確性) 文の終わりの記号、対応する括弧など、言語で定義している文法に違反している。文法が自然に身に付く。演算子の記号の使い方、場所などの間違いが文法誤りとなることもある。文法を先に教えて、文法誤りのエラーを出さずに過ぎていくと、文法誤りの直し方の訓練ができない。よく打ち間違えをする人は、文法誤りになる可能性もあり、自然に文法が身に付くかもしれない。

2. 処理不能文字 (F.2 正確性)

文字コードの違いなどにより、処理できないコードを含む場合。文法誤りに分類することもできるし、未定義の変数または範囲外に分類できるものもあるかもしれない。

3. 範囲外 (F.1 合目的性)

代入文などで、想定外の代入をエラーにする場合がある。異なる型の代入を型変換して代入する言語の場合には、キャストなどを陽に行って型変換の目的を明示するか、型検査ソフト (lint など) を用いて、異なる型代入の有無を確認するとよい。アンダーフロー、オーバーフローなどがエラーとなる場合と、そのまま代替値で先にすすむ場合とがある。

4. 意味の間違い (F.1 合目的性)

例えば、0 による割り算。0 による割り算を文法誤りとするか、範囲外の代入とするか、意味の誤りとするかは言語の仕様によるかもしれない。計算結果にもとづいて、誤差などにより実行時にエラーとなる場合に、事前に検査するソフトウェアもある。どこまでが文法で、どこからが意味かは、言語定義による。

5. 未定義の変数 (U.1 理解性)

言語の文法で定義していないし、利用者が宣言していない用語。文法で定義している用語に変更するか、自分で定義する。定義の仕方を覚えることができる。飛び先、分岐先がないなどもこの分類に入れる。未定義の参照、ファイルがないなどもこの分類に入れることができる。ただし、参照範囲が範囲外の場合は「範囲外」に含むこともある。いくつかのインタプリタでは、未定義の用語を文脈依存で解釈して処理してくれる処理系ではこのエラーはでない。定義してから用いるとよい場合と、定義せずに使って文脈依存で理解するとよい場合があるかどうかを検討事項となる。

逆に、宣言だけあって使っていない場合に警告を出してくれるものもある。

ソフトウェアに該当する事項を、文書に翻訳してみると次のようになる。文法誤りは、自然言語の文法誤り。未定義の変数は、文章において定義していない用語の利用。参考文献などで用語定義しているものを利用してよいが、参考文献間での用語定義の矛盾を確認していない場合がある。自然言語の文書で、参考文献が一杯あると、未使用の用語が多く、何を使っているかの索引があり、その用語がどの参考文献で定義しているかが分かるとよい。ソフトウェアでは実施していることを、自然言語でも実施するだけで分かり易くなるか、調べ易くなるかも。

範囲外の代入は、文書の場合には確認することが難しい。そのため、形式手法とよぶ記述で確認することがある。意味の間違いは、自然言語では間違いなのか立場の違いなのかを判定するのがむづかしい。用語の集合関係が立場によって違う。

文法上の課題は、同じプログラムが同じ振舞をするかどうか。コンパイラなどの処理系や、CPU の違いによって振舞が異なることを、CPU、処理系の発展のために許容している場合がある。それに対して、同じプログラムが同じ振舞をするように、空間的な部分集合を定義しているものが MISRA-C[34] である。論理回路において、順序的に同じ振舞をするように、時間的な部分集合を定義しているものが STARC RTL 設計スタイルガイド [35] である。

プログラムを図から自動生成する場合がある。UML の図から C 言語などを自動生成する方法である。また Matlab/Simlink から C 言語を自動生成する方法を制御系で利用することもある。いずれも、図で設計し、文字による文書 (プログラム) は自動生成するものである。

4.4 ソフトウェア導入

オープンソースの世界では、しばしば導入の失敗、コンパイルの失敗により利用を断念することがある。これは、ソフトウェアと文書の品質上の課題だと理解することができる。該当する文書または仕組みの自動化が機能していれば、うまく導入できる。

ソフトウェア導入の鍵となる事項について品質副特性を当て嵌めてみる。

1. 導入の制約条件、導入手順の明確化 (F.1 合目的性)
2. 導入ソフトウェアの構成管理 (M.1 解析性)
3. 被導入システムの構成管理 (M.1 解析性)
4. 有償ソフトウェアとの比較 (U.1 理解性)
5. 有償ソフトウェアとのデータ交換 (P.4 置換性)

機能か、文書のどちらかが十分であれば、解決する。

4.5 読書記録

ソフトウェア関連文書以外を例に、品質副特性を当て嵌めてみる。

題材としては、機能安全事業におけるスキル判定の基礎となる知識判定に用いた読書記録を題材 [36] とする。

読書感想文または読書記録の覚えをあるところを書いて、booklog, amazon, mixi に転載するにあたり、次のような校正、添削をしている。読書の覚えには、amazon に目次や書誌がないばあいには、書誌を優先して記録することがある。必ずしも書評を書いている訳ではない。

1. 否定文は肯定文に
「ふりがなが少ない」→「ふりがなが少しある。初出の漢字にすべて振ってもらえると嬉しい。」
2. 複文は単文にして、接続子を削除
「面白いが、複雑すぎてわかりにくい」→「面白い。複雑でわかりにくいところが一部ある。」
3. 思い込みの形容詞、助詞、接続子などは削除
「気が付いたら直すだけで、誤字を探すだけのために読まない。」→「気が付いたら直す。文章の入れ替えのために読んでみるのを主眼にする。」
4. 誤字脱字の訂正気が付いたら直す。文章の入れ替えのために読んでみるのを主眼にする。
5. 主題以外の内容は後書き p.s.) に
6. 主観記述から事実記述へ
7. 自分の興味の順から、他人の知りたいかもしれない順に、文を並べ替える
8. カタカナ語を漢字表記、English 表記または漢字 (Chinese Character) 表記に
9. いろいろな題材がある場合には、箇条書きにする。
10. 代名詞を具体的な名詞に置き換える。
「このソフトウェアに関連する技術」→「PAT Transer に関連する技術」どこから読み始めても、意味が通る。優先順位付けなどで行を入れ替える際に順番に依存しない文になっている。
11. 略号にはフルスペルをつける
12. 理由がわからない
13. 意味の幅があり、範囲がわからない。

これらの作業を品質指標との関係を示し、副特性ごとに整理しなおす。

1. 単文率、複文率 (M.1 解析性) : 単文率 + 複文率 = 1
2. 形容度 (U.4 魅力性, M.1 解析性) : 形容率 < 名詞または動詞に形容詞または副詞などがついてくるかの比率 >, 重み付き形容率 < 2 重に形容している場合は 3 倍または 4 倍などの重みをつけて測る > 魅力性はあげるが、解析性を下げる場合がある。

3. 肯定率、否定率 (U.4 魅力性) : 肯定率 + 否定率 = 1
4. 集中度 < 話題の比率 >, 散漫率 < 話題以外の比率 > (U.1 理解性) : 集中度 + 散漫率 = 1
5. 主観率 < 主観的な記述の割合 >, 客観率 < 主体以外が事実確認できる事項の割合 > (U.1 理解性) : 主観率 + 客観率 = 1
6. 優先順率 (U.1 理解性) < 優先順位が高いものが先に記述しているかの割合。優先順位順を 1, 逆順を 0 として、間の計算を重み付けして求める >
7. カタカナ語比率 (U.1 理解性) < 全体の単語のうちのカタカナ語の割合。1 文に 1 つが上限。2 つ以上入れると分かりにくくなる。 >,
8. 箇条書き率 (U.1 理解性) < 文章の中の箇条書きの割合。箇条書きが 9 割を超えると、文章というよりはデータと看做す場合もある。 >
9. 圧迫率 (U.1 理解性) < 意見を押し付けようとしているかどうかの割合 >
10. 誤字率 (F.2 正確性) : 全文字中誤字の割合。
11. 指示語率 (F.2 正確性) < あれ、これ、それなどの代名詞の割合。1 文に 1 つが上限。 >
12. フルスぺル率 (F.2 正確性) < 略号のフルスペルを初出で記載している割合 >, 略号率 < 文章中の略号の割合。3 割を超えると分かりにくくなり、5 割を超えたら、人工言語と思った方がよい >
13. あいまい数 (F.2 正確性) : 範囲、程度などが分からない用語で、同一段落で限定していないか、例示がない数。分母を決めにくいので比率で表さない。全く同一の曖昧な語は、重複して数えないが、出現数を数える。
14. 根拠不明数 (U.2 習得性) : 同一段落に根拠がないか、根拠を示す参考文献がない数。分母が無限大かもしれないので比率では表さない。比率にする場合には 1 ページあたりの根拠不明数を出すとよい。

理解性、正確性に関するものが多い。品質特性が同じ分類のものをまとめると、文書の特性が比較しやすくなる。形容度のように、よい面を捉えれば魅力性があがるが、悪い面を見ると解析性が下がることがある。文書の一つの技法が、一つの特性を上げて、他の特性を下げる場合がある。そのため各副特性ごとの値を輪郭 (priflie) として整理し、文書の属性として定量化して改善を行う。

4.6 技術文書の評価

技術文書を、これまでに例示のあった文書の性格に基づいて評価してみる。利用した基準は、機械翻訳にかける前の日本語の前処理として手作業でおこなっていた次の事項である。

1. 複文になっている。
2. 接続詞がなくても意味が取れるのに接続詞があるか。
3. 肯定系で書けるのに否定形で書いているか。
4. 専門用語でないのにカタカナ語を使っているか。
5. 受身を使っているため主語が分かりにくくなる。
6. 根拠, 理由を書いていない。
7. あいまいな表現を使っている。

副特性ごとに分類して, 2つの文書を 100 単位について測定してみた。

Table 3: 文書の輪郭の例

分類	文書 1	文書 2	品質指標
カタカナ語	10	15	U.1 理解
受身	4	1	U.1 理解
根拠	4		F.1 合目的
あいまい	4	4	F.2 正確
否定	2		U.4 魅力
複文	4		M.1 解析
接続詞	1		M.1 解析

文書 1, 文書 2 とも, カタカナ語の多用, 受身などがある。文書 1 は, 根拠を示さなかったり, 曖昧な表現を取り, 仕様としての課題がある。また, 複文, 接続詞の利用など解析性を下げる傾向もある。

これらの 7 項目を検査しておく, 機械翻訳で処理が通る確率が高くなっていた。

ところで, プロセス改善ナビゲーションガイドベストプラクティス編 [37] において, JAXA の取り組みの紹介において HAZOP の取組みがある。JAXA/IPA 主催 8th WOCIS において, HAZOP の講演の提出資料を HAZOP で分析した結果を添付 [38] した。文書を発表者と聞き手のそれぞれの立場に応じて HAZOP で分析することにより, 著者のみの視点での評価以外を, 量的に実施することができる。

文書は, 文章だけでなく, 図, 写真, 動画などのファイルを含む場合に, 文字以外の価値評価が重要である。なお, 本論文は, この視点での評価は実施していない。

5 まとめと今後の課題

ソフトウェア品質指標にもとづき文書評価する枠組みで品質副特性を当て嵌めてみた。文書として未熟なものは, 合目的性に欠け, 正確性, 理解性が乏しい事と, 解析性, 魅力性が不十分な場合がある。他の副特性が, 文書品質の中でどのような役割を果たすかは, より品質の高い文書での検討が必要である。

個々の文書全体の輪郭 (profile) を作成した比較を検討しているが, 全面的な書き直しには関係者の合意が必要なため, 文書品質が良くなるだけでは書き直せない。ISO の OSI の経験のように論理的に正しい文章には価値が少ない場合がある。複数の立場の文章が混在して矛盾を含んでいる状態の文書の法が価値がある場合もある。

今後は, ソフトウェア関連文書以外の品質副特性との対比を行いたい。またソフトウェア例えば日本の詩, 具体的には短歌集の今昔秀歌百撰 [39] を分析中で, これらの副特性がある。また提出論文は事前に著者と読者の 2 つの視点で文書評価し, 定量化するとともに HAZOP 分析をしておくことよきことが分かった。用語の木を作り, 図示するとよりよい文書になるが, これらをどのような品質指標として組込んでいくかが今後の課題である。

References

- [1] 日本語情報処理, Ken Lunde, ソフトバンククリエイティブ, 1995
- [2] 多マルチリンガル環境の実現—X Window/Wnn/Mule/WWW ブラウザでの多国語環境, 錦見 美貴子, 戸村 哲, 桑理 聖二, 吉田 智子, 高橋 直人, 半田 剣一, 向川 信一, プレンティスホール出版, 1996
- [3] 日本語語彙体系, 岩波書店, 池原悟, 宮正弘, 白井論, 1999
- [4] 今昔文字鏡 Ver.1.0, エーアイネット, 1997
- [5] PAT-Transer, ノヴァ, 1998
- [6] WRAPL による機械翻訳のためのテキスト編集と翻訳精度の向上, 佐良木昌加藤輝政小川清, 言語処理学会第 3 回大会, 1997
- [7] 英語複文の構文解析と編集, その論理と方法, 加藤輝政小川清佐良木昌, 電子情報通信学会 信学技報 Vol.97 No.199 pp.65-70, 1997
- [8] ISO/IEC 9126. Information technology – Software product evaluation – Quality characteristics and guidelines for their use, 1991
- [9] ISO/IEC 25000 Software Engineering – Software product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE, 2005
- [10] ソフトウェア品質評価ガイドブック, 東基衛, 日本規格協会, 1994
- [11] 組込みソフトウェア開発向け品質作り込みガイド, IPA/SEC, 翔泳社, 2008
- [12] あいまいな日本語表現考, 阿部圭一, ASDoQ, 2012
- [13] あいまいな日本の私, 大江健三郎, 岩波書店, 1995

- [14] 美しい日本の私, 川端康成, 講談社, 1969
- [15] 正確な文章の書き方, IJ 技術研究所, 山本和彦, 2009
- [16] 分かりやすく伝える 外来語 言い換え手引き, 国立国語研究所, 2006
- [17] JIS 等原案作成のための手引き第 12 版, 日本規格協会, 2011
- [18] ISO/IEC 専門業務用指針 (ISO/IEC Directives) 第 2 部 国際規格の構成及び作成の規則 (Part 2 Rules for the structure and drafting of International Standards), ISO/IEC, 2011
- [19] JIS Z 8301 規格票の様式及び作成方法, 日本規格協会, 2011
- [20] コンピュータ西暦 2000 年対応の標準化におけるデータ, 用語, 処理, 試験, 小林武史, 大角泰久, 小川清, 言語処理学会年次大会 2000
- [21] 日本語はどういう言語か, 三浦つとむ, 1956
- [22] 佐良木昌編, 言語過程説の探求 第一巻, 明石書店, 2004
- [23] 認知文法論, 山梨正明, ひつじ書房, 1995
- [24] リアルタイム組み込みソフトウェアの用語の木, 小川清, 齊藤直希, 吉川直邦, 伊藤正樹, 後田直樹, 藩建華, 情報処理学会研究報告. SLDM, 2003
- [25] ソフトウェアプロセスアセスメントの用語構造と診断, 小川清, 「言語・認識・表現」研究会, 1999
- [26] ソフトウェア開発工程における用語構造と翻訳辞書作成過程における課題, 小川清, 言語処理学会年次大会 2000
- [27] デジタル設計工学における言語の対称性についての検討, 小川清, 齊藤直希, 渡部謹二, 堀武司, 「言語・認識・表現」研究会, 2008
- [28] 形式手法における言語の対称性についての検討, 小川清, 齊藤直希, 渡部謹二, 「言語・認識・表現」研究会, 2009
- [29] ビジュアル博物館 第 4 8 巻 文字と書物, カレン・ブルックフィールド, 同朋舎, 1999
- [30] L^AT_EX 美文書入門, 奥村 晴彦, 技術評論社, 1991
- [31] サーチャー試験 データベース検索技術者試験 傾向と対策, 小川清, 安藤敏彦, 紀伊國屋書店, 1993
- [32] プログラミング言語教育の自然言語教育方式の応用, 小川清, 「言語・認識・表現」研究会, 1999
- [33] パソコンで学ぶエラー対策テクニック—新しいマン・マシン・インタフェースを考える, 伏見論, CBS 出版, 1986
- [34] 組込み開発者におくる MISRA - C:2004—C 言語利用の高信頼化ガイド, SESSAME/MISRA-C 研究会, 日本規格協会, 2006
- [35] STARC RTL 設計スタイルガイド Verilog-HDL 編, 2003
- [36] 安全分析, 状態記述と形式手法に着目した安全教育とスキル, 堀武司, 小川清, 齊藤直希, 渡部謹二, 森川聡久, 服部博行, 安全工学シンポジウム, 2009
- [37] プロセス改善ナビゲーションガイドベストプラクティス編, IPA/SEC, 2006
- [38] 一人 HAZOP を組み合わせた効率的な分析作業, 小川清, JAXA/IPA, 8th WOCS, 2011
- [39] 今昔秀歌百撰, 市川 浩, 谷田貝常夫, 特定非営利活動法人 文字文化協会, 2012