

修士論文

題目 ソフトウェア開発プロジェクトにおける
感情極性推移の分析

主任指導教員 水野 修 教授

京都工芸繊維大学大学院 工芸科学研究科

情報工学専攻

学生番号 16622027

氏名 中川 要

平成30年2月9日提出

学位論文の要旨（和文）

平成 30 年 2 月 9 日

京都工芸繊維大学大学院
工芸科学研究科長 殿

工芸科学研究科	情報工学専攻
平成 28 年入学	
学生番号	16622027
氏 名	中川 要 印

（主任指導教員 水野 修 印 ）

本学学位規則第 4 条に基づき、下記のとおり学位論文内容の要旨を提出いたします。

1. 論文題目

ソフトウェア開発プロジェクトにおける感情極性推移の分析

2. 論文内容の要旨（400 字程度）

自然言語処理の分野において感情推定が注目されている。これは一般的に、文章から筆者や話者の意見の極性を特定することに焦点を当てたテキスト分析である。近年では、ソフトウェアエンジニアリングの分野に感情推定を適用する多くの研究が行われている。本論文では、ソースコード管理に用いるリポジトリに蓄積されたコミットコメントに対して感情推定を行い、開発者の感情極性の推移とプロジェクトの特性について調査を行った。4つのソフトウェアプロジェクトに対して実験を適用した結果、成果物の特性や開発言語に基づいて2グループに分けた場合に、コミットコメントの感情極性推移において、それぞれの感情極性において有意な差が存在するところを発見した。また、コミットコメントに感情推定を適用する際に不都合が生じるコメントが存在することを発見し、その分析を行った。

Analysis of emotional polarity transition in software development project

2018

16622027

NAKAGAWA Kaname

Abstract

Sentiment analysis has attracted attention in the field of natural language processing. This is generally a text analysis focusing on identifying the polarity of the opinions of authors and speakers from sentences. Recently, a lot of research applying sentiment analysis to the field of software engineering has been conducted. In this thesis, we analyze the relationship between the transitions of the developers' emotional polarities of commit comments and the characteristics of the projects, by applying sentiment analysis to the commit comments stored in source code repositories. We apply this experiment to four open source software projects. As a result, we found that there are cases where there is a significant difference in their emotional polarities among those two groups of the four projects which divided based on the characteristics of the projects and mainly used programming language. We also found and analyzed that there are comments that inconvenience when applying sentiment analysis to commit comments.

目 次

1. 緒言	1
2. 関連研究	3
3. 研究設問	4
4. コミットコメントの感情推定	5
4.1 コミットコメントの抽出	5
4.2 感情推定	5
4.3 感情値のタグ付け	6
4.4 感情推移の可視化	7
4.5 意図しない影響を及ぼすコミットコメントの特定	7
4.6 特性の異なるプロジェクト間の感情極性	7
5. 適用実験	9
5.1 実験の手順	9
5.1.1 対象プロジェクト	9
5.1.2 コミットコメントの抽出	10
5.1.3 感情推定	10
5.1.4 感情極性推移の可視化	10
5.1.5 影響を及ぼすコミットコメントの抽出	10
5.1.6 プロジェクトグループ間の差の調査	11
5.2 実験結果	11
5.2.1 影響を及ぼすコミットコメント	11
5.2.2 感情極性推移	13
5.2.3 プロジェクト間の感情極性	14
6. 考察	20
6.1 研究設問に対する考察	20
6.2 妥当性の検証	22
6.2.1 オープンソースプロジェクトに適用した実験	22

6.2.2	バージョン管理システムの機能	22
6.2.3	感情極性推定手法	22
6.3	今後の課題	22
7.	結言	24
	謝辞	24
	参考文献	26

1. 緒言

ソフトウェアの開発をより効率的に行うため、様々な研究が行われている。その1つにリポジトリマイニングという手法がある。リポジトリとは、バージョン管理システムを用いたソフトウェア開発プロジェクトにおけるソースコードやコミット情報をまとめて保管することができるデータベースである。このデータベースを分析し、ソフトウェア開発における有益な知見を発見することがリポジトリマイニングの目的である。

リポジトリマイニングの研究の一環として、ソフトウェアリポジトリに含まれるコミットコメントなどのテキストデータに対して、テキスト分類やトピック分析などの自然言語処理の手法を適用する研究が進められている。

近年、自然言語処理の研究分野において、感情分析が注目されている。これは一般的に、文章から筆者や話者の意見の極性を特定することに焦点を当てたテキスト分析である。商品のレビューやソーシャルメディアなどに適用され、マーケティングや顧客サービスを効率的に行うことを目的に利用されている。

感情分析の手法をソフトウェアリポジトリ内のデータに適用することで、ソフトウェア開発プロジェクトに携わる開発者らの、開発作業時の感情推移と、彼らの作り出す成果物の特性などの間に存在する関係を発見することができるのではないかと考えられる。そこで本研究では、ソフトウェアリポジトリに蓄積されたコミットコメントに対して感情分析を行い、開発者らの感情極性の推移と成果物の特性との関係について分析を行う。

本研究では、リポジトリに記録されているコミットコメントを取り出し、それらに対して感情分析の手法を適用することで、それらのコメントが良い印象（ポジティブ）と悪い印象（ネガティブ）をどの程度含むのかについて分析を行う。また、成果物の特性が異なる複数のリポジトリに対して分析を行うことで、特性の異なるプロジェクト間において、感情推移の傾向の違いを調査する。さらに感情分析を行うにあたり、意図しない影響を及ぼし、不都合が生じる原因となるコミットコメントが存在するのではないかと考え、コミットコメントの分析を行う。

本論文の以降の構成を述べる。2章では研究の基礎となる関連研究について述べる。3章では本研究において設定した研究設問を述べる。4章ではコミットコメント

の感情推定の手法について述べる．5章ではコミットコメントの感情推定の手法を対象のプロジェクトに適用した実験の手順，及び結果について述べる．6章では実験の結果に対する考察，及び今後の課題について述べる．7章では本研究のまとめについて述べる．

2. 関連研究

ソフトウェア開発は、複雑なソフトウェアシステムを作成し維持する際に、開発者同士の共同作業となるケースが多く存在する。人間が他者と関わる際、喜びや悲しみなどの感情が引き起こされるが、これはソフトウェア開発の現場においても例外ではない。

ソフトウェア工学分野においても、感情推定を適用した様々な研究が行われている。Murgiaらは、ソフトウェア開発活動に感情が影響する可能性を示唆し、バグ管理システムのバグレポートに含まれる感情を自動的に分析するツールを作成した [1]。Wrobelは56人のソフトウェア開発者を対象に調査を行い、ほとんどの開発者はポジティブな感情を持っている時は生産性が向上し、ネガティブな感情を持っている時は生産性が向上しにくいという結果を示した [2]。Guzmanらはソフトウェア開発プロジェクトのためのWeb共有サービスであるGitHub[3]を対象に、異なるオープンソースプロジェクトのコミットコメントに対して感情分析を行い、用いられるプログラミング言語やコミットコメントが書かれた時間・曜日などと感情との関係を調査した [4]。

これらの研究を踏まえ、本研究では、オープンソースプロジェクトに参加する開発者のコミットコメントの感情極性に注目し、特性の異なるプロジェクト間におけるプロジェクト毎の感情極性の推移について、傾向の違いを調査した。

3. 研究設問

本研究は将来的に、プロジェクトの開発現場において、開発者らがもつ感情の推移傾向を特定し、プロジェクトの進捗停滞などの原因特定や予測を行うことにより、開発者の精神的負担を軽減することを目的としている。

この目的のために、本研究では、開発者の感情極性の推移とコミットコメントの間に関係が存在するかを調査する。

従って、本研究では次に設定する研究設問について調査を行う。

RQ1 コミットコメントから感情分析を行う際に不都合が生じるコメントが存在するか。

RQ2 特性の異なるプロジェクトにおいてコミットコメントの感情極性に差があるか。

RQ1 では複数のプロジェクトを選定し、コミットコメントから感情極性を抽出する。そして、感情極性推移を分析する際に不都合が生じると考えられるコミットコメントについて調査する。

RQ2 では複数のプロジェクトに対して特性ごとにグループ分けを行い、コミットコメントから感情極性を抽出する。そして、特性の異なるプロジェクトの間に、有意な差が存在するかどうかを調査する。

4. コミットコメントの感情推定

本研究では，以下の手順でコミットコメントの感情極性を推定し，その推移の可視化を行う．

1. ソフトウェアリポジトリからコミットコメントを抽出する．
2. 抽出したコミットコメントから感情推定を行う．
3. 全てのコミットに，対応するコミットコメントの感情極性をタグ付けする．
4. 感情値タグを用いて，プロジェクト毎に感情極性の推移をグラフ化する．

また，対象プロジェクトにおけるコミットコメントのうち，感情推定やその分析に際して意図しない影響を及ぼすコミットコメントを調査する．

さらに，特性の異なるプロジェクト間の感情極性推移について調査し，プロジェクトの特性と感情極性推移の関係を調査する．

4.1 コミットコメントの抽出

バージョン管理を行っているソフトウェア開発プロジェクトでは，ソースコードの更新をリポジトリにコミットする際，その更新内容についてのコメントを残すことが慣習となっている．それらのコメントをコミットコメントと呼ぶ．コミットに紐付けて管理されており，必要に応じてコミットに関する情報とともに参照することができる．

本研究では，Git リポジトリで管理されているオープンソースソフトウェアプロジェクトを研究対象とする．

python パッケージの `gitpython`[5] や `git` コマンドを用いて，本研究に用いるコミット ID，コミット時刻，コミットコメントを Git リポジトリから抽出することができる．

4.2 感情推定

前節で得られたコミットコメントを文章とみなして感情推定を行う．

本研究では，NLTK[6] に実装された VADER[7] の感情極性推定手法を用いた．VADER はマイクロブログのようなソーシャルドメインに用いられるテキストに対

して有効であることが分かっている。また、VADER による感情推定において、学習データは不要である。この手法では、入力した英語の文章に対して感情推定を行い、次の3つの感情極性を 0.0 から 1.0 の範囲をとる数値として出力する。

positive : 良い印象を与える度合い

negative : 悪い印象を与える度合い

neutral : 良し悪しの判断がつかない度合い

これらの三要素の総和は 1 となる。NLTK に実装された VADER によって感情推定を行う場合、これらの三要素に加え compound という値が同時に出力される。これは positive, negative, neutral を統合した数値であり、-1.0 から 1.0 の範囲をとる。

例えばある文章に対し、NLTK を用いて VADER の感情推定を行い、以下の結果が出力されるとする。

```
{'neg': 0.437, 'neu': 0.316, 'pos': 0.247, 'compound': -0.4215}
```

このとき、三要素のうちの negative の数値、つまり悪い印象を与える度合いの数値が他の二要素より大きく、また compound の数値がマイナスの値をとっていることが分かる。よって、この感情推定を行った文章は比較的悪い印象を与える文章であるといえる。

また本実験では、得られた感情極性の4つの値のうち、統合値である compound の値を感情推定の結果として採用した。これにより、後に行う分析や可視化に際して、処理を簡単に行うことができる。

4.3 感情値のタグ付け

前節で行った感情推定の結果を、ソフトウェアリポジトリのコミットに対してタグ付けし、参照を容易にする。

本研究では、バージョン管理システムである Git のタグ機能により、全てのコミットにそれぞれ対応するコミットコメントからの感情推定値をタグ付けする。このとき、感情値は1つのプロジェクト内において同値をとるものが存在すると予想されるが、Git タグの仕様により同値のタグを付与することはできない。そのため、感

感情値に番号を付随させたものをタグとして付与し、可視化や分析を行う際に、再度感情値のみに変換する方法を取る。

4.4 感情推移の可視化

前節で付与した感情値タグを用いて、ソフトウェアリポジトリ全体における感情極性の推移を可視化する。

Git のログ取得機能を用いて、コミットの感情値をタグから抽出し、同時にそのコミットが行われた時刻情報を取得する。

また取得した感情値データを、折れ線グラフとして出力する。グラフの出力には Python のパッケージである Plotly[8] を用いる。

4.5 意図しない影響を及ぼすコミットコメントの特定

対象のプロジェクトにおけるコミットコメントのうち、感情推定やその分析に際して、意図しない影響を及ぼす可能性のあるコミットコメントを、以下の手順で特定する。

1. 完全に同じ文章のコミットコメントの出現回数を数える。
2. 出現回数の多いコミットコメントの感情値を調査する。
3. 2の結果から、感情推定やその分析に際して何らかの影響を及ぼすコミットコメントを特定する。

本実験では、「何らかの影響を及ぼすコミットコメントである」と判断する基準として、以下の二項目を採用した。

- 感情値が `neutral` ではなく、`positive` または `negative` に偏っている。つまり、`compound` \neq 0.0 である。
- 該当するコミットコメントの出現回数が全コミット数の 1%以上である。

4.6 特性の異なるプロジェクト間の感情極性

まず、プロジェクトの成果物とプロジェクトの主な開発言語に基づき、特性の異なる複数のグループにプロジェクトを分類する。

このプロジェクトグループのそれぞれの要素であるプロジェクトについて、グループ間のすべての組み合わせにおいて、コミットコメントから抽出した感情極性に差があるかを調査する。これに基づき、特性の異なるプロジェクトグループ間に差があるかを調査する。

5. 適用実験

5.1 実験の手順

本研究では，以下の手順で適用実験を行う．

1. 対象プロジェクトを選定する．
2. コミットコメントを抽出する．
3. 感情推定を行う．
4. 結果の可視化を行う．
5. 意図しない影響を及ぼすコミットコメントを特定する．
6. 特性の異なるプロジェクトグループ間の差について調査する．

5.1.1 対象プロジェクト

本研究では次の4つのオープンソースプロジェクトのリポジトリを実験対象とする．現在も開発が継続している統合開発環境 (IDE) の開発プロジェクトに限定した．また，それぞれの特徴をまとめたものを表 5.1 に示す．

- Apache NetBeans [9]

NetBeans は，Java をはじめとするいくつかのプログラミング言語に対応したオープンソースの IDE である．NetBeans の開発言語には主に Java が採用されている．

- Eclipse Platform [10]

Eclipse は，Java をはじめとするいくつかのプログラミング言語に対応したオープンソースの IDE である．Eclipse Platform の開発言語には主に Java が採用されている．

- mbed OS [11]

mbed は，同名のワンボードマイコン専用の IDE である．mbed OS の開発言語には主に C 言語が採用されている．

- Arduino Platform [12]

Arduino は，同名のワンボードマイコン専用の IDE である．Arduino Platform の開発言語には主に C 言語が採用されている．

表 5.1 対象プロジェクトの特徴

	コミット数	主な言語 (割合)	種別	期間	データ取得日
Apache NetBeans	1042	Java(94.5%)	汎用	2017/4/21-現在	2018/1/26
Eclipse Platform	8071	Java(98.8%)	汎用	2001/5/2-現在	2018/1/28
mbed OS	13823	C 言語 (94.1%)	組み込み	2013/2/18-現在	2018/1/28
Arduino Platform	6641	C 言語 (72.0%)	組み込み	2005/8/25-現在	2018/2/6

5.1.2 コミットコメントの抽出

4.1 節の手法を用いて、対象プロジェクトにおける全てのコミットに対応するコミット id、コミットコメント、コミット時刻を得た。

5.1.3 感情推定

4.2 節の手法を用いて、前項で抽出した全てのコミットコメントについて感情推定を行った。Python のパッケージである NLTK の `nltk.sentiment.vader` を用いて、4.2 節で示した `compound` の値を得た。

また 4.3 節の手法を用いて、コミットコメントから得られた感情値を各コミットにタグ付けした。

5.1.4 感情極性推移の可視化

前項でタグ付けした感情値を、4.4 節の手法を用いて可視化した。時刻を x 軸、感情値を y 軸にとり、折れ線グラフとして出力した。また y 軸の範囲は `compound` の値のとりうる範囲と同じく、 $-1 \leq y \leq 1$ である。

5.1.5 影響を及ぼすコミットコメントの抽出

4.5 節の手順に従い、分析などに際して意図しない影響を及ぼすコミットコメントを調査した。これにより、本実験で用いた 4 つのリポジトリにおいて、4.5 節に示した基準を満たすコミットコメントを特定した。

5.1.6 プロジェクトグループ間の差の調査

5.1.1 項に示した対象プロジェクトを, 4.6 節に示した手法で複数のグループに分類した.

まず, 4つの対象プロジェクトを, プロジェクトの成果物の種別によって分類を行うと, 以下のようになる.

- 汎用統合開発環境 : NetBeans, Eclipse
- 組み込み系統合開発環境 : mbed, Arduino

また, プロジェクトの主な開発言語によって分類を行うと, 以下のようになる.

- Java : NetBeans, Eclipse
- C 言語 : mbed, Arduino

よって, 成果物の種別による分類と主な開発言語による分類は同じ組み合わせとなる. 本論文では, 以下のように2つのグループを定義する.

- GrA : NetBeans, Eclipse
- GrB : mbed, Arduino

なお, 本実験では, 前項の結果で特定したコミットコメントを含むコミットのデータを除いたものを実験対象として扱う.

これら2グループにおける全てのコミットコメントから抽出した感情極性に関して, それぞれのプロジェクトの組み合わせにおいて差があるかを調査する.

さらに, GrA , GrB から感情値が `neutral` をとるコミットを除いたものについても実験対象とし, それぞれ GrA_{zero} , GrB_{zero} と置く.

検定にあたり, 統計解析向けのフリーソフトウェアである R[13] を用いる.

5.2 実験結果

5.2.1 影響を及ぼすコミットコメント

5.1.5 項で抽出された結果のうち, 各プロジェクトにおいて出現回数が多かったものを表 5.2 に示す. なお, 4.5 節に示した基準をどちらも満たす結果については網掛けで示す.

表 5.2 出現回数の多いコミットコメント

NetBeans		Eclipse		mbed		Arduino	
出現回数	感情値	出現回数	感情値	出現回数	感情値	出現回数	感情値
4	0	2345	-0.2023	95	0	59	0
3	0	67	0	46	0	46	0
-	-	47	0	25	0	34	0
-	-	29	0.3182	22	0	28	0
-	-	25	0	12	0	28	0

表 5.3 プロジェクト毎の分析結果

	平均	分散	中央値	標準偏差	最大値	最小値
NetBeans	-0.01083	0.02222	0	0.14905	0.7506	-0.7778
Eclipse	-0.06785	0.03536	0	0.18804	0.7845	-0.8591
Eclipse _{com}	-0.01279	0.03940	0	0.19850	0.7845	-0.8591
mbed	0.00937	0.03025	0	0.17394	0.8047	-0.8316
Arduino	-0.00062	0.03408	0	0.18462	0.8198	-0.8294

表 5.4 プロジェクト毎の分析結果 (neutral を除く)

	平均	分散	中央値	標準偏差	最大値	最小値
NetBeans	-0.06558	0.13163	-0.2263	0.36281	0.7506	-0.7778
Eclipse	-0.14384	0.06404	-0.2023	0.25305	0.7845	-0.8591
Eclipse _{com}	-0.05008	0.15254	-0.2057	0.39056	0.7845	-0.8591
mbed	0.04559	0.14564	0.2263	0.38163	0.8047	-0.8316
Arduino	-0.00266	0.14572	0.0516	0.38173	0.8198	-0.8294

5.2.2 感情極性推移

実験方法に示した手順に従って，対象プロジェクトのリポジトリからコミット情報を抽出し，さらにそれらのコミットコメントから感情極性を抽出した．

プロジェクト毎の感情極性について，表 5.3，表 5.4 に示す．表 5.3 は取得した全てのコミットのデータを使用したものであり，表 5.4 は感情値が neutral をとるコミットを除いたデータを使用したものである．また表 5.2 において網掛けで示されたコミットコメントを含むコミットを除いたデータを使用したものを Eclipse_{com} の項目で示す．

また感情極性推移について，取得した全てのデータを用いた結果のグラフを図 5.1，図 5.3，図 5.7，図 5.9 に示す．また，感情値が neutral をとるコミットを除いたデータを用いたグラフを図 5.2，図 5.5，図 5.8，図 5.10 に示し，前項の図 5.2 において網掛けで示したコミットコメントを含むコミットを除いたデータを用いたグラフ

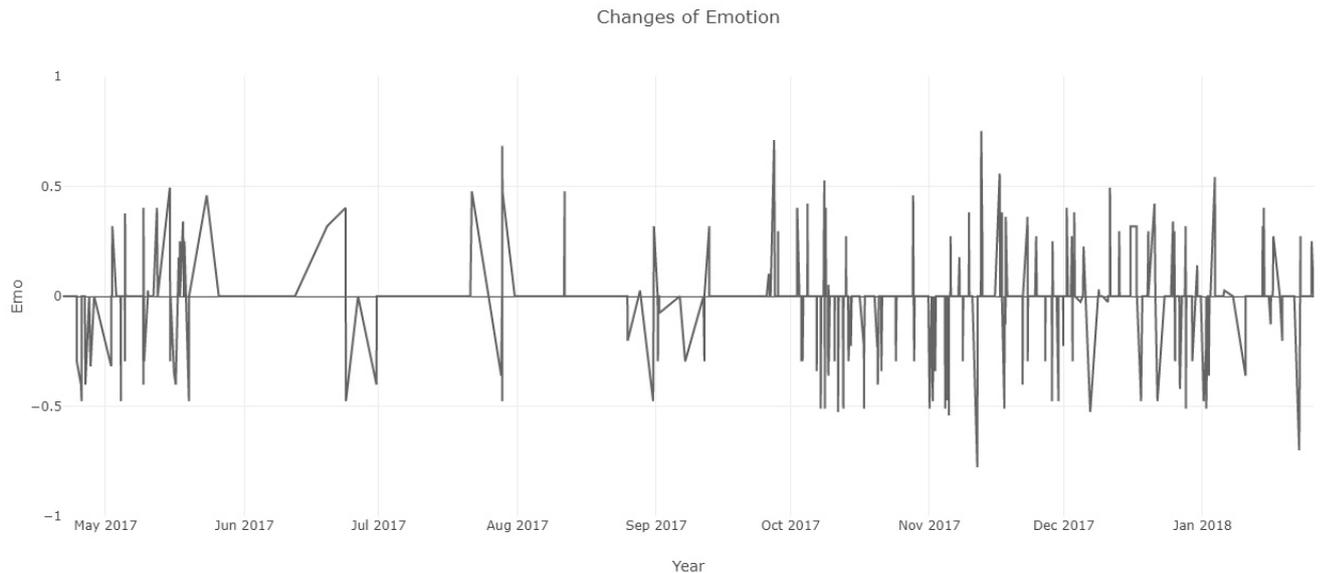


図 5.1 Apache NetBeans (全データ)

を図 5.4 に、さらにそこから感情値が neutral となるコミットを除いたデータを用いたグラフを図 5.6 に示す。

5.2.3 プロジェクト間の感情極性

実験方法に示した手順に従って、4つのプロジェクトを分類し、検定を行った。

なお本検定においては、図 5.2 において網掛けで示されたコミットコメントを含むコミットを除いたデータを用いるものとする。

本実験では、それぞれの標本は非正規分布に従うと仮定して Mann-Whitney の U 検定を行った。

GrA と GrB に所属するそれぞれのプロジェクトの感情値リストを検定対象とし、表 5.5 に U 検定を行った結果として出力された p 値を示す。

次に、 GrA_{zero} と GrB_{zero} に所属するそれぞれのプロジェクトの感情値リストを検定対象とし、表 5.6 に U 検定を行った結果として出力された p 値を示す。

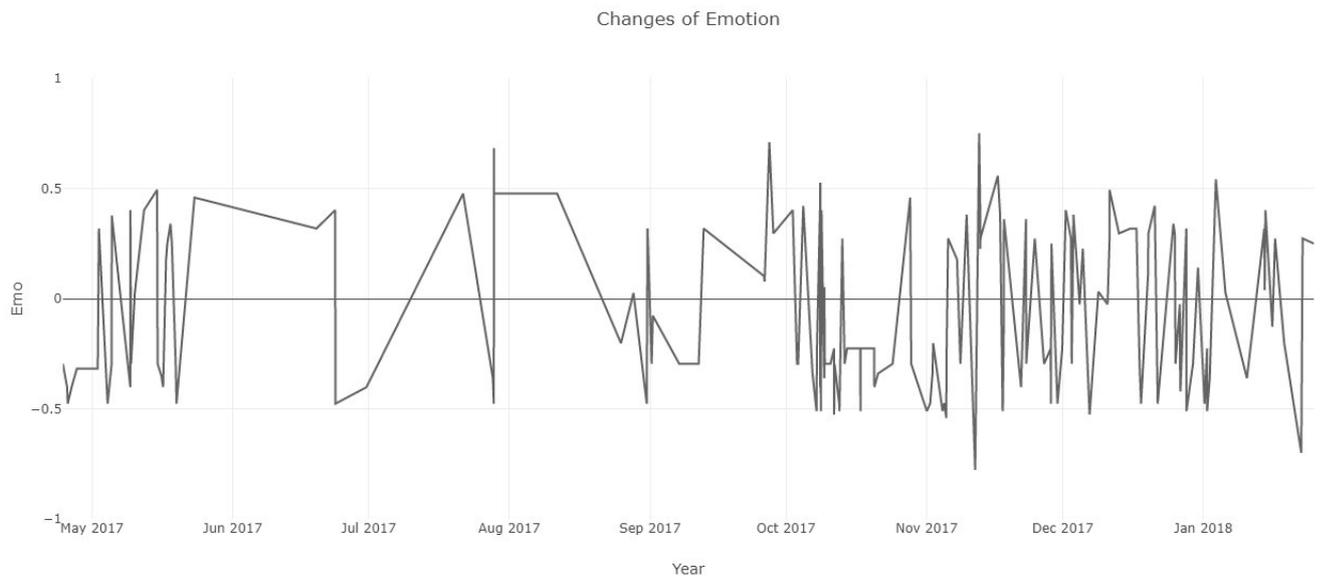


図 5.2 Apache NetBeans (neutral を除く)

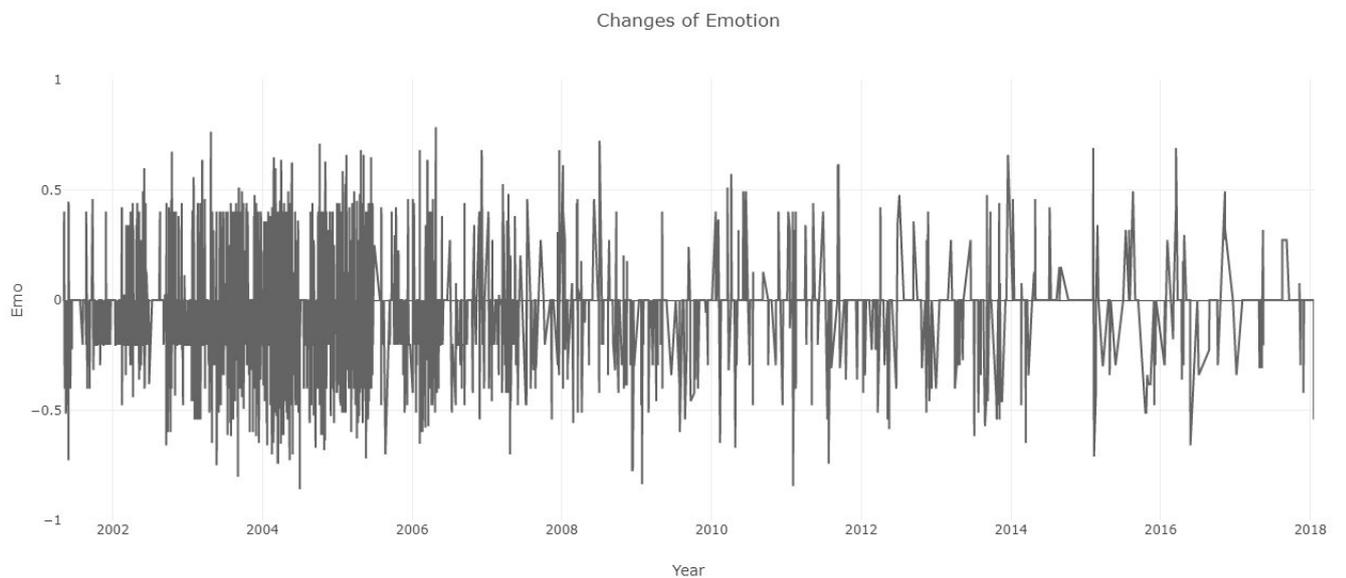


図 5.3 Eclipse Platform (全データ)

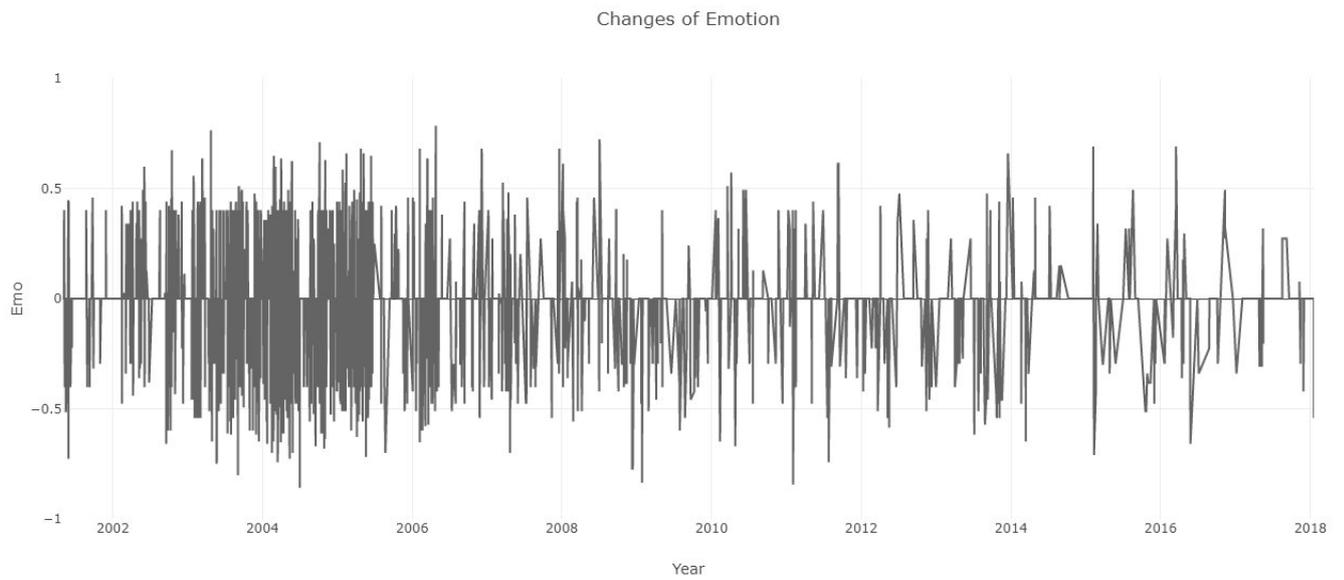


図 5.4 Eclipse Platform (一部コメントを含むデータを除く)

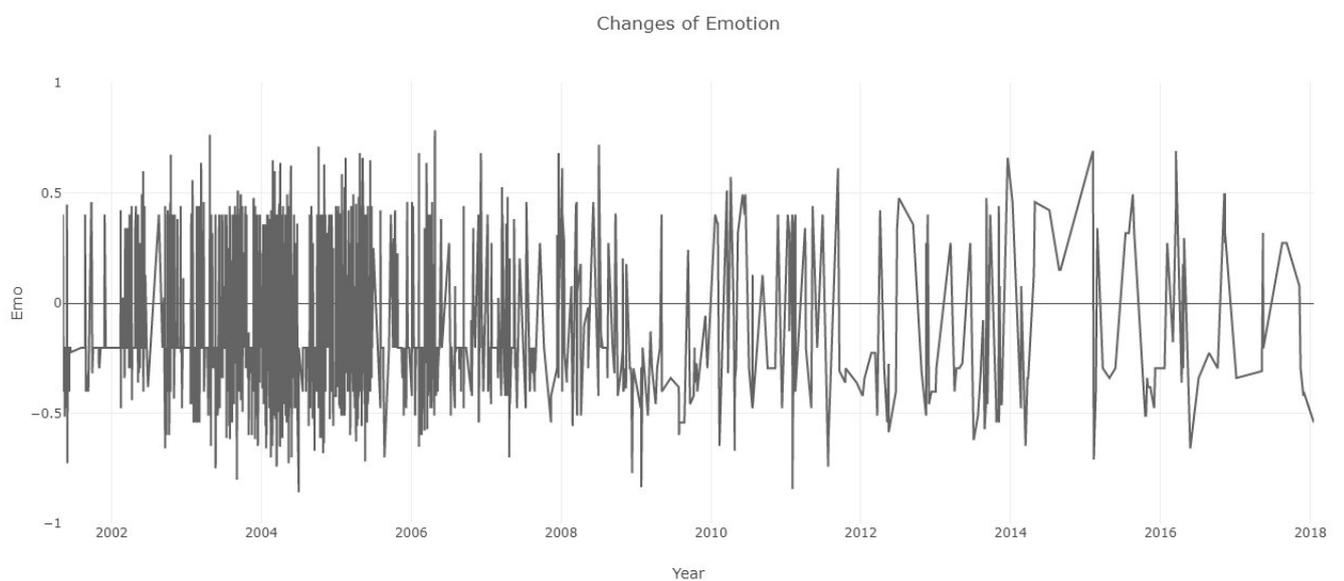


図 5.5 Eclipse Platform (neutralを除く)

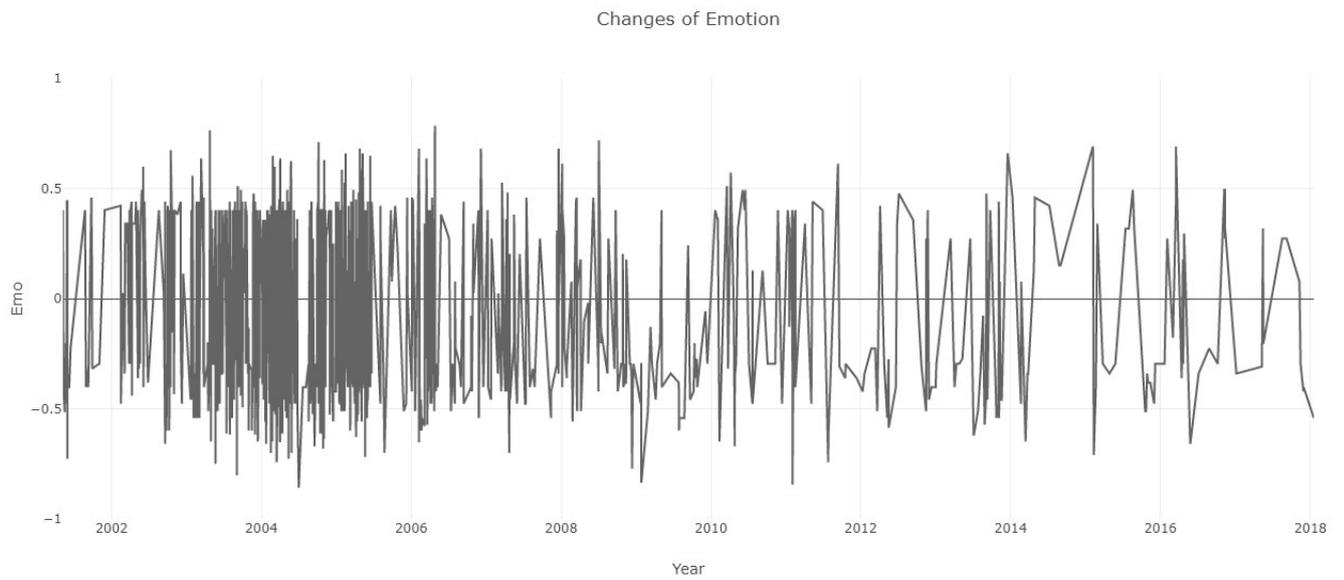


図 5.6 Eclipse Platform (neutral と一部コメントを含むデータを除く)

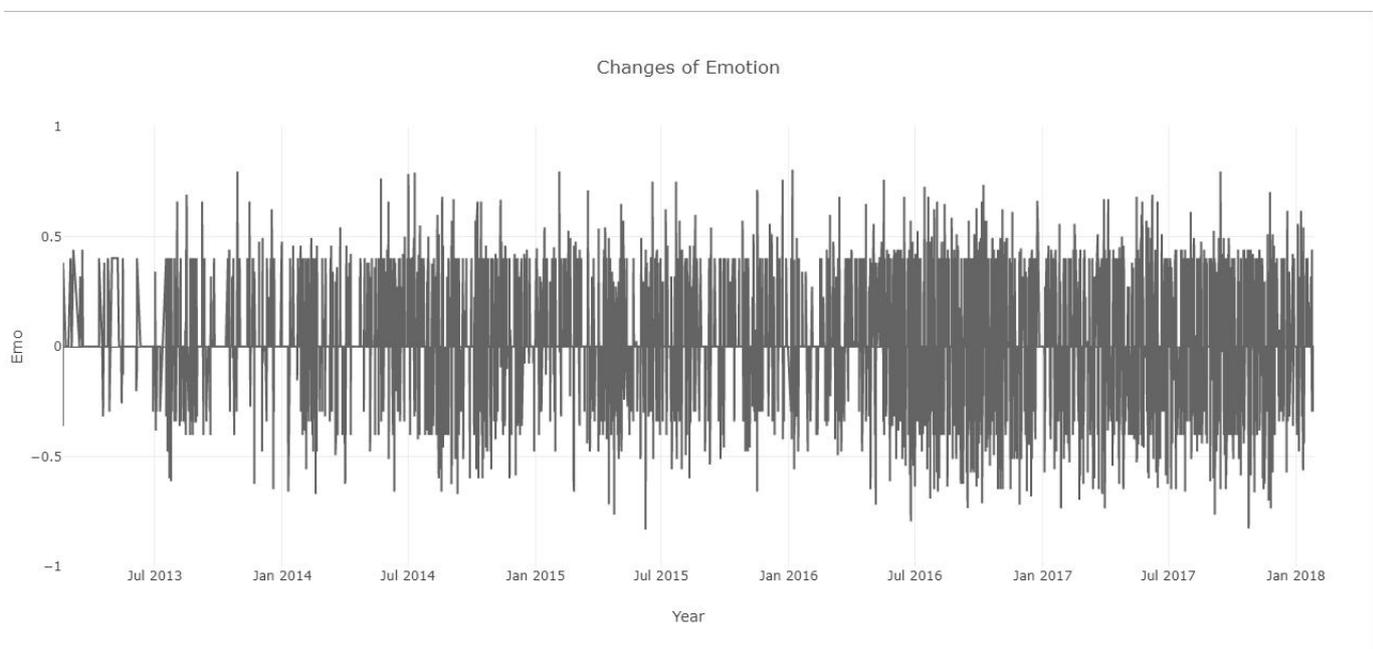


図 5.7 mbed OS (全データ)

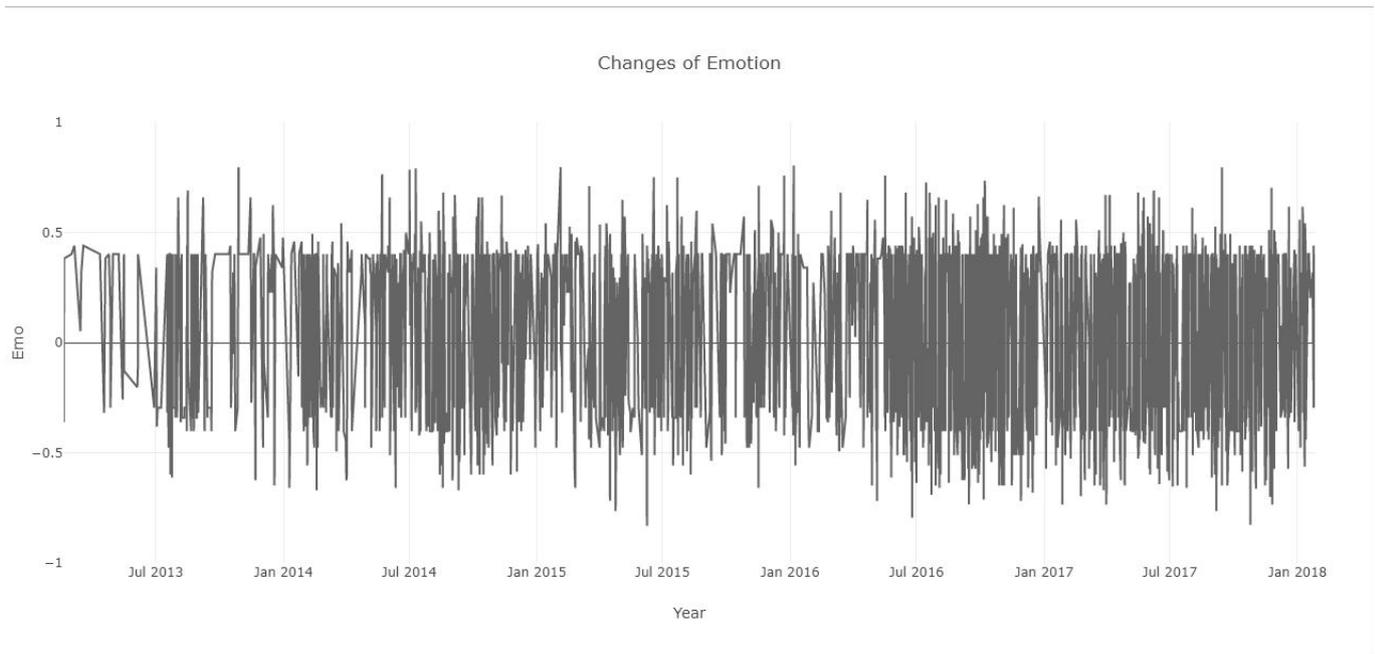


図 5.8 mbed OS (neutral を除く)

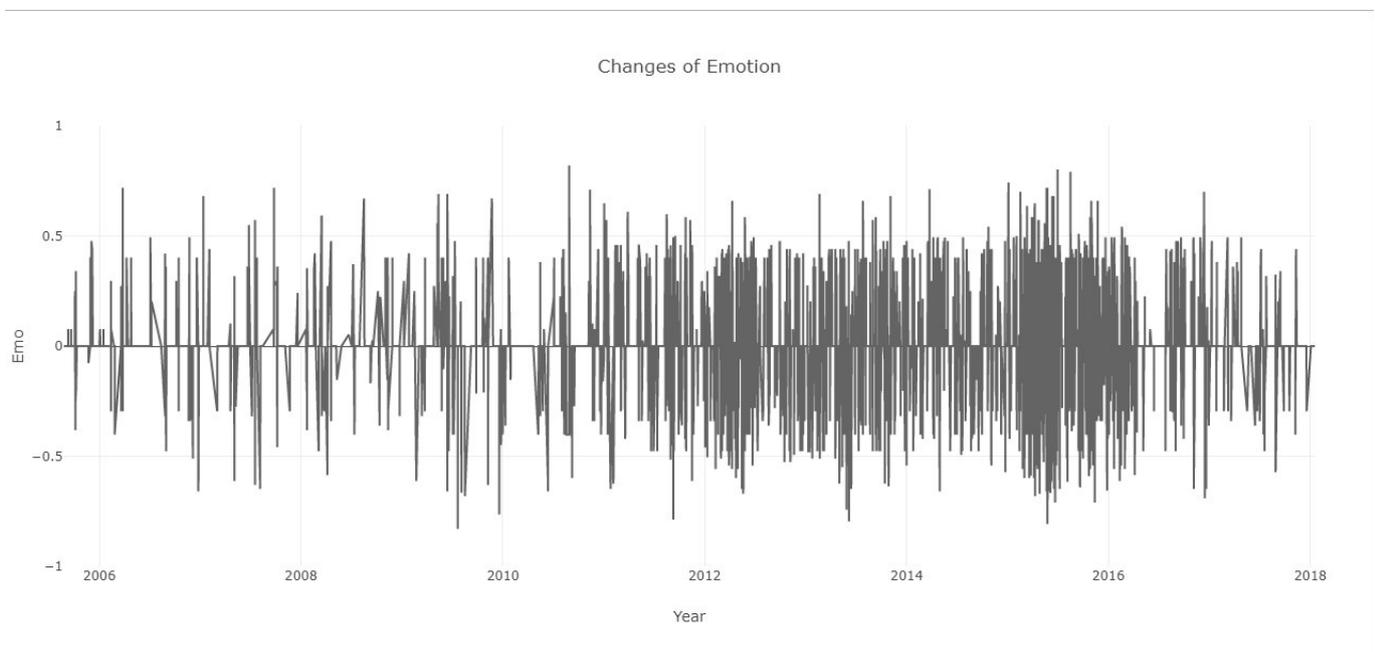


図 5.9 Arduino Platform (全データ)

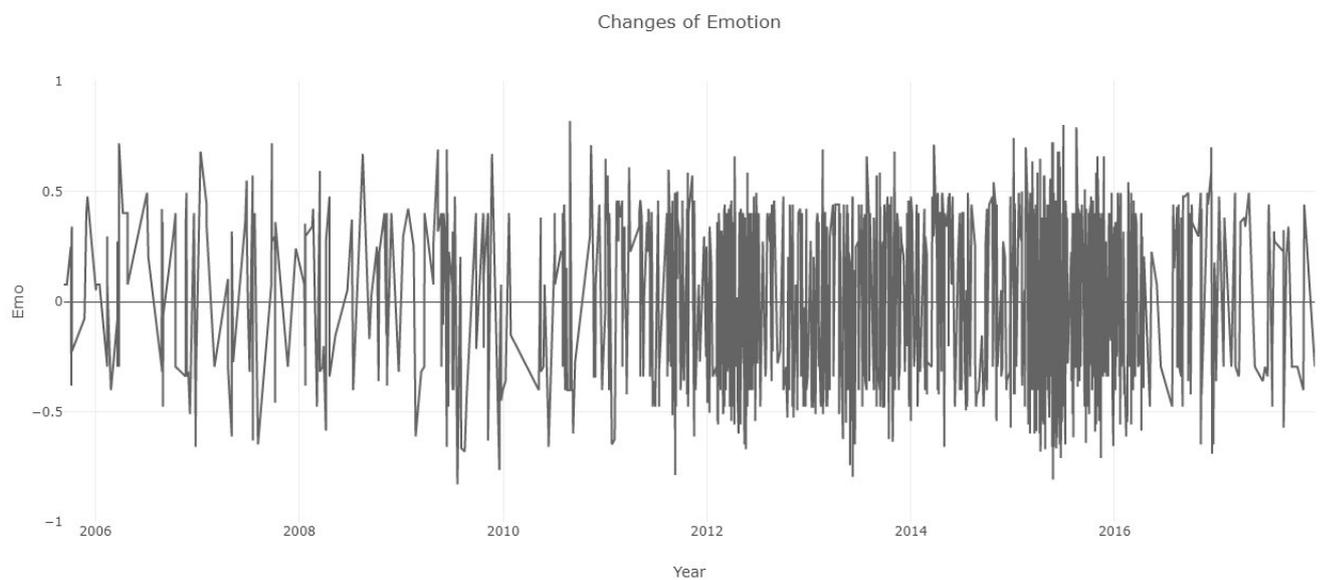


図 5.10 Arduino Platform (neutral を除く)

表 5.5 GrA と GrB に対する U 検定結果の p 値

$GrA \backslash GrB$	mbed	Arduino
NetBeans	3.42×10^{-4}	5.09×10^{-2}
Eclipse	4.15×10^{-12}	4.49×10^{-4}

表 5.6 GrA_{zero} と GrB_{zero} に対する U 検定結果の p 値

$GrA_{zero} \backslash GrB_{zero}$	mbed	Arduino
NetBeans	1.11×10^{-4}	3.36×10^{-2}
Eclipse	8.87×10^{-15}	1.45×10^{-4}

6. 考察

6.1 研究設問に対する考察

研究設問を通して，実験結果に対する考察を示す．

RQ1 コミットコメントから感情分析を行う際に不都合が生じるコメントが存在するか．

5.2.1 項で示したとおり，コミットコメントから感情推定や分析を行う際に何らかの影響を及ぼす可能性のあるコメントは，Eclipse Platform のプロジェクトにのみ存在し，以下の文字列であった．

```
*** empty log message ***
```

このコミットコメントは表 5.2 に示したように， -0.2023 の感情値をとる．これは，“empty”という単語が -0.2023 の感情値をとることが原因である．

また表 5.3，表 5.4 における Eclipse の項目と *Eclipse.com* の項目とを比較すると，該当のコミットコメントの有無によって，感情値の平均値に大きな差異が生じていることが分かる．

これらのことから，抽出されたコミットコメントは，コミットコメントから感情推定を行う際に意図しない影響を及ぼすコメントであると考えられる．

本論文内での紹介は省略するが，他の Eclipse の開発プロジェクトにおいてもこのコミットコメントと同じコミットコメントが多く見られた．また，このコミットコメントは Eclipse の開発プロジェクトに携わる複数の開発者によって使用されていることが分かった．よって，該当のコミットコメントは Eclipse の開発に際し，何らかの共通認識に基づいて用いられていたことがわかる．

また図 5.3 と図 5.4，図 5.5 と図 5.6 を比較することにより，このコミットコメントはプロジェクトの初期から用いられ，2008 年頃からはほぼ用いられていないことが分かる．今回対象とした Eclipse Platform においては，2011 年 1 月 3 日以後は一度も用いられていない．しかし表 5.2 より，Eclipse Platform の開発期間と同時期に開発が行われていた Arduino Platform の開発プロジェクトにおいては，同様に影響を与えるコミットコメントは見られなかった．これらのことから，該当のコミットコメン

トは時期によって一般的に開発者らに共通していた慣習とは言えず，Eclipse の開発プロジェクトに携わる開発者らの間でも，何らかの理由で徐々に用いられなくなったものと考えられる．

なお，バージョン管理システムにおいてコミットコメントを記述する本来の目的は，対応するコミットにおける変更内容の記録であり，これにより効率的にプロジェクトの開発を行うことが可能となる．今回抽出したコミットコメントの内容は，「空のログメッセージ」を意味しており，その目的に沿っていないと考えられる．

このことから，今回抽出したコミットコメントは，感情推定や分析に際して不都合が生じるだけでなく，バージョン管理システムにおいてコミットコメントを記述する本来の目的にも沿っておらず，使用を推奨されない内容のコミットコメントであると言える．

この結果より，感情推定や分析に際しては，不都合が生じるコミットコメントを含むコミットを除いたデータを調査対象として用いるのが妥当であると考えられる．

RQ2 特性の異なるプロジェクトにおいてコミットコメントの感情極性に差があるか．

GrA と GrB について，コミットコメントの感情極性に有意な差が存在するかどうかを調べるため，次のような仮説を設定した．

H : GrA と GrB の間には有意な差がない．

本検定における有意水準を 5% とすると，表 5.5 より，NetBeans と Arduino の関係について調査した検定結果が $p \geq 0.05$ であることが分かる．よって仮説 H は棄却されず， GrA と GrB に有意な差が存在するとは言えないことが分かる．

次に， GrA_{zero} と GrB_{zero} について，コミットコメントの感情極性に有意な差が存在するかどうかを調べるため，次のような仮説を設定した．

H_{zero} : GrA_{zero} と GrB_{zero} の間には有意な差がない．

本検定における有意水準を 5% とすると，表 5.6 より，全ての場合において $p < 0.05$ であることが分かる．よって仮説 H_{zero} は棄却され， GrA_{zero} と GrB_{zero} には有意な差が存在すると言えることが分かる．

これらのことから，プロジェクトの特性が異なる GrA と GrB において，以下のことが分かる．

- 感情値が neutral をとるコミットコメントを含む場合には有意な差が存在するとは言えない。
- 感情値が neutral をとるコミットコメントを含まない場合には有意な差が存在すると言える。

よって、感情値が neutral でないコミットコメントについて考慮した場合には、特性の異なるプロジェクト間の感情極性の推移に差が存在すると思われる。

6.2 妥当性の検証

次に、実験結果の妥当性について検証を行う。

6.2.1 オープンソースプロジェクトに適用した実験

今回の実験で用いたプロジェクトは、すべてオープンソースのものである。商用プロジェクトなど、オープンソースでないプロジェクトに今回の実験で用いた手法を適用した場合に、同様の結果が得られるとは限らない。

6.2.2 バージョン管理システムの機能

今回の実験で用いたバージョン管理システムには、過去のコミットの内容を削除、変更する機能が存在する。そのため、コミット時の感情を取得する意図で用いたコミットコメント等の情報が、コミット時のものではなく書き換えられたものである可能性がある。

6.2.3 感情極性推定手法

今回の実験で感情極性推定に用いたツールに実装された VADER の手法は、分類器の学習にソーシャルメディアのデータセットを用いたものである。そのため、VADER の手法で得られる感情値と、ソフトウェア開発分野における文脈での単語の感情値との間に、差異が存在する可能性がある。

6.3 今後の課題

以上の考察を踏まえ、今後の課題について考察する。

- 本研究では、5.1.1 節に示した 4 つのオープンソースプロジェクトを対象に実験を行った。より多くのプロジェクトを対象として実験を行うことによって、実験結果のさらなる一般化が期待できる。
- 本研究では、分類器の学習にソーシャルメディアのデータセットを利用した感情推定ツールを用いて実験を行った。より多くのデータセットを用いて分類器の学習を行う、あるいはソフトウェアリポジトリに記録されたコミットコメントそのものをデータセットとして分類器の学習を行うことにより、ソフトウェア開発分野における感情推定の精度の向上が期待できる。
- 本研究では開発者毎の感情の推移については考慮しなかった。プロジェクト内での開発者毎の感情の推移と、プロジェクト全体の開発者の感情の推移またはプロジェクトの特性との関係について調査を行うことによって、さらに多面から関係性を分析することができる。

7. 結言

本研究では、異なる特性を持つ4つのプロジェクトを対象として、バージョン管理システムのリポジトリから抽出したコミットコメントを文章とみなして感情推定を行い、開発者らの感情極性の推移とプロジェクトの特性の関係を調査した。4つのオープンソースソフトウェアプロジェクトに対して実験を適用し、以下の分析結果が得られた。

- コミットコメントから感情推定を行う際に、不都合が生じるコミットコメントが存在する。
- 感情値が neutral をとるコミットコメントを含むデータを含める場合、特性の異なるプロジェクトグループ間には有意な差が存在するとは言えない。
- 感情値が neutral をとるコミットコメントを含むデータを含めない場合、特性の異なるプロジェクトグループ間には有意な差が存在する。

これらの結果から、コミットコメントから抽出した感情極性の推移とプロジェクトの特性には関係が一部存在し、プロジェクトの特性に応じてコミットコメントに対する感情推定結果を分類することが有効であると考えられる。

今後の課題としては次のようなことが挙げられる。

- より多くのプロジェクトを対象に分析を行う。
- より多くの分類方法によってグループを設定し分析を行う。
- ソフトウェア開発に用いられる語彙をカバーした感情推定の方法を利用する。
- 開発者毎の感情極性推移とプロジェクト全体の感情極性推移を比較する。

謝辞

本研究を行うにあたり、研究課題の設定や研究に対する姿勢、本報告書の作成に至るまで、全ての面で丁寧なご指導を頂きました。本学情報工学部門 水野修教授に厚く御礼申し上げます。本報告書執筆にあたり貴重な助言を多数頂きました。本学情報工学専攻 田中健太郎君、黒田翔太君、西浦生成君、原田禎之君、近藤将成君、小林勇揮君、洪浚通君、情報工学課程 植村佳治君、北村紗也加さん、中村勝一君、

廣瀬早都希さん，渡辺大輝君をはじめとするソフトウェア工学研究室の皆さん，学生生活を通じて著者の支えとなった家族や友人に深く感謝致します．

参考文献

- [1] A. Murgia, P. Tourani, B. Adams, and M. Ortu, “Do developers feel emotions? an exploratory analysis of emotions in software artifacts,” Proc. 11th Working Conf. on Mining Software Repositories, pp.262–271, Hyderabad, India, May 2014.
- [2] M.R. Wrobel, “Emotions in the software development process,” Proc. 6th International Conf. on Human System Interaction, pp.518–523, June 2013.
- [3] GitHub, Inc., GitHub, (オンライン), 入手先 <https://github.com> (参照 2018-01-24).
- [4] E. Guzman, D. Azocar, and Y. Li, “Sentiment analysis of commit comments in github: an empirical study,” Proc. 11th Working Conf. on Mining Software Repositories, pp.352–355, Hyderabad, India, May 2014.
- [5] GitPython, (オンライン), 入手先 <https://gitpython.readthedocs.io/en/stable/> (参照 2018-01-24).
- [6] NLTK, NLTK: Natural Language Tool Kit, (オンライン), 入手先 <http://www.nltk.org/> (参照 2018-01-24).
- [7] C.J. Hutto and E. Gilbert, “Vader: A parsimonious rule-based model for sentiment analysis of social media text,” Conference: Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media, At Ann Arbor, MI, May 2014.
- [8] Plotly, Plotly, (オンライン), 入手先 <https://plot.ly/> (参照 2018-01-24).
- [9] Apache, Apache NetBeans, (オンライン), 入手先 <https://netbeans.org/> (参照 2018-01-24).
- [10] Eclipse, Eclipse Platform, (オンライン), 入手先 <http://wiki.eclipse.org/Platform> (参照 2018-01-24).
- [11] Arm, mbed OS, (オンライン), 入手先 <https://www.mbed.com/en/> (参照 2018-01-24).
- [12] Arduino, Arduino Platform, (オンライン), 入手先 <https://www.arduino.cc/> (参照 2018-01-24).

[13] R, (オンライン), 入手先 <https://www.r-project.org/> (参照 2018-02-06).