

2017年度 博士論文

インターネットの情報発信を用いた
信頼指標に関する研究



KEIO MEDIA DESIGN

慶應義塾大学大学院メディアデザイン研究科

藤川 真一

本論文は慶應義塾大学大学院メディアデザイン研究科に
博士(メディアデザイン学) 授与の要件として提出した博士論文である。

藤川 真一

指導教員：

砂原 秀樹 教授	(主指導教員)
加藤 朗 教授	(副指導教員)
古川 享 教授	(副指導教員)
岸 博幸 教授	(副指導教員)

審査委員：

加藤 朗 教授	(主査)
古川 享 教授	(副査)
岸 博幸 教授	(副査)
折田 明子 准教授	(副査, 関東学院大学)

インターネットの情報発信を用いた 信頼指標に関する研究

内容梗概

日々、インターネットを通じて見知らぬ人同士が情報のやりとりをする機会が増えているが、相互に情報交換を行い、実際に出会って見たら期待通りの相手ではなかったり、金銭の伴うやり取りを行ったら、相手に騙されてしまい商品が届かないことなどが起きている。インターネットを通じたコミュニケーションでは、情報発信者が望む情報がやり取りされる。そこには情報の非対称性が存在するため、短い言葉のやりとりのみで相手のことを信用するのは難しい。このような状況で正しく相手の信用判断を行うきっかけがあれば、試しに相手を信用してみようかとチャレンジしやすい状況を作ることができる。それに対して望ましい結果を得られれば相手に対して信用が生まれる。

本研究では、インターネットにおける情報の非対称性を前提とした不確実な状況下で、ユーザ行動や発信している情報からその情報発信者を信頼する手がかりになる情報を指標を生成するモデルの提案を行う。スコアリングモデルとして情報の発信頻度や情報に対する評価から、信頼に対する期待のされやすさを示すトラストスコアを算出する。信用とは相手に信頼への期待をし、期待通りの結果を得られた時に抱くことができるものである。しかし、信用は人の心の中に生まれるもので計測することは不可能である。そのため、コミュニケーションにチャレンジするためのきっかけとなるための指標としてトラストスコアを用いる。トラストスコアは相手の能力を判断する際の初期スクリーニングとして、能力に対する信頼への期待のされやすさを示す。

また、トラストスコアが示す能力には適用範囲が存在する。トラストという言葉には、その人を全て信用するなどといった意味を期待することがあるが、情報の非対称性を前提として算出されているトラストスコアではそのような判断は不可能である。そのためトラストスコアの算出対象となるウェブサービスが提供するサービスの文脈と評価機能に着目する。つまり、ウェブサービスの特定の機能を通じて活躍し、周囲から高い評価を得ているユーザは、そのウェブサービスの設計意図に対して望ましい行動をし、かつ特定の能力を有するユーザであると考えられる。

この考え方よりウェブサービスの行動特性から分析可能な能力を評価カテゴリとして定義し、行動履歴からスコアリングを行う。もし対象となるウェブサービスの社会的評価が高いのであれば、トラストスコアが高いユーザも、該当の評価カテゴリについては高い評価を得られる期待を抱くことができる。仮に、その評価カテゴリの能力を有する人物を採用したい企業があるならば、トラストスコアを人材のスキル評価における初期スクリーニングに活用することが期待できる。

トラストスコアの適用例としてソースコードの共有サービスである Github を活用する開発者スキルの解析を行った。Github において高いトラストスコアを持っている開発者が、Web サービス企業にとって採用したいと思う開発者であると判断できるなら、人材採用の判断における初期スクリーニングの機会を本スコアを通じて提供することができる。算出したトラストスコアについて、Web サービスをビジネスとして行っている業界の人材採用や給与相場に影響を与えている CTO (最高技術責任者) クラスの管理職兼エンジニアにアンケートを行い、トラストスコアの有用性の検証を行った。その結果、高いトラストスコアの開発者であれば、積極的に会いたいと判断することがわかった。トラストスコアを活用すると、CTO クラスの人材がその都度判断する時間を割かなくても、一定以上のトラストスコアを持つ人材であれば、人事担当者だけで初期スクリーニングを行いスピーディな人材採用活動を進められるようになる。このようにインターネット上に発信されている情報と評価情報を元にトラストスコアを算出することで、特にスコアの上位については情報の不確実性を減らす効果が認められ、能力への信頼に対する期待を持ちやすくなることが判明した。トラストスコアによって、インターネット上の意思決定をしやすくし、その結果として精度の高いコミュニケーションを提供し、インターネットをよりスムーズに活用できるようにする。

キーワード

インターネット, コミュニケーション, 信頼, インターネットビジネス

慶應義塾大学大学院メディアデザイン研究科

藤川 真一

A Trust Score based on Activities on the Internet

Abstract

The opportunity for strangers to communicate with each other via the Internet is increasing day by day. They exchanged information with each other and they met directly, but they were not as expected, they paid money, but products might not be sent.

In communication via the Internet, the information sender desires to exchange information. Because of the asymmetry of information, it is difficult to trust each other in short word interaction. If there is a chance to judge the trust, you can create a challenging situation that trusts you. This is very important.

In this research, in an uncertain situation premised on the asymmetry of information in the Internet, we propose a model that generates indicators based on user behavior and information being sent, which is a clue to trust the information sender. As a scoring model, a trust score indicating the likelihood of expectation for trust is calculated from the frequency of information transmission and evaluation on information. The trust is created when you expect expectations of trust and you get the results you expect. However, it is impossible to measure it because it is created in the human mind. Therefore, we use trust score as an index to challenge communication. Trust score can be used for initial screening when judging information.

There is a scope of applicability in the expectation of expectation for trust indicated by the trust score. In the word trust sometimes I expect the meaning to say that he trusts everyone. However, such a judgment is impossible with the trust score which is calculated on the assumption of asymmetry of information, so the category which is the scope of application is specified. In order to define this category, we focus on the function and evaluation value of a specific web service. It can be thought that a user with a high evaluation value for a specific action of the web service is a result of doing desirable behavior against the design intention of the web service. By thinking like that, we define capabilities that can be analyzed from the behavioral characteristics of web services as categories and

calculate them as scores. If the target web service is socially recognized, for users with high trust scores, companies that wish to hire people with the capabilities of the corresponding category can be utilized for initial screening of ability judgment

As an application example of the trust score, we analyzed the developer skill using Github which is a sharing service of the source code. If you can determine that a developer with a high trust score in Github is a developer you want to adopt for a Web service company, you can offer initial screening opportunities in the recruitment decision through this score. For the calculated trust score, he interviewed engineers of CTO classes engaged in recruitment of talent at Web service company, and verified its usefulness. As a result, even if it is not an engineer of the CTO class, we found that personnel managers can advance speedy recruiting activities by utilizing the trust score for initial screening. By calculating the trust score based on the information and evaluation information sending on the Internet in this way and by linking the ease of expectation to trust for the information being transmitted to real actions, It can be used more. Also, when making decisions on the Internet, we also provide opportunities to take the first step towards trust judgment.

Keywords:

Internet Communication , trust , Online business

Keio University Graduate School of Media Design

Shinichi Fujikawa

目 次

第1章	はじめに	1
1.1.	本研究の背景	1
1.2.	本研究の目的	2
1.3.	本論文の構成	4
第2章	インターネットで求められる信頼	6
2.1.	信頼の単位としてのインターネット上のアイデンティティ	6
2.2.	インターネット上のアイデンティティの特徴	9
2.3.	アイデンティティ管理技術におけるデジタルアイデンティティとの 関連性	11
2.4.	コミュニケーションに必要とされる信用	13
2.5.	インターネットのコミュニケーションにおける信用	14
第3章	信頼への期待のされやすさ	18
3.1.	情報発信の継続性	18
3.2.	情報発信に対する反応	19
3.3.	情報発信の継続と反応の組み合わせ	21
3.4.	炎上後に情報発信を継続することの重要性	22
3.5.	Web上で可視化されている評価情報	27
3.6.	Webサービスが指し示す能力としての評価カテゴリ	29
3.7.	アイデンティティの本人の特定可能性を公表することと信頼の関係性	31
3.7.1	匿名のアイデンティティ	31
3.7.2	Webサービスが本人の特定を確認しているケース	32
3.7.3	アイデンティティの継続による本人の識別性の確保	35
3.7.4	本人の特定可能性を表明するアイデンティティ	36
3.7.5	アカウントの特定可能性を表明していることと、信頼関係構 築の関係性	36

3.8.	アイデンティティの特定可能性の表明と得られる信頼についてのまとめ	38
第4章	トラストスコアの提案	39
4.1.	トラストスコア	39
4.1.1	指標としてのトラストスコア	39
4.1.2	指標化を行う必要性	40
4.2.	トラストスコアのスコアリングモデルの検討	41
4.2.1	アメリカの消費者金融における信用判断の検討	41
4.2.2	クレジットスコア	42
4.2.3	クレジットスコアによる信用情報の構造	44
4.2.4	クレジットスコアを参考にしたトラストスコアモデルの検討	44
4.2.5	Klout Scoreによるインターネットユーザの影響力の指標化モデルの検討	45
4.2.6	Klout Scoreの指標化モデル	46
4.2.7	Klout Scoreのまとめ	47
4.3.	トラストスコア的设计	48
4.3.1	トラストスコアの構成要素	48
4.3.2	トラストスコアの算出モデル	49
4.3.3	トラストスコアの算出モデルの構成要素	49
4.3.4	トラストスコアの決定	52
第5章	Githubの公開情報を用いたトラストスコア的设计	54
5.1.	Github	54
5.2.	Githubを用いる理由	55
5.3.	Githubの公開情報におけるエンジニアの信頼への期待のされやすさ	57
5.4.	Githubで計測可能なユーザアクティビティ	58
5.4.1	ユーザ (Users)	60
5.4.2	リポジトリ (Repositories)	60
5.4.3	コミット (Commits)	60
5.4.4	フォーク (Forks)	60
5.4.5	プルリクエスト (Pull Requests)	61
5.4.6	Issues	61

5.4.7	Watch	61
5.4.8	Starring	62
5.5.	Github におけるトラストスコアの設計	63
5.5.1	Github におけるアイデンティティの単位	63
5.5.2	評価カテゴリの設計	63
5.5.3	Github から抽出できるスコアリングの設計	67
5.5.4	トラストスコアの算出式	69
第 6 章	Github からトラストスコアの算出	72
6.1.	Github のデータの取得	72
6.2.	GHTorrent から取得できるデータ	73
6.2.1	GHTorrent のデータベース	73
6.2.2	日本人ユーザの特定	73
6.3.	コミット履歴を活用した情報発信量スコアの基礎評価	77
6.4.	月次コミット総数に関するスコアの算出	78
6.4.1	コミット数による長期的な情報発信量スコアの算出	78
6.4.2	コミット数による短期的な情報発信量スコアの算出	78
6.5.	Star 登録数を活用した評価スコアの基礎評価	79
6.6.	月次獲得 Star 獲得数によるスコアリング	79
6.6.1	Star 獲得数による長期的なりポジトリへの評価スコアの算出	80
6.6.2	Star 獲得数による短期的なりポジトリへの評価スコアの算出	81
6.7.	Github におけるトラストスコア	82
6.7.1	日本のアイデンティティにおけるトラストスコア	82
6.7.2	その他の国におけるトラストスコア	85
6.8.	スコアリングの考察	87
第 7 章	トラストスコアの評価	89
7.1.	トラストスコアの評価	89
7.2.	トラストスコアのヒアリングによる予備実験	90
7.2.1	インタビュー概要	90
7.2.2	インタビュー方法	90
7.2.3	対象者の特性	91
7.3.	インタビュー結果のまとめ	92

7.3.1	肯定的な感想	92
7.3.2	否定的な感想	93
7.3.3	その他の感想	95
7.3.4	各自の感想	95
7.3.5	予備実験のインタビューの考察	96
7.4.	トラストスコアに関するアンケート調査	97
7.4.1	アンケート調査概要	97
7.4.2	調査票の設計と分析方法	98
7.4.3	ペルソナの特徴	99
7.4.4	調査結果と仮説についての検討	100
7.4.5	アンケートに関するインタビューのまとめ	107
7.4.6	アンケートおよびインタビューの考察	111
第8章	全体考察	114
8.1.	トラストスコア設計に関する考察	114
8.2.	トラストスコアの貢献と限界	118
8.3.	Githubのトラストスコアの活用に関する考察	119
8.4.	Githubのトラストスコアの活用範囲	120
8.5.	今後の展望	121
第9章	結論	124
付録A	トラストスコアのアンケート	133
A.1.	アンケート表紙	133
A.2.	基本質問	134
A.3.	共通の選択肢	134
A.4.	職務経歴書アンケート1（最初の20人）	135
A.5.	職務経歴書アンケート2（残りの20人）	144
A.6.	回答集計	155

目 次

2.1	Web サービスのアカウント認証による同一人物の識別ロジック . . .	6
2.2	情報発信者とアイデンティティの関係性	8
2.3	サブアカウントとアイデンティティの関係性	10
2.4	エンティティとアイデンティティ情報	12
2.5	駅の待ち合わせのコミュニケーション成立に必要な相互の信用 . . .	13
2.6	信頼の分類	15
2.7	信頼への期待と結果としての信用	16
3.1	情報発信の継続性から得られる信頼への期待	19
3.2	否定的な反応が連鎖する情報のデリバリ	24
3.3	炎上後のアイデンティティ継続	26
3.4	評価を意識した情報発信	28
3.5	Intergram の画面上のフォロワ数表示	29
3.6	Twitter の認証マーク	33
4.1	クレジットスコアの算出フロー	44
4.2	Klout の Web 画面	46
4.3	Klout Score の指標化モデル	47
4.4	トラストスコアの算出モデル	50
5.1	github の Web 画面	55
5.2	github の Web 画面	56
5.3	Github API のオブジェクト構造	59
5.4	開発系技術者のキャリア分類	64
5.5	開発系技術者のキャリア特性	65
5.6	開発専門職におけるトラストスコア向上のサイクル	68
6.1	Ghtorrent の Web 画面	72

6.2	GHTorrent のデータベーススキーマの概略図	74
6.3	5カ国のトラストスコアの分布	87
7.1	採用に携わっているかについての分布	101
7.2	役職の分布	102
7.3	経験年数の分布	102
7.4	後半 20 人のトラストスコアと面接指標平均の相関図	106
7.5	前半 20 人のトラストスコアと面接指標平均の相関図	107
A.1	1 さんの評価結果	155
A.2	2 さんの評価結果	155
A.3	3 さんの評価結果	155
A.4	4 さんの評価結果	156
A.5	5 さんの評価結果	156
A.6	6 さんの評価結果	156
A.7	7 さんの評価結果	157
A.8	8 さんの評価結果	157
A.9	9 さんの評価結果	157
A.10	10 さんの評価結果	158
A.11	11 さんの評価結果	158
A.12	12 さんの評価結果	158
A.13	13 さんの評価結果	159
A.14	14 さんの評価結果	159
A.15	15 さんの評価結果	159
A.16	16 さんの評価結果	160
A.17	17 さんの評価結果	160
A.18	18 さんの評価結果	160
A.19	19 さんの評価結果	161
A.20	20 さんの評価結果	161
A.21	21 さんの評価結果	161
A.22	22 さんの評価結果	162
A.23	23 さんの評価結果	162
A.24	24 さんの評価結果	162

A.25 25 さんの評価結果	163
A.26 26 さんの評価結果	163
A.27 27 さんの評価結果	163
A.28 28 さんの評価結果	164
A.29 29 さんの評価結果	164
A.30 30 さんの評価結果	164
A.31 31 さんの評価結果	165
A.32 32 さんの評価結果	165
A.33 33 さんの評価結果	165
A.34 34 さんの評価結果	166
A.35 35 さんの評価結果	166
A.36 36 さんの評価結果	166
A.37 37 さんの評価結果	167
A.38 38 さんの評価結果	167
A.39 39 さんの評価結果	167
A.40 40 さんの評価結果	168

目 次

3.1	情報の受け手の感情分類	20
3.2	炎上時における閲覧者の反応	23
3.3	可視化されている評価情報	27
3.4	Web サービスが指し示す能力の評価カテゴリ	30
3.5	Web サービスによる本人情報の特定の目的	34
4.1	信用情報機関が収集する個人信用情報	42
4.2	クレジットスコアとトラストスコアの比較	45
4.3	トラストスコアの基準	52
5.1	Github から測定できる信頼への期待感	58
5.2	ユーザのタイプとトラストスコア	70
6.1	GHTorrent が提供するテーブル一覧	75
6.2	国別ユーザ数 上位 10 カ国	76
6.3	JP ユーザのコミット数別分類	77
6.4	JP ユーザの Star 数分布	80
6.5	日本における Github のトラストスコア	82
6.6	トラストスコア 5 のアイデンティティが使用している開発言語	83
6.7	トラストスコア 4 のアイデンティティが使用している開発言語	83
6.8	トラストスコア 3 のアイデンティティが使用している開発言語	84
6.9	トラストスコア 2 のアイデンティティが使用している開発言語	84
6.10	トラストスコア 1 のアイデンティティが使用している開発言語	85
6.11	アメリカにおける Github のトラストスコア	86
6.12	フランスにおける Github のトラストスコア	86
6.13	中国における Github のトラストスコア	86
6.14	ベトナムにおける Github のトラストスコア	87

7.1	インタビュー概要	90
7.2	トラストスコア検証の仮説	97
7.3	アンケート概要	98
7.4	ペルソナ前半20人	99
7.5	ペルソナ後半20人	100
7.6	面接指標	103
7.7	ペルソナ前半20人	104
7.8	ペルソナ後半20人	105
7.9	インタビュー内容	108
7.10	Q.書類審査する場合に、どういう情報に着目して判断したか? . . .	108
7.11	Q.トラストスコアがあることでのイメージに影響したか?	109
7.12	Q.トラストスコアが高い人は採用したくなりましたか?	109
7.13	Q.トラストスコアがあると、仮にネガティブな情報があっても、会っ てみようかと思うか?	109
7.14	Q.トラストスコアが低い人(1, 2)の人はどう思いましたか?	110
7.15	Q.海外の人のトラストスコアの有無は判断に影響しましたか? . . .	110
7.16	Q.その他、トラストスコアに感じたメリットなどや感想など	111
7.17	外国人の評価比較	113

第1章

はじめに

1.1. 本研究の背景

インターネット上の Web ページやコミュニティサイト、情報サイト等へ人間から発信されている情報は、発信者が発信したいと思った情報が発信されている。美味しいと思った感想をグルメサイトや SNS に発信することもある。店員の態度が悪いという不満のはけ口として否定的な情報を発信することもある。また、このお店は友達のお店だから有名にしたいと思って、美味しい美味しくないに関わらず意図的に紹介することもある。インターネット上に存在するあらゆる情報は、発信者が有利になるためのポジショントークとして発信されていると判断することも不可能ではない [1]。しかし、インターネット上で見知らぬ人とコミュニケーションをする場合、交わされた話の真偽が全く判断できないのであれば、その情報を信じた結果が望ましい結果になるのは偶然でしかなく、相手を信じてはいけないことになる。しかし、インターネットを通じて情報を交換したり、商品を取引する機会は増えており、多くの方は相手を完全には信じず疑っていたとしても、取引をしなくなっているわけではない [2]。インターネットを介して見知らぬ人とコミュニケーションを行ったり金品の取引を行ったりする際などに、相手が約束を守ってくれるか否かについて判別する方法があれば、多くの人にとってインターネットを介したコミュニケーションをより適切に活用できるようになると期待できる。

インターネット上の情報の真偽の判断に慣れている利用者の中には、対象となる発信者の情報を複数の視点から検索したり、記載されている情報を深掘りして調査を行うことで、信頼できそうかどうかを調べている人がいる。例えばネットショッピングを行う際に、ショップに記載が義務付けられている特定商取引法に基づく記載欄にある住所や会社名を検索したり、Google ストリートビューを活用して実際の店舗の写真を確認したり、決算情報や店頭公開している企業の情報など

を元に信頼性を判断し、この取引相手は悪いことはしないだろうと判断して購入するなどである。しかし、いくら調べても、その会社が明日不渡りを出して倒産するかもしれないことや、実は明確な悪意があって、クレジットカード番号を盗もうとしていることや、商品を送らずにお金だけだまし取ろうと思っていることまではインターネットの情報からは判断できない。確実な判断はできないまでも、何かの情報を元に相手を信じることに対する期待を抱けるからこそ、非対面にも関わらずインターネットを通じて取引を行うのだと考えるのが妥当である。

更に今日のインターネットでは、個人間の取引や個人事業主による取引も増えている。相手が個人の場合は、その人の情報をインターネットの情報から調べるのは難しい状況にある。そのような状況下で発信者を信頼できそうだという指標を提示することで、信頼に対する期待のされやすさを示せることは非常に有意義である。

本研究においては、そのようなインターネット上の情報発信という限られた情報の中から、相手を信じる第一歩となる「発信された情報が望ましい結果になることへの期待」を抱く判断に必要な指標化モデルを研究する。本研究の指標を用いてインターネットに発信されている情報が信頼できそうだことに対して期待を持ち、実際に行動を行った結果が期待通りであれば、そこに信用が生まれる。このような行動の積み重ねによって、インターネットに発信されている情報の信頼性向上に寄与することが期待できる。

1.2. 本研究の目的

本研究ではインターネット上に情報を発信している見知らぬ人に対する信用判断に活用できる指標を生成し、提供する。インターネットに発信されている情報は不完全であったり、発信者の意図に基づいた発言のみが発信されていることが多く、情報そのものを解析したとしても必ずしも信用できるとは限らないため、発信者の情報のみで発信者を信用するのは難しい。

本研究では、このような問題を解決するために、インターネットを通じて発信されている情報を元に、その情報発信者を信頼する手がかりとなる指標を生成するモデルの提案を行う。さまざまな Web サービスが持つ特徴から判断可能な特定の能力に対し、情報発信の頻度や量と、他のユーザからの評価情報を元にスコアリングを行う。

つまり、特定の Web サービス上で活躍している情報発信者は、その Web サービスで活躍するために必要な能力を有していると考えることで、Web サービス上の情報発信機能や評価機能から能力のスコアリングすることができるようになる。この Web サービスが示す能力を、その人に対して信頼可能な評価カテゴリと定義し、信頼可能な範囲を絞り込むことで、特定の能力を前提としたコミュニケーションを行うための判断軸を提供する。

このようなスコアを活用し、情報発信者が保有する能力や特性を評価できるようになることで、評価カテゴリに対応するスキルを持っているかどうかを判断できる能力を有しない人でも、情報発信者が特定の能力を有することに期待を持って接することができるようになり、インターネットに発信されている情報を元に、円滑なコミュニケーションを実現することが期待できる。また、多数の Web サイトのトラストスコアを集めていくことができると、その人の良い点が見えたり、もしくは持っていない能力等を判別できるようになり、より精度の高いコミュニケーションを実現することが可能になる。

スコアリングのために測定する情報は、情報発信者が利用開始時から行っている長期的な情報発信の頻度および評価と、直近の情報発信と評価という短期的な指標を組み合わせる。そのため本研究のスコアで高い評価を受けるためには、一定期間以上は長期的に情報発信をし続ける情報発信者に限られる。しかし、長期評価のみではベテランのみが評価される可能性が高くなるため、若い新進気鋭の情報発信者も評価できるようにし、特定の能力を保有していることへの期待をスクリーニングできるようにする。

スコアは、Web サービスが提供する行動文脈に連動する能力として定義された評価カテゴリに限定する範囲での信頼しやすさを示す。本研究の実施例として、Github を用いたソフトウェア専門職の開発スキルに関する信頼のされやすさについてのスコアリングを行う。本研究においては、Github の Public リポジトリを用いて、自分自身が開発したソフトウェアに関する情報発信と得られた評価を元にスコアリングするため、その範囲を評価カテゴリとした能力への信頼のされやすさを示す。これが全く違う分野であったり、測定対象のパラメータでは測れない領域の能力判断については適用することはできない。これは、それぞれの評価カテゴリに沿ったスコアリングの設計が必要になる。あくまでもトラストスコアは限定された評価カテゴリに対する信頼への期待のされやすさしか示さないため、一つのトラストスコアを元に、その人が人格的に信頼できる人であるかなどへの適用を行ってはいけない。

本研究の制約として以下の行為が行われた場合は正しいスコアリングは不可能である。まず、情報発信者が明確な悪意を持って長期間、本研究の指標の算出にとって望ましい値になるように成りすましを行い続けた場合や、複数のアカウントを用いるなどしてスコアのアルゴリズムをハッキングした場合などが挙げられる。

また、TVタレントなどの有名人のように実社会において高い認知と期待を得ていて、インターネット上に作ったアカウントを公開するだけで他の利用者から高い評価への期待がなされてしまうようなアカウントも正しく評価できない。本研究においては、社会的評価が高い著名人や、明確な悪意がある者については対象とはしない。

本研究のスコアリングのロジックでは情報発信の継続性と評価からスコアを算出するため、対象となるWebサービスが適切な能力を評価カテゴリとして示すように設計されていることが不可欠である。ビジネスとして成立し多数のユーザを抱えているWebサービスであれば、情報発信機能やその評価情報の組み合わせから適切な評価カテゴリを設計できる可能性は高いが、まだ少数のユーザしかいないような未完成、未成熟のWebサービスにおいては、正しく能力を示すためのコンセプト設計、カテゴリ設計に落とし込めていない可能性が高いため、そのようなケースでは本研究のスコアリングの適用は対象外とする。

本研究の成果として、Webサービスの設計において、活躍するために必要な能力を示す評価カテゴリを予め設計し、評価カテゴリに沿った情報発信機能と評価機能を持たせることによってユーザ間で信頼への期待を抱けるようになることで、明確にサービスの有用性を感じるユーザを生み出し、成功するWebサービスの設計論として確立したいと考えている。その結果、インターネット上のWebサービスを継続的に活用することが、人の能力に対する信頼のされやすさを表していくという考え方を定着させることで、インターネット全体の有用性と信頼性の向上に寄与する。

1.3. 本論文の構成

本論文の構成は以下のとおりである。第1章では、本研究で対象とする指標化手法の提案について、本論文の目的として設定した。第2章では、本研究で対象とする信頼について整理する。第3章では、指標の対象とする信頼への期待のされやすさを定義し、指標化に対する基本的な検討を行う。第4章では、信頼への

期待のされやすさを指標化するトラストスコアの提案を行う。算出方法について、先行研究であるクレジットスコアの算出方法や、他の事例から検討を行い、トラストスコアを算出するための必要条件について検討する。第5章では、トラストスコアの実施例として、Githubを用いたトラストスコア的设计を行う。第6章では、実際のデータを用いてGithubによるトラストスコアの算出を行い、第7章で評価を行い、第8章以降で各章で議論してきた事項について総評し、残された課題についてまとめる。

第2章

インターネットで求められる信頼

2.1. 信頼の単位としてのインターネット上のアイデンティティ

Web上にソーシャルネットワークやコミュニティ、コマースなどのサービスを提供するWebサービスにおいては、利用者が利用規約に同意した履歴や、プロフィール、行動履歴等を維持するために、Webブラウザのクッキー等を活用したセッション管理機構を用いて、その利用者が同一人物であるかどうかを保証している(2.1).

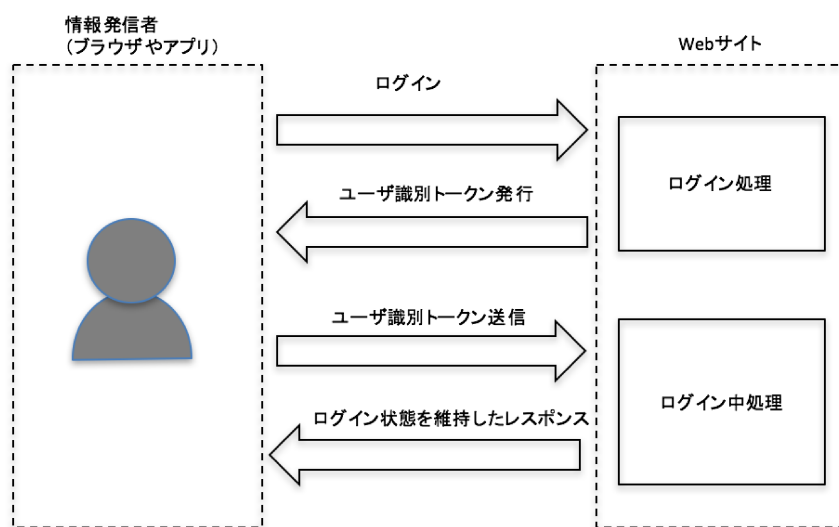


図 2.1 Web サービスのアカウント認証による同一人物の識別ロジック

初回ログインの認証として、ログインIDとパスワードを利用して行い、一度ログインが完了すると、Webサーバは今回のアクセスを継続する用途のセッションクッキーを発行しブラウザに保存する。通常はセッションクッキーはブラウザを

閉じるか数分から数十分という比較的短期間の間に操作されない場合に自動的に無効となりログアウトした状態となる [3]. また、次回ログインの簡便性を高めるために、セッション情報を自動復帰させるためのトークンを発行し、セッションクッキーとは別にクッキーをブラウザに保存させておいて、次回以降、パスワードの入力がなくても自動的にログイン状態を復帰できるように認証クッキーという仕組みを用いてアカウントの継続性を確保する Web サイトも多数存在する. スマートフォンにインストールされる専用のアプリケーションにおいては、ユーザアカウントの同一性を維持するトークンをアプリケーションの内部保存領域に保存し、次回以降の快適なアクセス性を提供したり、プッシュ通知などで利用者に連絡が取ることで継続的にアカウントの状態を維持することが可能である.

Web サイト上のアカウント管理は、原則として Web サービス毎に独立した情報となっている. そのため複数の Web サービスに登録しているアカウント同士が、同一の人物であることを簡単に識別する方法はシステム側には存在しない. そのような状況下で、特定の利用者が複数の Web サイトにアカウントを作成した場合に、友人に対して複数のアカウントが同一の人物の所有であることを認識させるためには、以下のような方法が考えられる.

- 共通のアカウント名や共通のプロフィールを設定し、同じ人が所有しているアカウントであることを判別できるようにする.
- 友人にアカウント情報を直接送付し、新しい Web サービスの友達登録などを行う.
- 自身が保有するブログやプロフィールサービスなどの第三の場所に、所有しているアカウント情報を記述して同一性を主張する.

これらの情報は、厳密には第三者のなりすましなどによって、その人を知っている人達が騙されることを完全に防ぐことは難しい. しかし、有名人でなければ、なりすましなどで攻撃される必然性は低いため、ほとんどのケースにおいて問題が起きないと考えられる. よって、これらの方法を通じて、複数の Web サービスをまたいで同一人物であることの表明が可能である. Web サービス自体、流行り廃りの影響を受けたり、運営者の変更、運営企業が買収されてサービス性が変更されたりすることがあるため、インターネットを日常的に活用する人であれば、一つの Web サービスだけを活用し続けることはなく、複数の Web サービスを活用す

ることとなる。本研究においては、このように他人から同一人物としてみなされている、もしくは、本人にとって同一人物と見なして欲しいと思っている一つまたは複数の Web サイトのアカウントの集合体をインターネット上のアイデンティティと定義する。

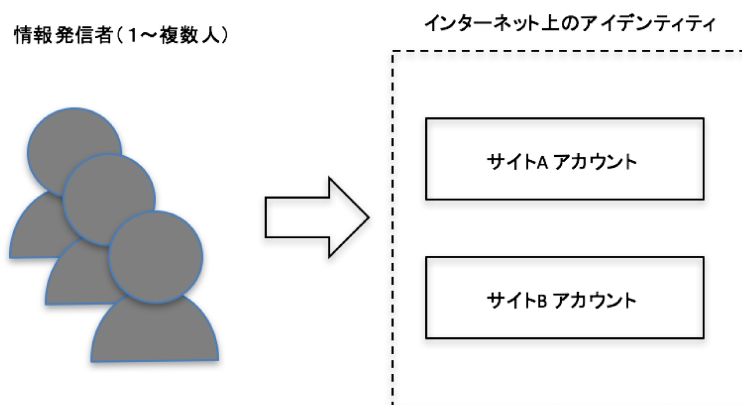


図 2.2 情報発信者とアイデンティティの関係性

インターネット上のアイデンティティを構成する Web サービスのアカウントは、多くは個人が 1 人で管理しているが、企業の担当者アカウントやサークルやチームのアカウントであれば、複数人で運用することもあることからアイデンティティは 1 人の人間を示すとは限らない。このようにインターネット上のアイデンティティは、必ずしも実在する個人に対応するとは限らない。しかしアカウントの一貫性は維持されていないと対話が実現しないため、関連する個人は複数人かもしれないが、企業理念やソーシャルメディアポリシーなどを通じて、同一視可能な人格という一貫性は実現していることが求められる。

折田 (2016) は、アイデンティティの見せ方と管理において、個人情報保護法改正議論において整理されていた個人の「特定」と「識別」という区別を用いて、必ずしも実名を名乗るなどの「特定」された状態でなくとも構わないが、同一人物であるという「識別」はなされる必要があると述べている [4]。

本研究においても「特定」可能性ではなく、同一人物、同一視可能な人格であるという「識別」可能性があることを重視する。つまりインターネット上から「識別」可能なアイデンティティに対して信頼を寄せることができるかについて検討

していくこととする。

2.2. インターネット上のアイデンティティの特徴

インターネット上のアイデンティティの特徴として、仮に一つの Web サービスのアカウントのみをアイデンティティとしている場合、そのアカウントを削除してしまえば、そのアイデンティティは消失してしまうことにある。昨今、ソーシャルネットワーキングサイト上のコミュニケーションで揉め事を起こし、他人から言葉で攻撃されるようなトラブルを起こすと、すぐにアカウントを削除し、新たにアカウントを作りなおす行為は頻繁に観察されている。これはアイデンティティを作り直すことで、新たにインターネット上の関係性を作り直すことができるメリットがある。しかし、この方法では、継続的な信頼関係を結べないため、インターネット上ではいつまで経っても他人との信頼関係を得られないこととなる。実生活における知人や友人に関しては、アイデンティティを作り直したという情報を直接連絡を取ることで、新しいアイデンティティと本人の紐付けを知ることが出来る。このようなアイデンティティの運用では、本研究の対象としている見知らぬ人に対する信頼を期待することは難しい。

また、単一の Web サービス上で複数のアカウントを作成し、繋がる相手や発言の内容によって使い分ける「メインアカウント」「サブアカウント」などと、アイデンティティを使い分けるケースも存在する (図 2.3)。例えば友達の間だけで本音を共有するためのアカウントと、就職活動のために面接官からツイートを見られてもよいアカウントに分離するなどの用途が考えられる。同様に、TwitCasting[6]で動画配信する際に、不特定多数のファンと繋がるためのアカウントなどの外向きの自分を表現するなどのケースが考えられる。 [5]

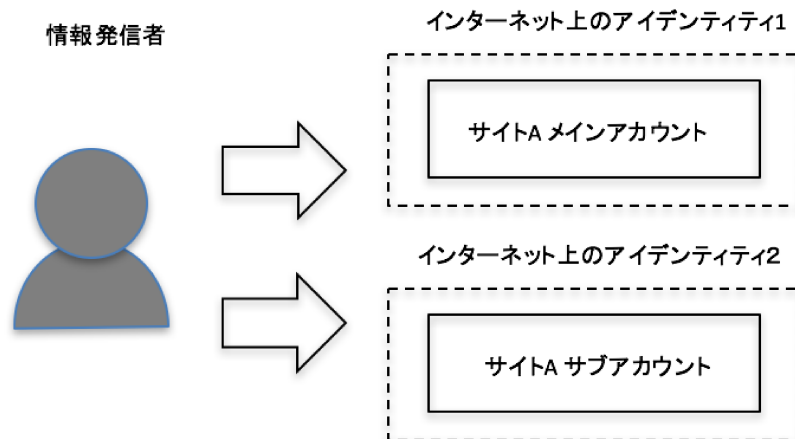


図 2.3 サブアカウントとアイデンティティの関係性

このようなケースでは、アカウントを保有する本人が相互のアイデンティティの関連性を否定し、インターネット上に公開するアイデンティティを使い分けたいと思っていることから、それぞれ別々のアイデンティティが存在するものとして考える。仮に、本研究のスコアリングを通じて、どちらかのアイデンティティに対する信頼への期待のされやすさを持たせることができたとしても、もう一方のアカウントとは無関係である。仮に、友達と本音を語り合うサブアカウントのアイデンティティでは不誠実で信用できない発言をしていて、外向きのアイデンティティでは、優等生的な発言をしていたとしても、相互のアイデンティティが切り離されている以上、二つのアイデンティティの評価を組み合わせた信頼への期待のされやすさの判断はできないことになる。このことから、インターネットのアイデンティティに対する信頼への期待のされやすさについては、実社会の個人への信用と結びつけることは難しく、限定的な評価に留めておくべき理由となる。

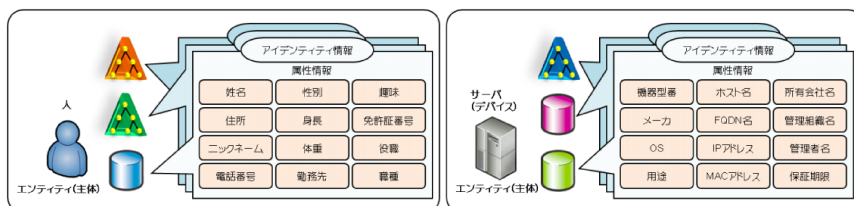
もしインターネットの情報から得られる信頼への期待のされやすさを、個人への信頼として評価する場合は、評価する側が実社会における個人を知っているか、本人が個人情報を公開しているケースにおいて、インターネット上に抱いた信頼と個人の信頼とを結び付けられる場合に限られるべきである。仮に特定のスポーツのファンが集うソーシャルネットワークを考えると、SNS上では継続的にアドバイスをを行い、高い評価を得ているようなアイデンティティがいたとしても、同じ本人にお金を貸した場合に、そのお金が適切に返済されるとは限らない。しか

し SNS 上での実績を元に、スポーツのアドバイザーの仕事として採用するのであれば、活躍することへの信頼に対する期待を抱くことはできると考えられる。SNS 上で培われた信頼への期待が、スポーツのアドバイザーという能力において認められたからである。

なお、タレントや政治家など、その存在が社会的にすでに認知されていて、その名前を公表してインターネット上のアイデンティティを構築する場合は、知名度を下地とした個人に対する認知やイメージが信頼のされやすさに繋がってしまうため、そのような存在は本研究の範囲外とする [7]。

2.3. アイデンティティ管理技術におけるデジタルアイデンティティとの関連性

アイデンティティという言葉を用いるにあたり、アイデンティティ管理技術で用いられているデジタルアイデンティティとの関連性を述べておく。アイデンティティ管理技術とは、人々がオンラインを活用する際のアクセス認可、認証管理、利用者の属性情報交換などに関する技術を総合的に示す概念である [9]。ISO/IEC 24760-1[8]において、アイデンティティ管理技術におけるアイデンティティを「実体に関する属性情報の集合」と定義している。デジタルアイデンティティにおいては人や組織などをエンティティ（主体）と定義し、システムに登録されたエンティティに関する属性情報の集合をアイデンティティ情報と呼ぶ。図 2.4 に示すアイデンティティ管理技術におけるデジタルアイデンティティの概念によると、一人のエンティティが複数のアイデンティティを持てるなど本研究のアイデンティティの概念と非常に近い概念となる。



アイデンティティ管理技術解説 [9],p5 より引用

図 2.4 エンティティとアイデンティティ情報

ただし、本研究におけるアイデンティティの単位は、アイデンティティという言葉の翻訳における「自我」の方が近いと考えており、近い概念でありつつも直接的に関連性を意識したものではない。デジタルアイデンティティで想定されているアイデンティティは、主体となるエンティティを人間と定義した時に、その人が複数のアカウントを保有した場合に、複数のアイデンティティを持つことができるという考え方で整理されているが、本研究におけるアイデンティティでは、エンティティである本人から見て、本アカウントやサブアカウントなどのようにインターネットにおける存在そのものを使い分けたい場合には、人に対して複数のエンティティを持つ必要がある。デジタルアイデンティティは情報管理技術の考え方なので、人が複数のエンティティを持つことは想定されていないかコンセプトの範囲外であると考えられる。一方、本研究においては、それぞれのアイデンティティにおいて人間関係や信頼関係を使い分けることができたり、アイデンティティを削除したり作り直すことが可能なことは重要な前提条件である。本研究をデジタルアイデンティティが想定しているアイデンティティに対する属性情報の一つとして捉えることで整理可能だとは考えられるが、アイデンティティという言葉自体が想定している範囲が違うため、この議論においては本研究の対象外であると述べるに留める。

2.4. コミュニケーションに必要とされる信用

街中で二者がコミュニケーションをしているケースを考えてみる。相互に存在を知っている場合と、そうでない場合では、どのような差異が起きるだろうか。駅で待ち合わせをしている女性が後ろから男性に肩を叩かれて振り返った時に待ち合わせ相手であれば、待ち合わせは成立し肩を叩かれたことは、特に問題なく許容される可能性が高い。しかし肩を叩いた人間が全く知らない男性であったら、痴漢として交番に突き出されてしまうかもしれない。この違いにおいて肩を叩いた男性は、待ち合わせをしている女性に対して、「肩を叩いても痴漢として突き出されない」ことに対する期待を抱いていて、実際に期待通りにコミュニケーションが成立したので、問題が起きなかったと考えることができる。

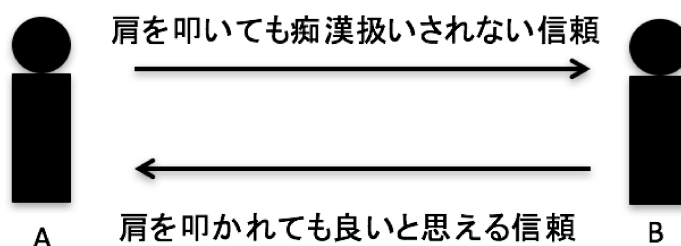


図 2.5 駅の待ち合わせのコミュニケーション成立に必要な相互の信用

信用という言葉には「それまでの行為・業績などから、信頼できると判断すること」という意味があるが、相手の肩を叩いても痴漢として突き出されないという信頼に対する期待を抱けるからこそ、肩を叩くという判断をしたとも考えられる。その要因は、お互いが知っている人同士で、人となり判断しているからこそできることだと言える。この判断を確信するために抱いているものが信用である。

しかし、仮にその男性が会社の上司だった場合、実は肩を叩かれたことを不快に思っている可能性があるかもしれない。それについては女性自身が相手に表明しない限り、不快であるという情報が出てこないため判断は不可能である。また、その女性が日頃、ボディタッチされることに対する不快感を表明していたとしたら、男性は肩を叩く以外の方法で、その女性のことを呼ぶ可能性が高い。つまり待ち合わせで相手呼び止める行為として、肩を叩くことが不快に思われないで

あろうと期待できるか否かは、それまでに得ることができていた情報によっても判断が変わる。

女性が実は嫌がっているという情報を知らないのであれば、そのことを汲み取りようがないため、肩を叩くか否かは、男性の側の性格や経験に依存してしまう。男性の側が女性に抱いている信用は真実としての信用を示すわけではなく、常に相互の関係性に依存し変化するものである。相手を信頼しコミュニケーションを行うという行為は、それまで得ていた信頼を元に判断するしかなく、得られた情報の積み重ねからこの行為は相手は受け入れてくれるであろうと都度都度、期待をしながら行動することが求められる。

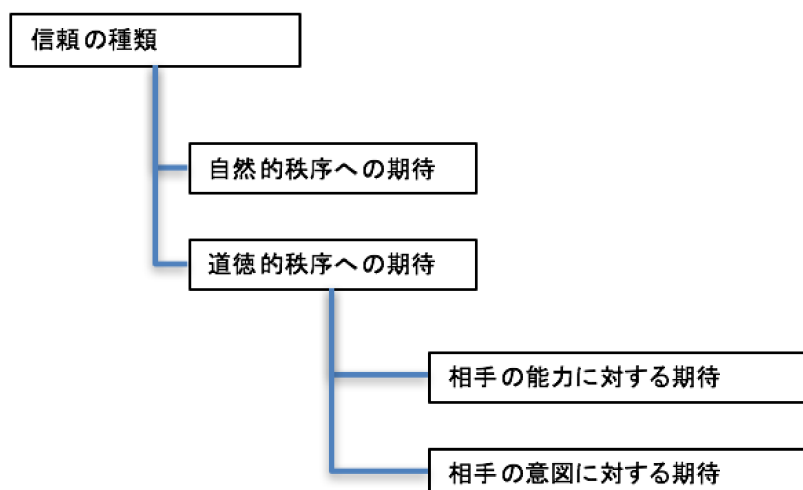
コミュニケーションを取る際には、そのコミュニケーションがうまくいくであろう信頼から、相手が受け入れてくれるであろうという期待をし、取りうる行動に対する意思決定をしていると考えられる。それぞれが相手に対して抱いている信頼が、現実的にどう考えられているのかは、人間の心の中にある感情のため解き明かすのは不可能であるが、表出している情報から特定の行為についてならば望ましい結果になることへの期待ができそうだという計測は可能であると考えられる。

2.5. インターネットのコミュニケーションにおける信用

インターネット上のコミュニケーションにおいては、相手を実生活上の知人でない限り、お互いが得られる情報は限られている。相手の本性はおろか、どのような仕事をしているのか、性別すら不明なこともある [10]。そのため、たとえインターネット上では見知った関係性でも限定的な関係性であることを認識してコミュニケーションを取っていく必要がある。ところがインターネット上のコミュニケーションにおいて一切の信用を抱けない場合、相手とコミュニケーションをし始めることや、ネットショップやオークションで取引を行うのは難しくなり、インターネットを活用するのは難しくなってしまう。信用の判断が全くできない状態では騙されてしまう可能性も高くなるため相手を盲信するのは得策ではないものの、なんらかしらの手段で信用判断を行い、多少なりともリスクは覚悟した上でコミュニケーションを取ることが求められる。実生活における信用が、主に対面で会っていることを踏まえた信用だとすると、インターネットにおける非対面のコミュニケーションは、それ以外の方法で信用を担保していくしか手がない。一般的に信頼、信用という言葉は、「この人は信頼できる人なので、犯罪を犯さない

はずだ」などの、人格的に望ましい人物であると判断することや、社会倫理に照らしあわせた上での期待を含むことがある。しかし、本研究において検討している信用は、こういったことを保証しない。山岸らが研究する信頼の構造 [11] によると、社会的不確実性の高い環境において必要されるものを信頼と定義しており、本研究もその考え方に従う。

図 2.6 に山岸らが述べている信頼の分類を示す。



信頼の構造 - こころの社会の進化ゲーム,[11],p47 を元に作成

図 2.6 信頼の分類

山岸らの研究によると、Luhmann (1979) は信頼の種類を、自然的秩序への期待と道徳的社会秩序への期待の 2 つに分類している。自然的秩序とは、「日はまた昇る」などと言った世の中に存在する秩序や規則性に対する確信を信頼とするものであるが、山岸らが研究対象にしている道徳的社会秩序への期待については、Barber (1983) が提唱した「能力に対する期待」と、「相手の意図に対する期待」に区別している。

能力に対する期待とは、目的とする行為を適切に実現する能力を持っているかであり、意図に対する期待は、目的とする行為にやる気があるかを判断して期待することを言う。このような期待を相手に抱くことができることを社会的不確実

性が高い状況における信頼と定義付けている。

本研究においては、この「相手の能力に対して期待すること」または「相手の意図に対して期待すること」を重要視する。何故ならインターネットに発信されている情報は、発信者が発信したいと思った情報で構成されているからであり、見知らぬ情報発信者に対して信頼判断しなくてはならない状況は、まさに社会的不確実性が高い状況において、その情報を信頼できるか否かということに他ならない。発信されている情報に対して、それが正しく遂行できそうであるという「能力」もしくは「意図」に対して期待をし、それが期待通りに遂行されれば信頼が生まれる(図2.7)。その経験の結果、情報の受け手は特定の能力について、この発信者を信用できると判断することが可能になる。

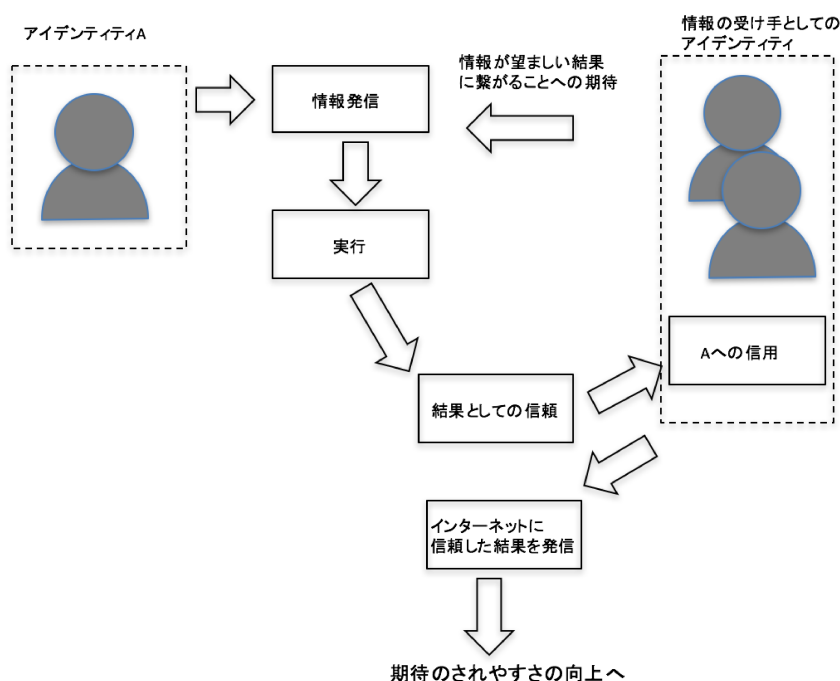


図 2.7 信頼への期待と結果としての信用

信用そのものは情報の受け手が結果として心の中に抱くものであるため、定量化して把握することは不可能である。しかし、信用判断に辿り着く手前の段階に、相手を信頼について期待できるかについて、判断のステップが必ず存在する。

現状、インターネットで探した商品がどうしても欲しい場合には、検索エンジ

ン等を通じて見つけたネットショップに対して、特に信頼できるかどうかを確認することなく商品を購入するような行為は日常的に行われている [12]. 多くのケースにおいて取引は正しく成立しているものの、一定数は商品が届かないなどで騙されてしまう事実も存在しているが、信頼できるショップかどうかの判別は困難なことも多い.

そのようなインターネットを介した商取引の際にも、ネットショップのオーナーが日常的に行っている正しい活動を評価する方法があれば、購入者が購入する際に、ネットショップが適切な取引をする能力や意図を有しているかを判断できる. それによってアイデンティティであるネットショップのオーナーは、普段の活動を通じてより高い評価を得られるようになり、販売機会を生かすチャンスが増える. この問題を解決するために、情報発信者が発信した情報を元にした「信頼への期待のされやすさ」に着目する. 情報発信者がインターネット上で行っている情報発信という行為を発信者の信用構築へ結びつける方法論を検討していく.

第3章

信頼への期待のされやすさ

インターネットの不確実性の高い状況において、情報発信者が情報を発信することに対し、信頼について期待されるために必要な要素について検討する。

3.1. 情報発信の継続性

まずは情報発信を継続することについて検討する。例えば、あるアイデンティティが、あるテーマでイベントを行うために匿名のままブログに記事を書き、参加者を募り、先払いでお金を集めたとして、お金を払った参加希望者に対して、その役務が正しく遂行されたかどうかは当日になってみないとわからない。参加希望者が先払いでお金を払うのは、そのアイデンティティに対して「イベントが適切に開催される期待」を持ったからである。もし、開催者が全く信用されていないのであれば、参加を躊躇しお金を支払う人はいないはずである。

「イベントが適切に開催される期待」を得る要素として、発信しているイベントのテーマに共感することや、過去に企画されたイベントが期待通りの成果をしているという実績などが挙げられる。その他、所属している企業や実名を明かしていることや、登壇者が既に得ている信用を借りることなどが考えられ、さまざまな要因から期待を生み出すことができる。その後、イベントがスケジュール通りに適切に開催され、その品質が参加者の期待通りのものであったならば、そのイベントに対して信頼が生まれる。

信頼が生まれると次回開催においても「高い品質のイベントに参加できるという期待」を得ることになる。そのような信頼の元、イベントの評判としてインターネット上に口コミが発生することで、以降のイベント開催において集客等がしやすくなり [13]、イベントの継続可能性が高くなる。集客のしやすさは、イベントや運営者の知名度と提供品質に対する期待と連動するが、知名度が高くても品

質が悪ければ参加者の継続率が下がることになるので、長期的にはイベントの継続性は知名度よりも、結果としての信頼に依存する。

このような活動を続けているアイデンティティは、同一人物であるという識別可能性を前提として、イベント参加者からイベントのクオリティに対する信用を得ることになる。信用を得ることで、次なる期待を得られやすくなる。つまり、継続的な活動に対して、次の取り組みという不確実な行為に対する品質実現への期待が生まれる(3.1)。そして、期待通りの品質が提供された時に、受け手が抱くものが信頼である。

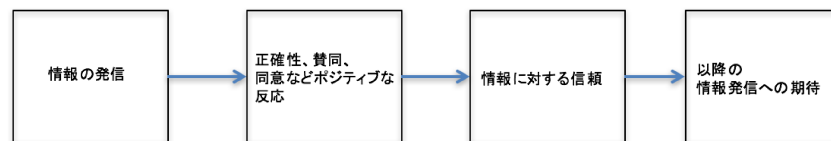


図 3.1 情報発信の継続性から得られる信頼への期待

このような継続的な活動において情報の受け手は、発信者に対する信用の他に、次なる取り組みへの信頼への期待を抱いていると考えられる。この期待感があるからこそ、相手がまだ信用できることが明確になっていなくても、コミュニケーションや取引を行いたいと思えることが可能であると考えられる。この「信頼への期待のされやすさ」をインターネット上の情報から指標化することができると、適切な情報発信者であればあるほど、新たな期待を伴ったチャンスを得られる。その際の指標化の重要な要素として、情報発信の継続性を考慮することが不可欠である。

3.2. 情報発信に対する反応

次に情報発信に対する反応について検討する。インターネットでは情報発信者が発信したいと思った情報が発信されることが特徴であるが、その情報に対し、閲覧者の反応が情報として可視化されることが大きな特徴である。可視化される情報とは、ソーシャルネットワーク上で感想と共に URL のシェアされた情報、いいねボタンが押された情報、フォロワーへの登録、ソーシャルブックマークへコメントの登録、ブログに感想の書き込みすることなどである。[14] [15].

本研究においては、情報の受け手による評価や反応が Web サービス上で可視化されていることに着目する。つまり Web サービス上における評価や反応は、情報の受け手であるアイデンティティが新たに発信した情報であり、情報発信者に対する測定可能な評価行動として考える。

その反応が肯定的な評価であれば、一般的に良いとされている行動を行い、否定的な評価であれば、否定的な感情に基づいた情報が発信されることがある。また、それ以外の状態として「価値を感じなかった」というケースも存在する。もし、情報の受け手が、全く価値を感じなかったのならば、そのことについて手間や時間をかけてまでインターネットに情報を発信することはしないと考えられる。このような状況を無関心と定義すると、無関心という評価をインターネットから測定するのは困難である。

表 3.1 情報の受け手の感情分類

評価	情報のシェアや反応	観察可能な感情
肯定的	する	肯定的
	しない	計測不可
特に評価なし	する	フラット
	しない	計測不可
否定的	する	否定的
	しない	計測不可
無関心	しない	計測不可

表 3.1 に分類するように、情報の受け手が肯定的もしくは否定的な感情に対して、実際に感想を書くほどの関心を得た場合は、インターネット上で感情に対する反応を測定することができる。このように発信した情報に対する反応については複数のパターンが想定されるが、すべての閲覧者から肯定的な反応を得ることはできないし、すべてが否定的に捉えられるわけでもない。多くのケースにおいては無関心として反応を得ることは難しいことを意識しながら情報発信を心がけることは情報継続を行うモチベーション維持にとっても重要な要素となる。

3.3. 情報発信の継続と反応の組み合わせ

情報発信の継続性と可視化される評価について検討したが、その二つを組み合わせた場合に起きる現象を検討する。情報発信を継続するにあたって、発信した情報に共感を得ている場合には肯定的な反応を得られるが、否定的な反応については、Web サービスに備え付けられているコメント機能やSNSを通じて可視化される。とりわけ情報発信を継続する場合に起きる否定的な評価は発信者への評判を毀損する反応と捉えることは可能であるが、あくまでも人間対人間のコミュニケーションであるため、その思惑にはいくつかパターンが想定される。

例えば、単純に相手が憎くて仕方がないというケースもあるだろうが、相手が有名人で相手にして欲しいなどの期待から、あえて否定的に振舞っているケースや、特定の知識に詳しいという理由で教育的な目的で振舞う、特定の目的を意図して否定的に振舞うなどが考えられる [16]。そこに愛情があるケースもあれば、愛着があるネットワークを守るために戦っているケース、もしくは単純に面白がって否定的に振舞うケースなど、動機は多数想定できる。これらの本当の動機はインターネットからは伺い知ることは困難なことが多く、本当はこう思っていたということを字面だけで判断することは不可能で、もちろんテキストマイニング等で分析することも不可能である。

また、インターネットにおいて否定的な反応は、一般に肯定的な反応よりも多く見られる。また、肯定的な反応がWeb サービスに具備されているいいねボタンやコメントなどで情報発信者に直接表明され、多くは当人同士の閉じたネットワークを構成する傾向にあるのに対して、否定的な反応は他者に共感を求めるために広く発信されやすくなる。更にアイデンティティが情報発信を継続していればするほど、過去の情報で形成された印象や文章の中にある落ち度などに対して指摘のされやすさも向上していくため、感情的軋轢などを下地とした批判的な反応も発生しやすく、単体の意見に対する議論よりも複雑な動機に基づく否定的な反応も少なくない。

批評という行為も否定的な意見で構成されると考えれば、圧倒的に批判的な反応の方が多く、肯定的な評価とは別に整理が必要である。そして、インターネットで観測可能な否定的な評価に対する反応は、他のアイデンティティによって否定的な発言が表明される行為であるが、通常の場合において批判される意見と共にURL等が共有され、元の情報発信者への参照情報が共有される。参照元の情報がなければ批判の対象を他者と共有するのは難しいため、元の情報を晒す形で関

覧者に共感を求める。

本研究において否定的な反応は評価の一つと捉えるだけでなく、URLを共有するという特徴から否定的なラベルがついてはいるながらも情報の伝播手段と捉えて議論を進める。否定的な評価に付随して、新しい情報流通経路を通じて情報閲覧者を獲得するきっかけとなるという考え方である。

このように情報発信者であるアイデンティティにとって否定的な情報発信が他のアイデンティティによって行われた際に、そのまま情報発信を継続するためには、どのような障壁が起きて、なおかつ情報発信者がどのように評価を得ることができるのかについて、炎上と呼ばれる現象を元に検討を続けることとする。

3.4. 炎上後に情報発信を継続することの重要性

情報発信の継続性を維持する重要性について、炎上と呼ばれる現象から述べる。炎上とはインターネットへの発言をきっかけに、沢山の人のによって否定的な評価への表明が連鎖的に行われてしまう現象である。平井(2012)[17]によると、ブログ、ミクシィ(mixi)、ツイッター(Twitter)などに投稿されたメッセージ内容、ならびに投稿者に対して批判や非難が巻き起こる現象と定義し、田代・折田(2012)[18]は、情報発信者が管理するブログやSNS日記などの個人向けCGMに嫌がらせコメント等が殺到する現象であるとしている。また、平井(2007)[19]はその背景に存在するものとして、ネットにある空気＝「規範」に着目し、炎上の発生要因をネットの発言が、諸個人・諸集団を取り巻く規範に抵触するような行為が見られる場合などに発生するものとして定義している。

インターネットを積極的に活用し、情報発信のハブとなるインフルエンサーと呼ばれる利用者層が否定的に取り上げる情報については、とりわけ否定的な反応の連鎖が起こりやすい。例えば、発信する内容が犯罪自慢やモラルに反する行為の自慢のようなケースであることが挙げられる。友達の間であれば相互の信頼関係の元で許されるような話題でも、ソーシャルメディアを介してインターネットから誰でも閲覧できる状況になった際に、その話題に対する否定的な反応の連鎖が生まれることがある。

しかし、炎上の状況で見られる現象として、情報発信者の思想、行動規範など信念に基づく発言が、そのネットワーク上に存在する多くの人たちにとっての規範に反する場合に炎上起きることがある。つまり、犯罪自慢やモラルに関する

行為の自慢など明らかな非が認められる情報ではなく、生活水準や職業、身分が違うことでの認識が折り合わない時に、行き違いが発生するようなケースである。

例えば、経営者を経験していなければ得ることが難しい視座や知見について、経営者を経験したことがないアイデンティティに伝えた時に、経験に対する視野の違いから炎上状態となり、否定的な反応が連鎖してしまうケースはしばしば Web 上で見られる。このようなケースにおいて、情報の発信者が発言を撤回することなく物議を醸し続けた場合には、一見否定的な発言で埋め尽くされて、あたかもこの経営者は間違った発言をしているように見えるかもしれないが、実際は一定の支持者層が存在することがある。否定的な情報を発信している人たちを情報の伝搬に加担している人たちとみなすと、その情報の閲覧者は流れてきた否定的な話題に対して、その情報のどちらが信じるに値する情報なのかを判断していると考えることができる。

炎上状態にあるアイデンティティをアイデンティティA、その否定的な反応を表明した者をアイデンティティBがそれぞれ情報を発信している場合、その情報を閲覧したアイデンティティCの反応が計測できた場合に、どのような反応を示すかについては、表 3.2 に示す判断が行われると考えられる。

表 3.2 炎上時における閲覧者の反応

判断	現象
A の発言を信用 > B の発言を信用	炎上状態にある A を支持
A の発言を信用 < B の発言を信用	否定的な B を支持
A,B の発言は信頼に足らず	どちらも支持しない
不明	判断不可, 無関心

アイデンティティCとアイデンティティBが友達であれば、アイデンティティCはアイデンティティBの発言を信用しがちかもしれないが、冷静に両方の発言を判断して、どちらの発言に対して信頼の期待をするかは、アイデンティティCの経験、性格や判断力に依存することになる。

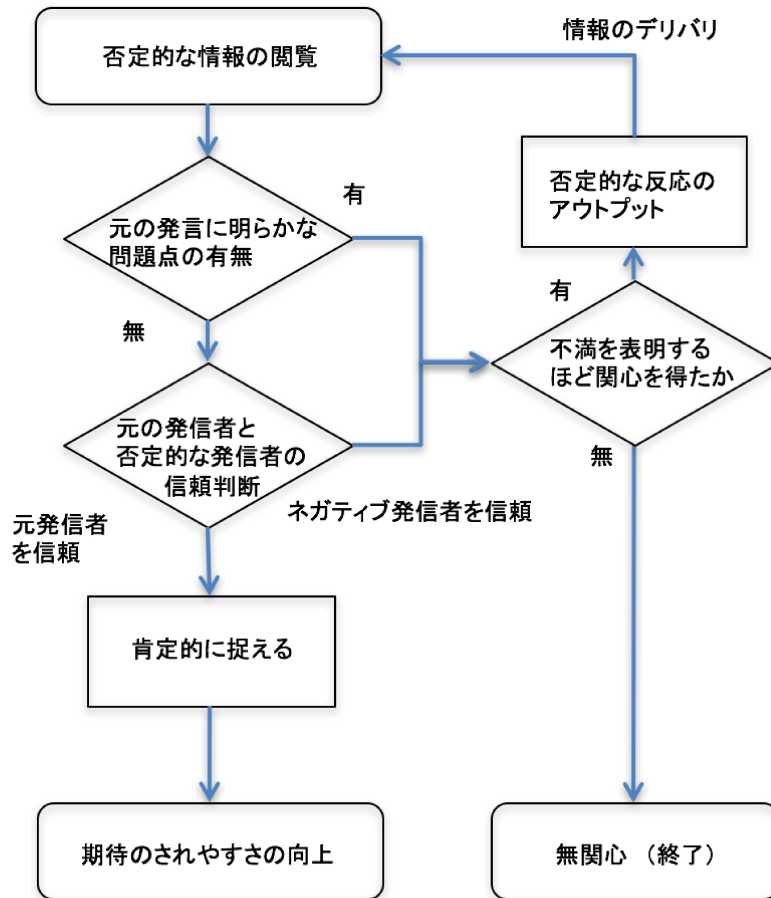


図 3.2 否定的な反応が連鎖する情報のデリバリ

炎上状態が否定的な意見のシェアを誘発したとすると、図 3.2 に示すように次なる情報閲覧者を生む。情報閲覧者が無関心でなければ、インターネット上に何らかしらの反応を示すことが期待できる。その際に、元の発言者に対する信頼への期待を抱くこともあれば、逆の反発を抱くこともある。好意的な反応が連鎖することもあるが、多くは否定的な反応だからこ他の人にも連鎖すると考えられ、炎上に伴う情報の情報の伝搬は、新たな情報閲覧者の獲得を産む。この際に、否定的な意見に影響されなかった閲覧者からは、新たな賛同や意見に対する信頼を得る可能性がある。

このような現象を利用したものとして、Web上の情報共有を広めるために炎上マーケティング [20] と呼ばれる、意図して炎上させることで商品の情報や、自分の思想を伝達したりなどの行為はよく知られている。炎上マーケティングが存在する理由は、インターネット上で意図的に炎上させることで生まれるマイナスの感情よりも、人から注目され商品や情報の認知を得るメリットの方が利益が大きいという考え方ができるからである。アテンションエコノミー [21] という考え方では、情報過多時代において他人からの興味、関心を得ることの重要性が認知されており、その一つの方法としてたびたび活用されている。

しかし、このような行為を計算づくではやっていない多くの非炎上者は、否定的な評価表明の連鎖によりアイデンティティの継続の意思を失いアカウントを削除してしまったり、情報発信を自粛するケースが多々観察されている。

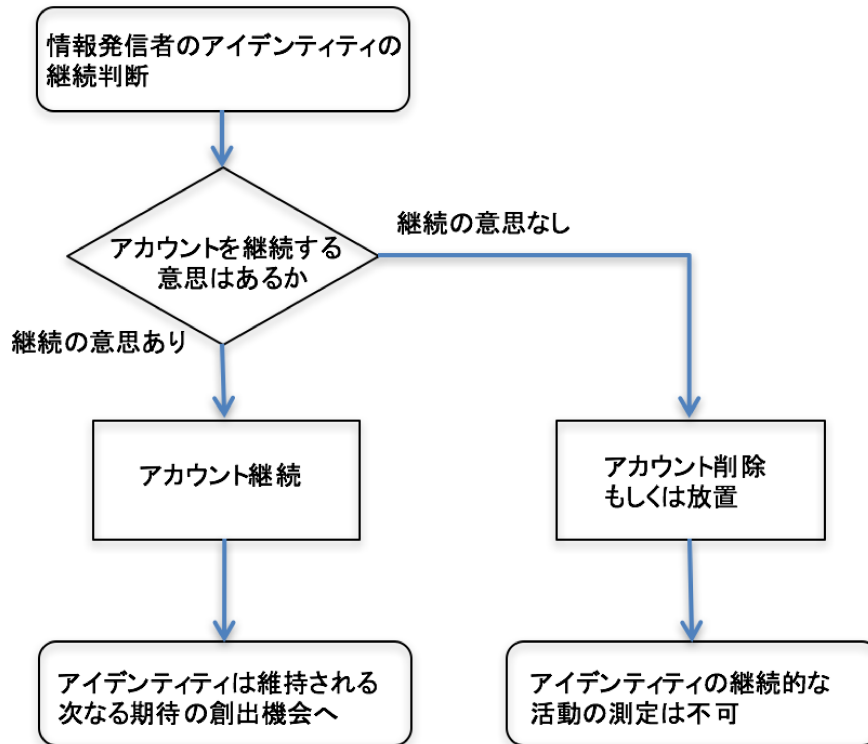


図 3.3 炎上後のアイデンティティ継続

図 3.3 に示すとおり、仮に炎上状態になった際に、本人の意思によりアカウントの継続を辞めてしまえばアイデンティティの継続は途切れるが、継続の意思があれば情報発信は継続可能である。炎上にめげずに情報発信を継続するアイデンティティは、その後新しいファンを獲得したり、その後の成長を認められて周囲からの評判が変わることがあるため、これらの特性を理解し情報発信を継続することが何よりも信頼への期待のされやすさを獲得していくためには不可欠である。

また、仮に周囲から見限られるようなアイデンティティが存在した場合は、情

報を発信しても閲覧される価値がないと判断され、ページビューが下落するなど
で評価が下がることから計測が可能である。仮にそのようなアイデンティティが
存在した場合は、情報発信の継続性に加えて、評価を計測することで、信頼への
期待のされやすさが縮小していくことを推し量ることは可能だと考える。

3.5. Web上で可視化されている評価情報

さまざまなテーマにおいてコミュニケーションを促す機能を提供する Web サービスでは、コミュニケーションを促進させるために、アイデンティティによる情報発信に対する他の利用者からの評価を公開している。代表的な Web サービスとアイデンティティへの評価の情報を表 3.3 に示す。

表 3.3 可視化されている評価情報

Web サイト	評価情報
Github	Starring 数, Watch 数, フォーク数
Instagram	フォロー数, いいね数
クックパッド	つくれぽの数, コメント数
Twitcasting	閲覧数, アイテム受取数, 通知登録数

これらの公開情報を評価値とみなした場合、その数値が高いユーザは、そのサービス運営側からは、望ましい成績の優良ユーザであると判断できる。アイデンティティ自身は、自分が行った情報発信に対して「いいね」が沢山つくことを嬉しいと思えばこそ、質の高い情報発信を心がけ、他のアイデンティティに喜ばれる情報発信を継続するようになるということを考えると、各 Web サービスが公開している評価情報は、その Web サービスを使う利用者の行動原理を規定することになると考えられる。

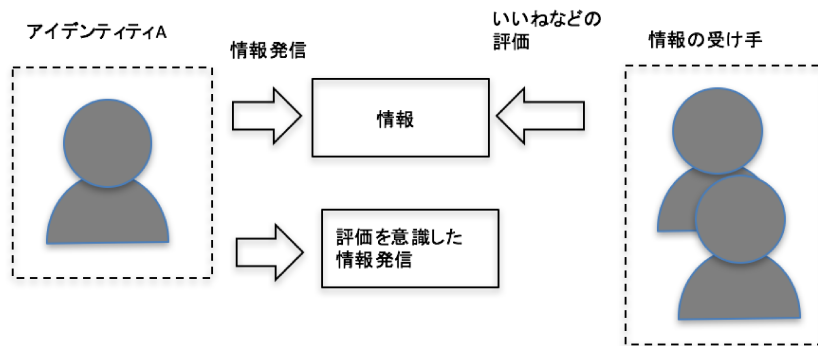


図 3.4 評価を意識した情報発信

多数の利用者を獲得し継続的に活用されている Web サービスにおいて、可視化された評価の存在そのものが、Web サービスが支持される一因として考えられる。例えば、図 3.5 のように Instagram のユーザアカウントのページにはフォロー数が目立つように表示されており、この数字の多寡でアイデンティティが Instagram でどれくらい活躍しているかを把握できると共に、この数字をあげることは望ましい行為であると教育されていると考えることもできる。自分が投稿した写真に沢山の「いいね」がつく要因の一つとして、高い頻度で関心を持ってくれるフォロワー数が多いことが望まれる。そのため Instagram をより楽しむためには、フォロワーを増やすことが望ましいと考え、閲覧の接点を増やすために写真にハッシュタグなどをつけるようになる [22]。

もし評価値が可視化されていることで、その Web サービスの利用にストレスを受ける利用者が多いのであれば、その数字を公開していることを要因として、その Web サービスに定着しない可能性が高いので Web サービスの運営側は公開を控えるであろう。数字を公開し続けることができるのは、この数字の存在が無意味であるか、利用者に友好的に受け入れられているかのどちらかである。しかし、スマートフォンのアプリケーションや Web ブラウザの小さな画面でサービスを提供している場合、限られた画面の面積に無意味な数字を表示する理由はないので、この数字が表示され続けるということは、表示された数値の向上を望ましいとする利用者がマジョリティとして存在していると考えることが可能である。つまり、公開されている数値情報は、その Web サービスが最も重要なテーマとしている行動特性を示しており、その Web サービスが使われ続ける理由として欠かせない重

要情報と考えられる。よって、画面上に表示されている数値が高いことは、アイデンティティがその Web サービスにおいて発揮している能力に対する評価を示すと考えることが可能である。



図 3.5 Intergram の画面上のフォロー数表示

3.6. Web サービスが指し示す能力としての評価カテゴリ

もし写真共有の SNS で、写真について沢山のファンがついていることがわかるなら、そのアイデンティティは、写真のアウトプットに対して人の心を揺さぶるほど有益な情報を提示していると考えられる。動画サイトであれば、動画における表現力やファンを獲得する面白さについては期待できると判断できる。それぞれの Web サービスで可視化された評価が、その Web サービスが持つ文脈に対して望ましい評価であるならば、そのサービスが指し示す能力に対する「期待のされやすさ」が高いと考えることができる。

各 Web サービスから取得可能な評価情報から「その Web サービスが指し示す能力についての期待のされやすさ」を指標化することを考える際に、Web サービスが指し示す能力を「評価カテゴリ」として定義づけることで、高い評価を得られているアイデンティティは、その評価カテゴリについて信頼への期待を抱くことができると判断ができる。

表 3.4 Web サービスが指し示す能力の評価カテゴリ

Web サービス	行動	評価カテゴリ
Github	ソースコードの発信	優れたソフトウェア開発者
クックパッド	料理レシピの公開	美味しい料理を作れる人
Twitcasting	動画の生放送	動画における振る舞いがうまい

プログラミング技術を得意とするアイデンティティが、オープンソースソフトウェアを開発し、githubにソースコードを公開したとする。その行為に対して Starring がついたり Watch に登録して今後の動向を注目する人が多数出てきた場合、そのアイデンティティが持つプログラミング能力や製品開発力、企画力などのエンジニアとしての能力が評価されたと考えることが可能である。その一方で、そこで評価されたことが、料理のうまさや運転技術に対する能力を有するか、更に人格的に優れているであるとか、異性関係にルーズであるなどと言ったことは判断不可能である。つまり、Web サービス上での評価情報から得られる信頼への期待のされやすさの対象となるものには評価可能な範囲が存在し、その範囲に限定するべきである。

また、タレントや政治家などの有名人はもちろん、特定の業界の実績者や有名人は、「この人は、すごい人」という形容がされることがある。「すごい人」などと言われてしまうと、全人的にも優秀な人と思われがちな表現になってしまうが、本研究における信頼の期待のされやすさは、特定の能力の範囲に限定する。もしかしたら実際に会ってみたら、本当に人格的にも能力的にも立派な人であった可能性は否めないが、インターネットからはその判別は不可能のため、本研究においては範囲外とする。

もし、多数の Web サイトから判断可能な信頼への期待のされやすさの情報を寄せ集めることができれば、その人の本質を知ることが可能になると期待できる。一方で、トラストスコアで定義されているカテゴリの範疇を超えて相手に期待することは、非常に不確実な判断をすることとなる。それがうまくいくか行かないかは、相手の性格や能力に依存する。一方で試しにやってみて信頼を産んでいくためには、見込みがありそうな人にチャンスを与えて、実際にどう成長していくかについて結果として得られた信頼を通じて判断していくしかないし、そのようにすべきであるということを本研究の基本的なスタンスとする。いずれにせよ不確実

性のある状態から、特定の能力についての信頼への期待のされやすさを抱き、期待をした結果、期待通り成功した場合は信用を得ることができる。その信用を下地として新たな期待を抱くサイクルを描くことが重要である。

3.7. アイデンティティの本人の特定可能性を公表することと信頼の関係性

情報を発信するアイデンティティ自身が本名や所属企業等を公開することなどで、発信者本人の特定可能性を公開している場合、信頼への期待のされやすさにとって有益か否かについて議論する。

3.7.1 匿名のアイデンティティ

まず特定可能性を公開しているアイデンティティの対局に位置する、匿名サービスのアカウントについて検討する。特定のアイデンティティが匿名のまま利用できるサービスにおいては、本人に関する特定可能性は原則として得られない。アクセス元の IP アドレスによってアクセス元の接続している国の判断はできるが、それ以外の情報は詐称が可能である [23]。犯罪を犯した場合のみ、警察からの要請で ISP や Web サービスの運営者がアクセスログを提出することで、本人を探し出すことは可能であるが、それが可能なのは警察から正式な手続きを経た場合の特殊な状況に限られる [24]。このようなアイデンティティについては、発言された内容以外はインターネット側から継続性が判断できない。他の人によるなりすましも許容している可能性もあり、アイデンティティに対する信頼を産むのは現実的に不可能である。

匿名のまま、同一のアイデンティティであるという識別性を確保する取り組みとして、匿名の掲示板である 2ちゃんねるにおいては、匿名のままアイデンティティの識別性を表明するための「トリップ」と呼ばれる仕組みがある [25]。トリップは、発言者が入力したパスワードから生成されたハッシュ値を固定的に画面に表示することで、同一の人物が発信していることを保証できる仕組みである。パスワードを知っている本人のみが、同一のトリップを維持することができるため、パスワードとトリップの生成ロジックが流出しなければ、書き込みに対して継続されたアイデンティティであることは保証される。匿名なので本人の特定可能性は

不確実のままであるが、該当するアイデンティティが女性か否かを判別するために、このトリップが写り込んでいる自身の身体の一部を映した写真をアップロードすることで、女性であることの信用を得る行為も行われるなど利用者間での工夫が行われている。このような状態でアイデンティティの識別性に対する保証が必要になる理由としては、その女性固有の話題を継続的に行っている発信者であることを保証したいということが挙げられる。匿名掲示板においては、情報を発信している個人が特定されないことが魅力とされている Web サービスであるが、その場限りのアイデンティティの識別性を認識させたいシーンが存在している。トリップはその際に活用されている機能である。そのためトリップの利用をやめたり、別のトリップのためにパスワードに変えたりする段階で、それまでトリップに支えられていたアイデンティティは消失する。

3.7.2 Web サービスが本人の特定を確認しているケース

次に、仮名や匿名だったとしても Web サービス自身が利用者本人を特定することで、アイデンティティの識別性を保証しているケースを議論する。Twitter のアカウントには、Twitter 社が本人に電話をかけることで本人の特定を行うことで、同一アカウントであるという識別性の保証を行う認証アカウントというものがある。



図 3.6 Twitter の認証マーク

Twitter の該当アカウントの画面を表示すると認証されていることを示すマークが画面に表示される。これはテレビに出演しているような有名人やインターネットで知名度のあるアカウントにおいて、なりすましを防ぐために行われているものである。Twitter は、アイコンの設定やアカウント名を自由に変更されるために、有名人になりすますことが可能である。そのため影響の大きい有名人や企業アカウントに対して認証マークを提供することで、情報の混乱を防ぐものである。Twitter の利用者は、その人が本当のアカウントであるかを認証マークで確認することで、ツイートが本人によってなされたものかを識別することができる。

一方で、出会い系サイトにおいては、免許証を確認することで一定の本人の特定を確認している。これは「インターネット異性紹介事業を利用して児童を誘引する行為の規制等に関する法律 [26]」にて求められている本人の年齢確認である。本人確認方法は、免許証等の公的証明書の年齢部分をオンラインで送るというものであるが、顔写真や住所、本名等の個人情報にあたる情報は送らない旨のルールになっている。本人確認の目的は年齢が未成年でないことを確認しているもので、本人確認が担保している範囲は非常に限定的である。そのため出会い系サイトを通じて、公開されている写真と違う人と会うことや、風俗業者による偽物のアカウントから未成年の子を紹介されてしまい、意図せず条例違反を犯してし

まうことなどを防ぐことができない。

また、レンタルサーバ事業を行うさくらインターネットでは、登録した利用者に葉書で送った本人確認のためのコードをオンラインで入力することで、住所の到達確認を行っている。住所自体はレンタルオフィスなども指定できるため、厳密に法人の登記簿上の住所や、登録者の特定を確認しているようなものではない。しかし情報の到達性を確認していることで、何か問題があった時の連絡先を予め確認していることになる。

表 3.5 Web サービスによる本人情報の特定の目的

Web サービス	確認内容	目的
Twitter	電話による確認	なりすまし防止
出会い系	免許証の年齢	法規制対策
さくらインターネット	ハガキによる到達確認	連絡可能性の確保

表 3.5 にまとめたように、Web サービス側が本人の特定を確認していても、確認している内容はあくまでも Web サービス側のビジネスの要求レベルによって変わる。それはビジネスの実現やリスク低減を目的としたものであり、必ずしもアイデンティティの信用を高めようとして行っているものではない。その理由として、非対面を前提とした Web サービスにおいては、Web サービスの運営者であっても、アイデンティティ本人の特定を確認するために電話をしたり葉書を送るのは容易ではない上に、利用者に手間をかけさせることになるので、利用者の側にメリットが存在しなければ、利用者の確保ができなくなることから、ビジネスの要求と一致しないためである。クレジットカードなどの本人が特定されている手段を活用することで、結果的に本人特定が行われている状況を作ることはあるが、事業者そのものがそれを行っているわけではない。つまり、不正決済を防ぐためや、法規制に対応するためなどの、ビジネスにおける防御を目的とした本人特定は最低限の範囲に留めておきたいというのが Web サービス運営者の本音であり、そこに本人特定を通じてアイデンティティの信頼を高めるという行為は、多くの Web サービスにおいては積極的に行われていない現状である。

3.7.3 アイデンティティの継続による本人の識別性の確保

Webサービスの多くは、新規登録の容易性を実現するために、IDやメールアドレスおよびパスワードで、アイデンティティの継続を実現するログインアカウントを作成できる。ログインアカウント登録の際に、多くのWebサービスがメールアドレスに認証用URLを記載したメールを送ってメールアドレスの存在確認を行っている。しかし、メールアドレスの到達性は確保しているものの、本人の特定を完全に確認したものではない。このような本人確認は、利用者がパスワードを忘れてしまった時などに、再登録のメールの送り先を確保することで、二度とログインできないような状態を防ぐためのものである。また、登録プロセスを厳密にしているサービスは、認証用URLを表示しなければアカウント登録が完了しないサービスもあるが、認証URLをクリックせずとも、特に制約が一切なく使えるWebサービスも存在する。そのようなWebサービスでは、システムにおいて本人の特定を行う機能は原則的に保有していない。

本来そのような状況下では、アイデンティティ本人の識別性の保証はできないと考えられるが、アイデンティティ自身が発信する情報に継続性と一貫性があることで、本人が特定に繋がる情報を表明していなくても同一人物であると判断することが可能になることがある。例えば、仮名で活動しているソフトウェア開発者がアプリケーションを公開している場合に、成果物の品質に対する継続性と評価を通じて、実質的に同一人物であるという識別性を確保可能なケースである。このようなケースにおいては、発信している情報に対する信頼の期待のされやすさを得ることができる。

その場合に得られている信頼への期待のされやすさは、第3.6項で示したとおり、評価を得られているWebサービスが示す能力の評価カテゴリーの範囲に限られるが、本人の特定をWebサービス側が保証するなどとは無関係に信頼関係を得ることができる。技術的には、ID、パスワードによるログインや公開鍵認証などによってアイデンティティの識別性が確保されているという部分があるにせよ、本研究における信頼への期待のされやすさについては、あくまでも発信されている情報によってもたらされるものである。

更に、その評価から経済的な実利を得られるのであれば、インターネット上のアイデンティティに対する信用が失われることが、当人の社会的評価におけるデメリットに直結し、実生活においても無視できない状態になることも考えられる。

もし、そのような Web サービスで活躍することに十分なメリットがあるのなら、アカウントを継続し、情報発信を継続することでの識別性を維持することが、実質的に本人の社会的にも経済的にも大きな価値をもたらすという逆転現象が起きる。その場合は、むしろインターネット上のアイデンティティの価値を適切に維持することこそが、本人の情報が特定されることよりも重要な価値を帯びてしまうことも起こりうる。

3.7.4 本人の特定可能性を表明するアイデンティティ

特に実名などを表明しなくてもよい Web サービスにおいて、タレントや有名人など、インターネットに本名や所属を明かして活動している人が、オンラインのアイデンティティにおいて、自分しか知り得ない写真のアップロードを行ったり、オフィシャルサイトへ SNS のアカウント名を記載することを通じて、そのアカウントが本人のものであることが明らかにしていることがある。

このように個人の特定に紐づく情報を発信するアイデンティティは、実名を明かすことでファンと交流を図るのに活用している。このようなアイデンティティは、実社会における知名度を利用して、即座に有名人としての信頼を得ることができる。ファンとなる側のアイデンティティも日常生活において出会うことができない人と交流できることから、多数の人が注目し、ちょっとしたアクションに対して、大量の「いいね」や「リツイート」と言った評価情報をつけやすい。そのような状態は、その人個人の評価よりも、タレントとしての知名度の方が圧倒的に大きいいため、何を発言しても「いいね」ボタンが押されるなどの過剰な期待を受けたり、ちょっとした失言が過剰に否定的に取り上げられたりしてしまうなど、Web サービスから得られる評価情報が正しい評価にならないことが特徴である。

3.7.5 アカウントの特定可能性を表明していることと、信頼関係構築の関係性

タレントや有名人に限らずアカウントの特定可能性を表明するために実名や社会的な立場を公表することは、本人がその立場にふさわしい行動をすることへの期待を抱かせる効果が得られる。実名や所属する組織、立場を公表している人物であれば、社会的立場や雇用を失うような不用意な発言をインターネットに向け

で行うのは避けようとするはずだから、その部分に関しては信頼できるという考え方である。

しかし、専門性の範囲を超えて、人格的に望ましい発言をし続けるかというとその限りではない。SNSで実名などの特定可能性を表明していても、不適切な政治信条を語ってトラブルを起こしたり、インターネット上で否定的なコメントを投げかけて、喧嘩を行ったり顰蹙を買うようなできごとは日常的に起きている。

Retty[27]というグルメ情報の掲載サイトでは、実名で飲食店のレビューが投稿されることをコンセプトとしている。実名であれば、いい加減なことは書かないであろうことをシステムが保証するコンセプトである。しかし、実名だからと言って、懇意にしている飲食店を過剰にアピールすることを書かないという保証はないし、なにより味音痴であることや、流行に流されやすくテレビに出たお店ばかりを高く評価してしまうと言ったことは、実名で情報発信することとは無関係かつ無自覚に発生しうる。情報品質を高くするために、誰かにレビューされる可能性があるならともかく、趣味で情報発信している範囲において、実名を伴うことが情報品質に対する高い信頼に結びつくとは考えにくい側面もある。

一方で、フードライターのような仕事で食に関する記事を書いている人が、恣意的な評価を書くことで悪評が立つと、その人の仕事の継続に支障をきたす可能性があるのも、そういう人であれば、本人の名前を公開することで責任編集という側面から情報発信のクオリティは期待できる。実名を表明することは、その人が得意とする専門分野において、責任を持って発言していることに対しては評価可能である。以上のことから、実名であることが信頼への期待のされやすさに影響を与えるのは、本人が専門として社会的に評価されているカテゴリと、発信している情報の評価カテゴリが一致している場合に限るべきである [30]。

しかし、第3.7.3項では、本人の特定可能性を表明せずとも、アカウントを継続的に運用することで信頼への期待のされやすさを確保することは可能であると述べており、特定可能性を表明することは信頼を得るための必要条件ではなく、発信している情報と自身の専門性が一致した場合において、よりスムーズに信頼を得る手段として活用できると考えられる。

3.8. アイデンティティの特定可能性の表明と得られる信頼についてのまとめ

以上の議論から、単純に本名や職業を表明することでの特定可能性が、ネットコミュニケーションにおける信頼のされやすさに繋がるものではないということがわかった。むしろ、2ちゃんねるのような全くの匿名サイトにおいてもトリップ機能を活用することで、その場限りのアイデンティティの識別性を確保してコミュニケーションの成立を目指す動きが存在していることは興味深い。

また、Web サービスが本人の特定を行い、その存在を保証する動きは、信頼の実現を目的としたものではなく、ビジネスリスクを避けるために、最低限の範囲で行っている側面が強い。仮にアイデンティティ本人が詐欺行為などを働いた場合は、利用規約違反のもとでアカウントを止める行為などで安全を確保することはありうるが、決してWeb サービス側からその人の本名を晒して罰したり、ましては当事者間の揉め事の仲裁に入るようなことはない。むしろアイデンティティの継続性を阻止する形でサービスの安全を確保する試みとも言えるため、本研究で想定している信頼の実現に寄与するものではない。

また、本人が特定可能性を表明していることが、発信している情報への信頼のされやすさに寄与するのは、本人の専門性と発信している情報の評価カテゴリが一致している時にスムーズに信頼を得ることができると述べた。

それに対して、本人の特定可能性を表明して信頼を得るという行為を行わずとも、Web サービスの活動として情報発信の継続性と評価を通じて信頼への期待のされやすさを得ることができる。むしろ発信している情報が、本人が表明している専門性に対して的外れであった場合は、個人の能力への信用を失うリスクがあることを考えると、情報発信の継続性と評価を得ることが最も重要な要素であり、特定可能性の表明は評価の得られ易さを高めるための補助的な情報と考えることもできる。

本研究においては、アイデンティティの特定可能性の公表は限定的なメリットしかないと考え、アイデンティティの識別性を確保するためにも必要な情報発信の継続性と、そこに伴って発生する評価情報の組み合わせによって得られる信頼への期待のされやすさに着目し、検討を進めていく。

第4章

トラストスコアの提案

4.1. トラストスコア

4.1.1 指標としてのトラストスコア

これまでの議論から、インターネットで識別可能なアイデンティティが、継続的に情報を発信し続け、それがインターネット上で評価されることで、評価を得ることに必要とされた特定の能力について信頼への期待のされやすさが判断できることを示した。

アイデンティティが発信した情報が全く評価されないのであれば、情報閲覧者が関心を持たなかったものとして、信頼への期待のされやすさには繋がらない。それに対して、炎上と言われる現象のように肯定的、否定的な意見の双方が発生しうる機会は、発信した情報の存在が広く知られるきっかけとして捉え、アイデンティティ自身が、その後も情報発信を継続し続けることが可能であれば、新たな期待を獲得するきっかけになることも示した。

アイデンティティにとって、情報発信の継続性を失うのは、本人がやる気を失ってしまうことや、本人の社会的立場を失うような事態に発展して、情報発信を止まってしまうケースに限られる。つまり、アイデンティティを継続し、情報発信をし続けることが、信頼への期待のされやすさに対する必要条件になる。そして、発信した情報に対して可視化された評価情報は、新たな信頼への期待のされやすさに繋がると考えられる。

インターネットの情報は、アイデンティティが発信したいと思った情報のみが発信されるため、意図的かどうかに関わらず発信されていない情報については判断することができない。そのためインターネットから得られる情報を元にした信頼への期待のされやすさから、アイデンティティに対して得られる信頼には限りがある。それ故に特定の Web サービスに情報発信をしているアイデンティティが

活躍するために必要な能力の範囲については、アイデンティティに対して信頼への期待のされやすさが判断可能であると考えerることによって指標化が可能になると考える。この信頼への期待のされやすさをトラストスコアと定義して、研究を進めることとする。

4.1.2 指標化を行う必要性

Yahoo!オークション [28] やメルカリ [29] などの不特定多数の個人同士が取引を行うオークションやフリーマーケットの Web サービスにおける取引の安全性の実現のためには、まず、取引の代金を Web サービスが一旦預かり取引が完了したらお金が支払われるエスクローシステムによる安心の実現と、それに加えて売り手の評価システムが存在する。過去の取引に伴う評価情報を公開し、新しい取引者が過去の取引実績を元に、信用できそうかについて購入者が判断をしている。特定の Web サービスにおいて、取引の安全性を実現するために必要なルールである。多くのケースにおいては、ていねいな包装や素早い発送を行うようなインセンティブに繋がっているのだが、一方で低い評価をつけられないように慎重にコミュニケーションを取っているケースや、評価の付け方一つで揉めてしまったりと、評価がもたらす影響が無視できないが故に、健全なコミュニケーションが実現できているとは言にくいケースが散見される。このような用途の評価値は、何事もなければ最高点が付与されることが期待されており、なんらかしらの瑕疵が相手の主観による判断を含めて認められた場合に低い点がつけられてしまうという罰則的なスコアリングがなされてしまうのが特徴と言える。これは実質的な減点型の評価手法と言える。このような評価手法は減点されたユーザーを排除する方向に取り扱うことで、プラットフォームによる取引についての安全、安心は提供しているが、相互の信頼を提供しているわけではないことが要因である。山岸らの研究 [11] によると、このような関係は「やくぎ型コミットメント関係」と定義している。やくぎの親分が子分を信頼できるのは、親分を裏切った子分は処分されてしまうことがルールで決まっているからであり、その関係性に根ざしているのがやくぎ的コミットメント関係である。これは一つの Web サービス運営における成功法則としては活用可能であるが、本研究にあるような利用者間の信頼の研究の対象としては不適切である。

本研究では、減点型の評価によるシステムの安全性の実現ではなく、積み上げ型の評価手法による信頼概念の研究を行う。知らない人同士がコミュニケーション

するための判断軸を提供する方法論として、情報発信を積極的に行い、そこから得られる評価を元に、加点方式で信頼への期待のされやすさを推し量るスコアリング手法を検討する。

減点型の評価システムは、事前に最高点を示す想定済みのユースケースが想定されているが、本研究においては Web サービスが示す能力に対する評価を積み上げる方法で算出する。不確実性が高い状況下において信頼への期待を抱けそうな人をあぶり出すというアプローチを取ることで、さまざまな Web サービスに適用可能な方法論として研究し、インターネット全体の信頼性の向上を目指す。

4.2. トラストスコアのスコアリングモデルの検討

指標化を検討するにあたって、先行研究を調査する。特にインターネット上の活動の研究ではないが、特定の分野における活動に対してスコアリングを行う方法として、アメリカのクレジットカードの信用情報を指標化するクレジットスコアのモデルを参考にした。また、ネット上の行動を指標化する試みの先行研究として、ソーシャルネットワークの行動履歴を解析してスコアリングする Klout スコアを研究する。

4.2.1 アメリカの消費者金融における信用判断の検討

アメリカの個人向けの消費者金融には、適切に返済が行えるかについての与信判断を行うための個人信用情報機関が存在している [31]。与信の指標としてクレジットスコアを算出している。クレジットビューロー型とよばれる個人信用情報機関は営利目的の情報機関で、与信に必要な情報を収集し与信業者に毎月の支払履歴をまとめたクレジットレポートの発行と、個人の信用力を審査する際の判断材料となるクレジットスコアの販売を行っている。代表的な信用情報を表 4.1 に示す。

アメリカには、600 以上のクレジットビューローが存在するが、最初に設立されたクレジットビューローは、1841 年に設立された Dun & Bradstreet 社である。米国全土で多様な地域グループに属する顧客を相手に円滑にビジネスを営むためには、迅速かつ適正に信用力を図る基準が必要だった。信用力の基準として、人種や性別にかかわらず同じ基準で公正に与信が行われる必要がある。その制約の中で与信を行うためには、個人の属性よりも支払い能力を重視する傾向にあった。Dun

表 4.1 信用情報機関が収集する個人信用情報

個人の属性情報 (氏名, 生年月日, 住所, 勤務先など)
契約内容に関する情報 (ローンの借入日, 借入金額, 借入残高, クレジット契約日, クレジット契約機額, 支払い回数, 残高, 購入商品など)
延滞などの情報 (発生年月日, 延滞や貸し倒れ, 代位弁済など延滞に関わる内容)
公的情報 (破産宣告・失踪宣告などの公的記録)
手形交換所の第一回目不渡り・取引停止処分

& Bradstreet 社においても、家族や社会的、政治的コネクションなどの所属するコミュニティよりも、経済状況だけでなく、誠実さなどの人格的な特徴から判断される個人の支払い能力を重視していた。

信用情報機関の経済的意義については、Stiglitz と Weiss (1988) が、情報の非対称性が貸し金市場に与える影響という観点から、信用情報の必要性を論じている。与信の際に返済能力が十分ある借り手と、貸し倒れリスクの高い借り手の区別ができなければ、貸し手は平均的な金利を全ての顧客に等しく貸すことで貸し倒れのリスクに備えるしかない。しかし、十分な信用情報を利用できれば、借り手のリスクを正確に測定し、適正なローンの条件を顧客一人一人の返済能力とリスクに応じて設定することができる」と述べている [32]。多様性のあるアメリカにおいて、金融ビジネスが発展するためには不可欠な仕組みであったと言える。

4.2.2 クレジットスコア

クレジットスコアは、クレジットカードやローンの毎月の支払履歴を元に指標化されたものであり個人情報である [33][34]。数社がそれぞれのアルゴリズムによって指標化されており、クレジットカードや住宅ローンの与信等で参照されているものはフェアアイザック社が開発した FICO スコアである。各社の詳細のアルゴリズムは公開されていないが、概略の情報は公開されており、それぞれの点数範囲と、影響を及ぼす要素を以下に示す。

FICO スコア

FICO スコアは、300 から 850 の範囲で決められ、大多数の人は 600~ 800 の範囲に収まる。FICO スコアは以下の要素で決まるとされている。

- 返済履歴
- 借入残高
- クレジット履歴の期間
- 新規のクレジットカードを作ったか

Vantage スコア

Vantage スコアは、FICO スコアと競合するものとして開発された。点数は 500 から 990 の範囲で決められる。Vantage スコアは以下の要素で決まるとされている。

- 支払履歴
- 利用率
- 借入残高
- クレジット履歴の期間
- 新規のクレジットカードを作ったか
- 利用可能金額

FICO スコアと比べて違う要素は、利用率と利用可能金額であるが、利用率とは、現在の借入残高を借入限度額で割ったものを指す。この割合が低いほうが、スコアは上がるとされている。また、利用可能金額とは、利用限度額から現在の残高を引いたものである。

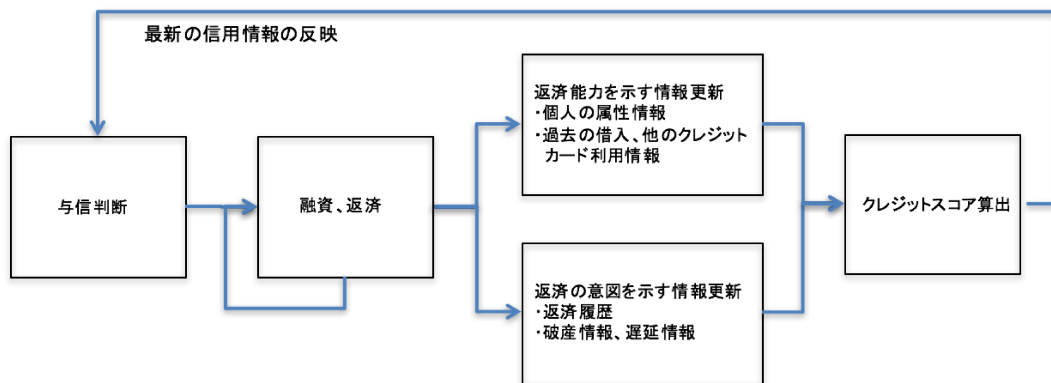


図 4.1 クレジットスコアの算出フロー

4.2.3 クレジットスコアによる信用情報の構造

クレジットスコアリングモデルにおいては、信用力を判断するための個人信用情報として、1. 借入金を返済可能であるかという情報、2. 適切に返済をし続けているかという時間軸に沿った継続的な返済履歴を測定する仕組みを内包している。

図 4.1 に、時間軸に沿ったクレジットスコアの算出フローを示す。融資および返済をしているという事実に対しクレジットスコアを継続的に算出し続けることで、与信判断のための信用情報をループ構造で算出し続けることが可能になっている。もし、個人の資産状況等の属性情報がほぼ一定の状態から、クレジットスコアをあげようとする、適切に既存のクレジットカードを利用し続け、与信枠を上げると同時に、かつ遅滞なく返済し続けることで信用力を増加させていくことができる。

4.2.4 クレジットスコアを参考にしたトラストスコアモデルの検討

クレジットスコアは、クレジットカードの利用実績や借り入れ実績に基き、返済能力という評価カテゴリについてスコアリングされ、その結果として融資額が増えていく。これは行動と評価の組み合わせによって新たな信用創造が生まれていくというわかりやすい例である。アメリカでは、クレジットカードの支払いが定着しているため、クレジットスコアが賃貸の審査や、就職にも用いられたりするなど、クレジットカードの返済能力というカテゴリの枠を超えてスコアリング

そのものが社会の信用力判断に定着している。そのため現金で支払うのではなく、わざわざクレジットカードで支払うようにしたり、どのようにしたらクレジットスコアを上げられるかという書籍が多数出版されていたりと、クレジットカードビジネスが決済のプラットフォームとして特別に定着している様も見受けられる。この背景には、多種多様な経歴を持った移民を前提としているため、経歴による信用判断が簡単に行えないという不確実性が高い状態における信用判断の仕組みとして定着していることが挙げられる。表 4.2 にクレジットスコアとトラストスコアの比較を示す。本研究で述べている、インターネット上で個人を判断しにくい不確実性が高い状態においてアイデンティティに対する信用判断の指標化を行うという部分に関して、クレジットスコアから多いに参考にすべき部分が多い事例である。

表 4.2 クレジットスコアとトラストスコアの比較

要素	クレジットスコア	トラストスコア
信頼範囲	返済能力	特定の Web サイトにおける能力
継続性要素	借入と返済の実施	一定期間以上の情報発信
評価要素	適切な借入と返済の実績	情報発信に対する可視化されている評価

4.2.5 Klout Score によるインターネットユーザの影響力の指標化モデルの検討

ソーシャルネットワークサービス (SNS) に発信されている情報を元にアイデンティティの影響力を測定する Web サービスに Klout という Web サービスがある [36]。Klout は 9 つの SNS の情報を収集し、750 万を超えるユーザをスコアリングし、Klout Score を算出している。スコアリングの方法は、SNS 上の情報発信に対して、フォローなどのユーザがどのように反応したかを測定し、また、発信されている情報を機械学習を用いて、スコアリングおよびカテゴライズを行い影響力を測定する。これによりインターネット上のアイデンティティが、どのような分野において、どれだけの影響力を持っているかを毎日算出している。また、有名人においては、オフラインにおける影響を補完するために Wikipedia に掲載され

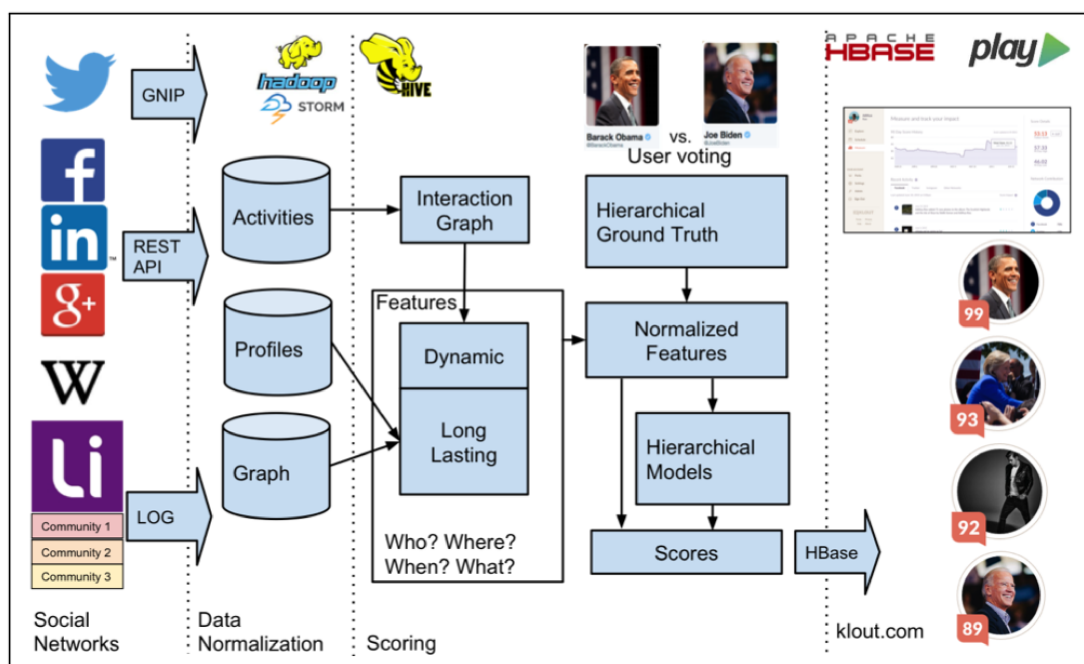
ている情報を活用し、発信している情報のみでは取得しにくい情報を補完する方法でアイデンティティの評価カテゴリを特定し、影響力を算出している。



図 4.2 Klout の Web 画面

4.2.6 Klout Score の指標化モデル

Klout Score の指標化モデルを図 4.3 に示す。各種 SNS から発信された情報に対する反応や、プロフィール、ソーシャルグラフを取得し、データの正規化を行った後に、コメントやシェア等の短期的な反応と、フォロワー数やページランク、もしくは学歴などのプロフィール等の長期的なスパンで変化をする情報に分類を行ってスコアリングを行う。Klout Score における短期的なスパンとは 90 日以下のことを示し、長期的な変動は 90 日以上時間幅にて測定している。また、複数の SNS の情報を一つにまとめてスコアと評価カテゴリを算出できるようになっているのも特徴である。



Klout Score: Measuring Influence Across Multiple Social Networks[36],p3 より引用

図 4.3 Klout Score の指標化モデル

4.2.7 Klout Score のまとめ

Klout Score は、インターネット上の情報発信に対する反応を集約し、スコアリングするという部分で大いに参考になる。特に、短期的な情報と、長期的な情報に分けて測定する部分は、本研究において情報発信の継続性を評価するという部分にも近い考え方である。本研究と相違する部分は、スコアリングの目的が違ふことが挙げられる。Klout Score は、Twitter 等で汎用的な話題を発信しているアイデンティティの影響力を算出し、企業のマーケティングなどに活用しようとするものである。

また、カテゴリの解析においてテキスト情報を解析してラベリングを行っており、多様なカテゴリを分析しようとしている分、精度に問題があるなど信頼判断を目的としたものではなく、カテゴリは影響力に対するラベリングとして存在しておりアプローチが異なる。それ故に、有名人をカテゴリライズするために Wikipedia

の情報を参照する形で、影響力ありきでのスコアリングであると考えられる。

また、Klout Scoreの問題点として、他人の情報に影響された発言を通じてノイズが発生してしまうことにある。図4.2には発言者の発言から特徴的だと思われる情報がタグとして表示されており、必ずしも解析結果が現実的な専門性を示しているわけではない。何故なら、アイデンティティがインターネットに発信している情報は、情報発信者が発信したいと思った情報が発信されることが特徴であるが、インターネット上のニュースなどの興味のある情報をツイートした内容が、必ずしもその人の能力や専門性を示すとは限らないからである。

例えば、今後勉強したいと思っている興味関心のある内容をツイートしたが、他人に多数リツイートされるような話題であれば、Klout Scoreでは、その人の専門性として高く評価されてしまう可能性がある。とりわけTwitterのような自由なWebサービスで発信されている情報から発信者の特性を判断するには、それが本人の情報であるかについて信頼を寄せる判断軸が足りないという問題がある。

4.3. トラストスコアの設計

以上の先行事例の検討および、それまでの議論を元に、トラストスコアの指標化設計について述べる。

4.3.1 トラストスコアの構成要素

トラストスコアを設計するにあたり、信頼可能な範囲を限定的な範囲に留めることが必要である。クレジットスコアは、クレジットカードの返済能力に対する信頼をスコアリングしているものであり、Kloutスコアは影響力のあるカテゴリをテキストマイニングで判別するのに対し、トラストスコアは対象となるWebサービスで活躍するのに必要な能力を評価カテゴリと定義し、そのカテゴリに対する行動履歴から評価カテゴリに対する信頼への期待をスコアリングするものである。もしも特定のWebサービスで活躍していることが社会的に評価可能なものであれば、Webサービスで活躍するアイデンティティも、限られた能力とはいえ社会的に評価される信頼への期待を得ることができる。限られた範囲において「信頼への期待のされやすさ」をトラストスコアとして提供し、その範囲であれば取引やコミュニケーションが成功することが期待できそうという信頼に対する期待値を

提供できるようになる。このようなコミュニケーションの手がかりを元に、情報閲覧者が行動してみた結果、期待通りの成果を得られれば、情報閲覧者の心の中にアイデンティティに対する信用が生まれる。このような成果を期待してトラストスコアを設計する。

4.3.2 トラストスコアの算出モデル

4.1 説に示したクレジットスコアの算出モデルでは、クレジットカードで与信を受けてお金を支払ったり、返済したりを繰り返している行為に評価を行い、資産状況などの情報を加味してスコアリングを行っていた。クレジットカードや融資に関するキャッシュフローを元に試算されたスコアリングが設定され、与信等にフィードバックされる。これはあたかもクレジットカードというエコシステムによるプラットフォーム社会の上に乗っているようなものとも考えられ、利用者は継続的にクレジットカードを活用することで、クレジットスコアを高めていくことができる構造になっている。本研究のトラストスコアにおいても、類するフローを採用する。トラストスコアでは、特定の Web サービスにおいて活躍するのに必要な能力を評価カテゴリと定義し、評価カテゴリに必要な継続的な情報発信と、サービス内で他のアイデンティティから付与される評価情報を元にスコアリングを行う。

4.3.3 トラストスコアの算出モデルの構成要素

図 4.4 に示す通り、トラストスコアの算出モデルは、(1) 情報の取得、(2) 情報発信能力に対する数値測定、(3) 情報に対する評価、反応の測定、(4) アイデンティティの評価カテゴリ分類、(5) トラストスコアの決定 で構成される。各要素の詳細について以下に述べる。

情報の取得

本要素では、アイデンティティが保有する Web サービスのアカウントの情報を取得する。外部の Web サービスからデータを取得する場合は、該当の Web サイトの API を通じて情報を取得したり、ウェブサイトの HTML を直接、取得し、情報を解析するクローリングなどの手段を通じて行う。取得する情報は、情報発信頻

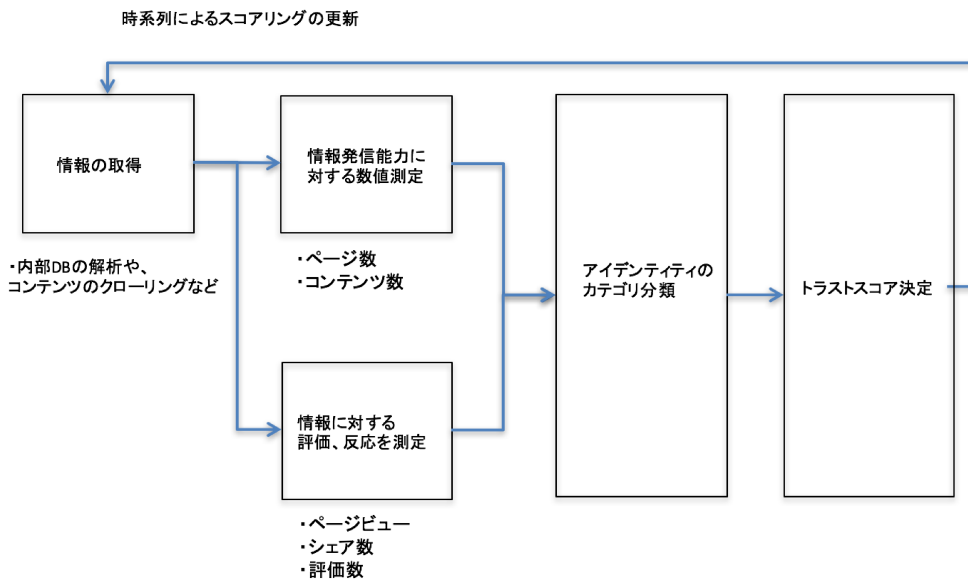


図 4.4 トラストスコアの算出モデル

度を判断するための情報，それに対する評価情報である．ブログサービスのよう
に過去の情報発信情報もクロールできるサイトであれば簡単にデータの取得はで
きるが，一定期間以上前は見るのが難しかったりする場合は，継続的にクロー
ルして情報を取得していく必要がある．

個別の Web サービス毎の情報の取得方法については，提供されている API は Web
サイトの構造に依存するため，アイデンティティが保有するアカウントの Web サ
イト毎に取得手順は異なる．

情報発信頻度のスコアリング

情報発信頻度のスコアリングには，情報発信の期間，頻度を考慮する．Web サ
イトの特性によって発信する情報の発信しやすさは変化し，適切な情報発信頻度
についても Web サイト毎に変化する．

情報発信がしやすいサービスの代表例は Twitter[37] である．情報発信のしやすさ
に特化した情報設計が特徴で，tweet，もしくはつぶやきと呼ばれる発信可能な情

報は、他人にあまり気を使わずとも独り言としても発信することができるようになっており、大量のツイートを送信しているアイデンティティも少なくない。しかし、あまりにも頻繁にツイートを発信すると、フォローのタイムラインを1人の書き込みで埋め尽くしてしまうため敬遠されることもしばしば見られる現象である。その場合は、情報の受信者がフォローを外すという行為で購読対象から外すことができってしまうため、フォローが外されることを嫌うのであれば、情報発信量を自然に抑制させることになる。情報発信頻度はWebサービスのコミュニティ上で作られている人間関係を良好に保つための影響を受けるため、発信可能な情報量はサービスの設計によって制約が存在する。

グルメ情報の食べログ [38] にお店についてのレビューを投稿する場合は、200文字以上書いてあることで優遇されている。以前は、情報の質を担保する指標として、200文字以上のレビューのみが投稿可能であったが、現在は緩和されて短い文字列でも掲載できるようになった。しかし、一部のページでは表示されないなどの制約がついていて、原則的には200文字以上のレビューが好まれる。食べログは飲食店のレビューのため、外食の頻度が多くても1日3~4回という身体的な制約があるため、情報の発信者が発信可能な情報発信頻度に制約が存在する。

またInstagramであれば、写真とコメントだけで投稿できる。写真のクオリティによって、閲覧者の評価がつくので、高いクオリティの写真や、面白いと思われる写真の撮影機会には限りがあるので、評価をされるための写真撮影機会による制約がつく。

Github[39] においては情報発信者がソフトウェアのソースコードを送るため、情報発信そのものの難易度が高く、情報発信頻度はそれほど高くはならない。またソフトウェアが完成してしまえば、以降はアップデートや周囲からの不具合修正に時間を使うようになると、情報の更新頻度は減ることも考えられる。

Webサービスには、そのサービスの特徴に即し、掲載可能な情報や閲覧者との関係性などによって情報発信頻度や情報量には制約が存在する。

そのため単純な発信数が多いことを評価するのではなく、Webサービスによって適切な発信数や頻度は変化する、また評価のされやすさと適切な情報発信頻度は、Webサービスの特性によって異なる。

情報に対する評価のスコアリング

情報を発信したアイデンティティに対し、情報の閲覧者が付与した評価情報のスコアリングについても、Web サービス毎に評価の意味や重要性が異なるため、Web サービスの特性にあわせた評価方法を設計をしてスコアリングを行う。例えばFacebookのいいねボタンやTwitterのお気に入り等はカジュアルに押される。評価という視点よりも前に、その記事を読んだという返答の役割で担われることもある。一方で、高く評価されている記事には、多くの評価が付与される。高い評価を得られる記事を選ぶという意味では、カジュアルに押される数を考慮する必要がある。また、レシピサイトのCookpad[40]では、つくればという実際にレシピを参考にして作ってみたことを報告するレポート機能が存在する。つくればをレシピ情報への評価情報とみなすと、Facebookのいいねなどと比べると実際に作った感想を書くという障壁の高さがある分、評価情報の価値は高くなり、1評価あたりの重みを高く設定する必要がある。

4.3.4 トラストスコアの決定

情報発信頻度、評価値から、分類されたカテゴリに対してスコアを決定する。本研究においては、0~5までの整数を用いた6段階で表現を行う。

表 4.3 トラストスコアの基準

スコア	概要
5	適切な頻度、期間で情報発信行っている。 過去の実績も十分ある
4	情報発信の頻度は適切で、 過去の実績も十分あり高く評価されている
3	直近の情報発信および過去の実績もほどほどに評価されている。
2	直近の情報発信または過去の実績のどちらかが目立つ人が多い。
1	最低限の情報発信をしている。あまり高い評価は得られていない
0	評価不可

図 4.3 に、トラストスコアの分類を示す。値は1~5の評価と、評価不可を示す0を加えた6段階に正規化して表現する。評価分類について、100段階などで評価

することも可能であるが、信頼への期待のされやすさの表現として、80点と81点の差に特筆すべき差がないことと、評価不可の0を除くと、5段階表現というのはミシュランガイド[41]のホテル評価に見られるようなカジュアルかつ受け止めやすい表現を意識し5つ星の表現を採用する。0については、インターネットに発信されていない情報については評価できないため情報が取得できない場合は0となる。インターネットにおける情報発信からのスコアリングにおいては、マイナスの評価が存在しないことも重要な特徴である。

第5章

Githubの公開情報を用いたトラスト スコア的设计

5.1. Github

Github[39]は、ソフトウェア開発プロジェクトのための共有Webサービスである(図5.1, 図5.2)。Githubは、git[42]と呼ばれるソフトウェアのバージョン管理ツールと連携する。gitはソースコード管理領域であるリポジトリに含まれているディレクトリ、ソースコード、画像ファイルなどのファイルの更新管理を行うことができる。Githubはgitで管理されているリポジトリと連携し、リポジトリの所有者であるアイデンティティがインターネット上にソースコードを公開することで、ソースコードを他人に共有することができる。ソースコードの閲覧者はリポジトリに保存されているソースコードをダウンロードしたり、git cloneと言うコマンドを通じてリポジトリの内容を自分の環境にコピーし、ソフトウェアライセンスに従って自由にビルドしたり、自分のプロジェクトに再利用するなど活用することができる。

Githubでは、ソースコードの参照者がgitを通じてcloneしたリポジトリは、元のリポジトリと関連付けたままダウンロードすることができるため、ソースコードを修正したり、新たな機能を追加した場合に、修正者はGithubリポジトリにアップロードされた修正内容をプルリクエストという行為を通じて、元のリポジトリの所有者に変更通知を送ることができる。プルリクエストを通じて、元のリポジトリのオーナーに修正コードを取り込んで欲しいという旨を申請し、ソースコードのオーナーが応じることで修正コードは取り込まれる。これにより、ただソフトウェアを利用するだけでなく、そのソフトウェアの発展に貢献することができる。これらの行為がGithubおよびgitシステムを通じて一元的に実現できるようになっているのが最大の特徴である。

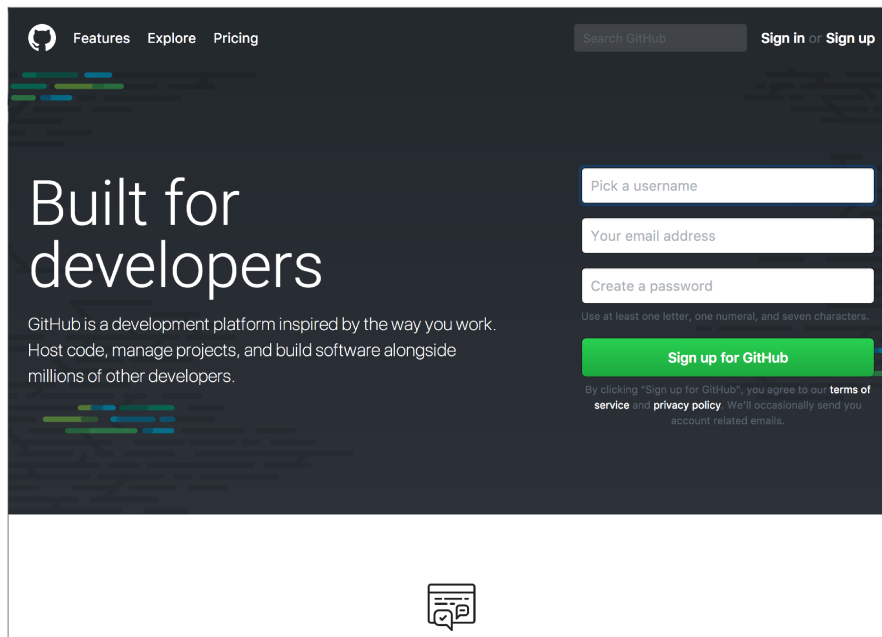


図 5.1 github の Web 画面

Github に登録されているリポジトリは、一般公開することもできるが、非公開のプライベートリポジトリを作り、社内のソフトウェア開発のソースコード管理にも活用することができる。Github の操作スキルを持っていることは、Web サービスを開発するソフトウェア開発者には必須のスキルとなっており、Github を活用した開発プロセスを採用することによる生産性の高さから [43]、Web サービスを開発する企業の開発プロセスにおいては事実上の標準化されたプロセスとして定着している。

5.2. Github を用いる理由

本研究で Github を利用する理由は、まず Github が Web やスマートフォンのアプリの開発者のスキルとして事実上の標準になっていることが挙げられる。全ての開発者が Github で積極的にソースコードを公開しているわけではないが、職場で扱うバージョン管理システムとしては、Github の非公開リポジトリ、Gitlab[47] や、Bitbucket[48] などのソフトウェアや Web サービスなどがあり、これらを通じて git を扱うスキルを持っているエンジニアは増加している。

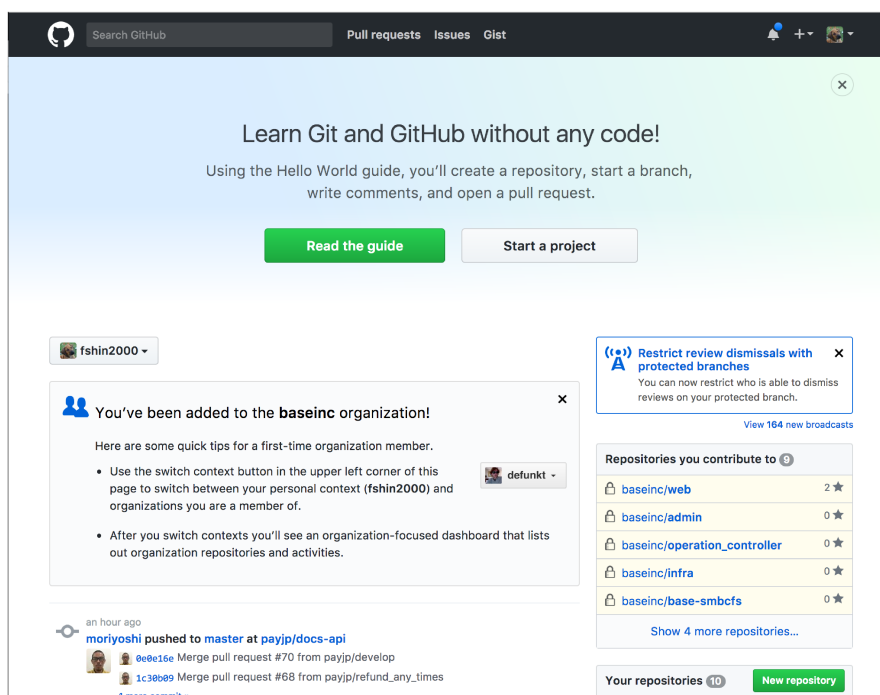


図 5.2 github の Web 画面

そして、ビジネスの採用活動において、エンジニアが転職する際に公開しているソースコードからスキルを判断するのに用いられていたり、開発者自身のプロフィールに Github の URL を掲載するなど、自分自身の経験やスキルをアピールする手段として定着している。しかし、プロフィールに Github の URL を公開しても、全ての人々がスキルを判断することは難しい上に、他のプロジェクトへの貢献や継続的に情報発信していることなどは簡単には把握しきれない。そのようなケースにおいて、トラストスコアを通じてアイデンティティの能力がある程度判別できることはメリットを有む。

Github で多様な人材を見つけることができれば、一言にソフトウェア開発者と言っても、いくつかのタイプのアイデンティティに対して期待を抱くことができるようになる [49]。ソフトウェア開発者として注目されている人は、Web メディアや自分でブログで情報発信したり、イベントに登壇するなど、より注目度を上げていく機会を生かしている。それらの人材は、技術力で活躍するだけでなく、カリスマとして他のエンジニアの吸引力が高いことで人材採用も影響しやすくなる。そのような人材は、他の企業からの誘いの声も多い。

それに対して、これから有名になる人材や、社内では活躍しておりビジネスを支えている人材については、その活躍を知られる機会はなかなかない。もし Github に発信している情報から、ソフトウェア開発者のスキルを素早く知ることができれば、そのような人材を先取りして採用したり、活躍の舞台を先に用意するなど、より肯定的な機会を生む。そのような開発者のスキルの初期スクリーニングを目的として Github のトラストスコアを研究する。

5.3. Github の公開情報におけるエンジニアの信頼への期待のされやすさ

Github のリポジトリは開発者自身や開発者が所属する企業によって管理されており、公開または非公開の設定が可能なりポジトリで構成されている。公開されているリポジトリについては、誰でもアクセスしソースコードをダウンロードしたり閲覧することができる。閲覧者としてのアイデンティティである Github のアカウント保有者は、ソースコードを自身の PC やサーバ領域にコピーできるだけでなく、ブックマーク機能などを通じて公開リポジトリに対して評価情報を表明することができる。本研究においては公開リポジトリの情報を元に、リポジトリのオーナーであるアイデンティティが、ソフトウェア開発者として信頼されやすいアイデンティティであることが計測可能であると考えられる。そのようなアイデンティティは、現状のソフトウェアをより発展させたり、新たなソフトウェアを開発し続けることに対する期待を得ることができる。

また、ソースコードの閲覧者だったアイデンティティが、Github を通じてプルリクエストを送っている人は、自分自身がオリジナルのプロダクトを開発する人ではなくても、修正などを通じて、特定のプロダクトへの貢献を示すことが期待できる。特に Github には Ruby[44] や Python[45], Ruby on Rails[46] などの世界中で使われている著名なプロダクトも公開されている。そのようなプロダクトのコミット権限を持っているコミッタと呼ばれるアイデンティティは、ソフトウェアエンジニアとしても一目置かれている存在となっている。積極的に他のプロジェクトを支えているという視点で、非常に重要な役割であり、チーム開発の有能な構成員という能力においても信頼の期待のされやすさを示す存在と考えられる。

また、他人が公開しているソフトウェアの情報を積極的に集めている人も実社会でも情報通として評価されることが期待される。ソフトウェアの情報をよく知っ

ていることは、仕事における問題解決に繋がったり、ロコミの源泉としてブログなどを通じて情報を円滑な流通を支え、ソフトウェアの認知向上につなげる役割を担うなど、情報通としての信頼への期待のされやすさを示すと考える。

以上をまとめたものを表 5.1 に示す。本研究においては、Github API から取得できる行動情報を元に、オリジナルプロダクトの開発者としての信頼への期待のされやすさを求めることとする。

表 5.1 Github から測定できる信頼への期待感

行動	信頼への期待感
プロダクトを公開	オリジナルプロダクトの開発者として
著名プロダクトのコミッタとして参加	ハイスキルエンジニアとして
様々なプロダクトへのプルリクエスト	チームを支えるコミットメントの高さ
多数のプロダクトを Watch している	情報通, インフルエンサー

5.4. Github で計測可能なユーザアクティビティ

Github のユーザ行動をスコアリングするにあたり、Github API から取得可能な情報およびアクティビティについて述べる。

GithubAPI は、Github が開発者向けに API を提供しているサービスである。図 5.3 に示すような構造で、Github のユーザアカウント、発信した情報、評価情報を取得できる。

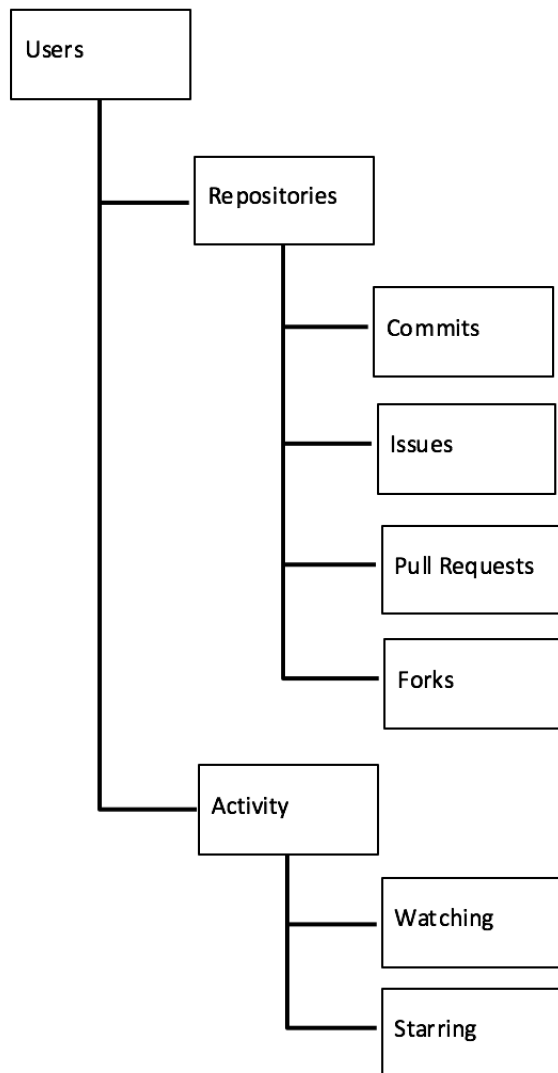


図 5.3 Github API のオブジェクト構造

5.4.1 ユーザ (Users)

本研究におけるアイデンティティと対応する単位がユーザである。ユーザはソフトウェアを公開するためにリポジトリを複数所有することができる。ユーザは、個人で管理することもあれば、企業が所有するアカウントでは複数人で管理も可能である。なお大規模ソフトウェアについては、ユーザがリポジトリを所有するのではなく、ユーザグループである組織 (Organization) の単位で管理されることもあり、この場合はアイデンティティは Organization の 1 ユーザとして所属することとなる。

5.4.2 リポジトリ (Repositories)

リポジトリには、アイデンティティが公開しているソフトウェアプロジェクトの単位となる。リポジトリには、ソースコードの更新情報や、ドキュメント、コラボレーションしている他のアイデンティティとのやりとりなどが全て含まれている。

5.4.3 コミット (Commits)

コミットとは、特定のブランチにおいてソースコードを更新した際のソースコードの変更差分のセットを示す。コミットの履歴を調べると、過去にソースコードにどのような変更を加えてきたかが全てわかる。多くのソフトウェアは、最初は小さく産んで、さまざまな修正を加えたり、新しい機能を追加し成長していく。多くの利用者があるソフトウェアであればあるほど要望等が出て来ることため、それにあわせてコミットの数が増えることになる。コミットをすることが、その製品の最新のソースコードを公開することになり、オンラインに情報発信をしていることを示す。

5.4.4 フォーク (Forks)

フォークとは、他の人が管理しているオリジナルのリポジトリを、自分自身の github ページにプロジェクトまるごとコピーを行い、独自の開発を加えることができる。フォークしたリポジトリは、フォーク元のリポジトリと関連付けが行われており、github の画面を見るとどこからフォークしたのかがわかる。フォークし

たりリポジトリは、独自の拡張をしても良いが、修正した差分情報を、元のリポジトリのオーナーにプルリクエストを通じて提供することもできる。元のプロジェクトが有用で貢献したいが変更点が多かったり、独自の拡張を行いたい時に使う機能である。

5.4.5 プルリクエスト (Pull Requests)

プルリクエストは、誰かが特定のブランチで行った作業が終わった後に、その修正を提出する際に発行されるものである。Git コマンドで更新されたソースコードを取り込む pull コマンドに由来しており、pull を促すという意味のコマンドになっている。プルリクエストを受けたソースコード管理者は、その差分情報を確認し、取り込む意思があれば取り込むし、そうでなければ取り込まないことも可能である。インターネットを介したオープンソースの開発においては、この機能があることで誰でも稼働するソースコードを以って修正提案を行うことができるし、ソースコードの管理側が、差分を取り込む判断をできることから、開発者間で自由な開発を促す重要なコミュニケーション機能として考えられている。

5.4.6 Issues

イシューは、そのプロジェクトにおける課題を管理するための機能である。簡易なタスク管理機能であるが、ソースコードの変更状況と連動して管理することができるため、ソースコード視点のプロジェクト管理機能として使うことができる。プロジェクト毎の運用ルールに依存するが、イシューを活用しているプロジェクトにおいては、活発な課題が出ているか、チーム間で問題の管理が行われているかのバロメータとして測ることができる。

5.4.7 Watch

Watch は、そのリポジトリの更新情報を購読するために使う。このリポジトリページを訪れた閲覧者が Watch ボタンを押すと、リポジトリの更新情報を閲覧者アカウントのダッシュボードに表示することができる機能である。また Chrome ブラウザなどの通知機能を通じて、更新情報を即座に知ることもできる。Watch

に登録するとは、プロジェクトに対する今後の変更について注視しているという表明であり、この数字が大きいことは、多数の人が興味を持っていることを示す。
[50]

5.4.8 Starring

Starring は、そのリポジトリに対してお気に入り登録したものである。Star を登録したアイデンティティは、Github の自分のページから、登録したプロジェクトを簡単に探すことができるブックマークとして機能する。お気に入り登録数が多いプロジェクトは、より注目されているプロジェクトとして話題となることがある。Watch と同様に、プロジェクトに対する人気のバロメーターとして考えることができる。

5.5. Githubにおけるトラストスコアの設計

Github API 取得したアクティビティに基づくトラストスコアの設計について述べる。

5.5.1 Githubにおけるアイデンティティの単位

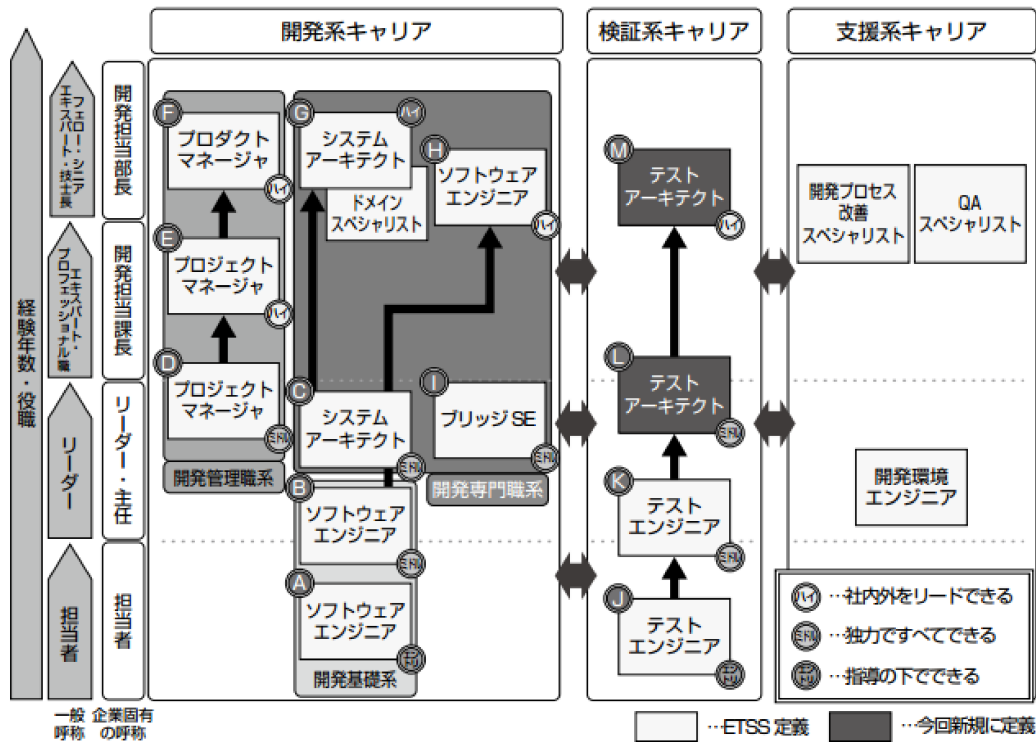
Github における情報発信と評価は、アイデンティティであるユーザが作成したリポジトリの単位で計測する。本研究において調査対象とするユーザは主に個人所有のリポジトリを調査するが、Github 上のリポジトリは Organization というグループが管理しているケースも存在する。例えばプログラミング言語 Ruby のリポジトリは、Ruby コミュニティによって維持されており、Ruby の開発者である松本行広氏の Github アカウントに紐づいているわけではない。本研究においてアイデンティティの単位は、一個人には限らないと定義している通り、Organization も一つのアイデンティティと考えることが可能である。

またアイデンティティは、複数のリポジトリを所有することができる。情報発信の頻度と評価に対するスコアリングは、Github のオブジェクトの構造からリポジトリに付与される。アイデンティティは複数のリポジトリに対するトラストスコアを得ることになるが、リポジトリの中で最も優れたスコアをアイデンティティのトラストスコアとする。アイデンティティの評価をリポジトリへのスコアの平均にしない理由は、それぞれのリポジトリの管理は限られた時間の中で行われていることから、平均を取ると試しに作って見たリポジトリで評価を下げてしまうことになることであり、また合算にしない理由は、合算してしまうと評価されている理由がわかりにくくなるからである。

5.5.2 評価カテゴリの設計

Github の公開情報から信頼に対して期待できる評価カテゴリの設計について述べる。

Github の公開情報から得られる信頼に対する期待として、ソフトウェア開発者の能力の高さが考えられる。しかし、一口にソフトウェア開発者と言っても、多様な役割が考えられる。



組込み技術者向けキャリアガイドの開発 [51],p2 より引用

図 5.4 開発系技術者のキャリア分類

図 5.4 に、「組み込み技術者向けキャリアガイド」 [51] に示されている開発系のキャリアを参考にすると、ソフトウェア開発者は開発専門職系のハイクラスのソフトウェアエンジニア、システムアーキテクト、または開発管理職系としてプロジェクトマネージャ、プロダクトマネージャなどに分かれることが示されている。その中で、Github でオープンソースでソフトウェアを公開するようなソフトウェア開発者は、主に開発専門職のキャリアに進んでいることが多い。Github の公開リポジトリで情報発信を行い高い評価を得ているソフトウェア開発者は、ソースコードを書く能力が高いことから開発専門職としての信頼への期待に対する評価カテゴリとして定義できる。

なお、本研究においては公開リポジトリでソースコードを発信しているアイデンティティを対象にしているが、Github には非公開のプライベートリポジトリが

存在する。こちらは友人とだけ共有するソースコードや、企業アカウントが製品のソースコードを管理する用途で使われているが、プライベートリポジトリへの情報発信のみで活躍しているソフトウェア開発者がいたとすると、パブリックリポジトリで情報発信しているソフトウェア開発者とは違うタイプの開発者である可能性がある。

パブリックリポジトリで活躍しているソフトウェア開発者は、自分で企画し開発しているプロダクトを公開しているが、プライベートリポジトリにおいては必ずしもそうではなくチームの中の一員として活躍している可能性がある。パブリックリポジトリで情報発信しているアイデンティティは、自分のために開発していたり、広くインターネットに向けて発信することを望むタイプで、プライベートリポジトリのみで活躍している場合は、主に携わっている仕事に対しての活躍とみなすことができ、企業にとっては一概にどちらが望ましい人材であるかを比較するのは難しい。

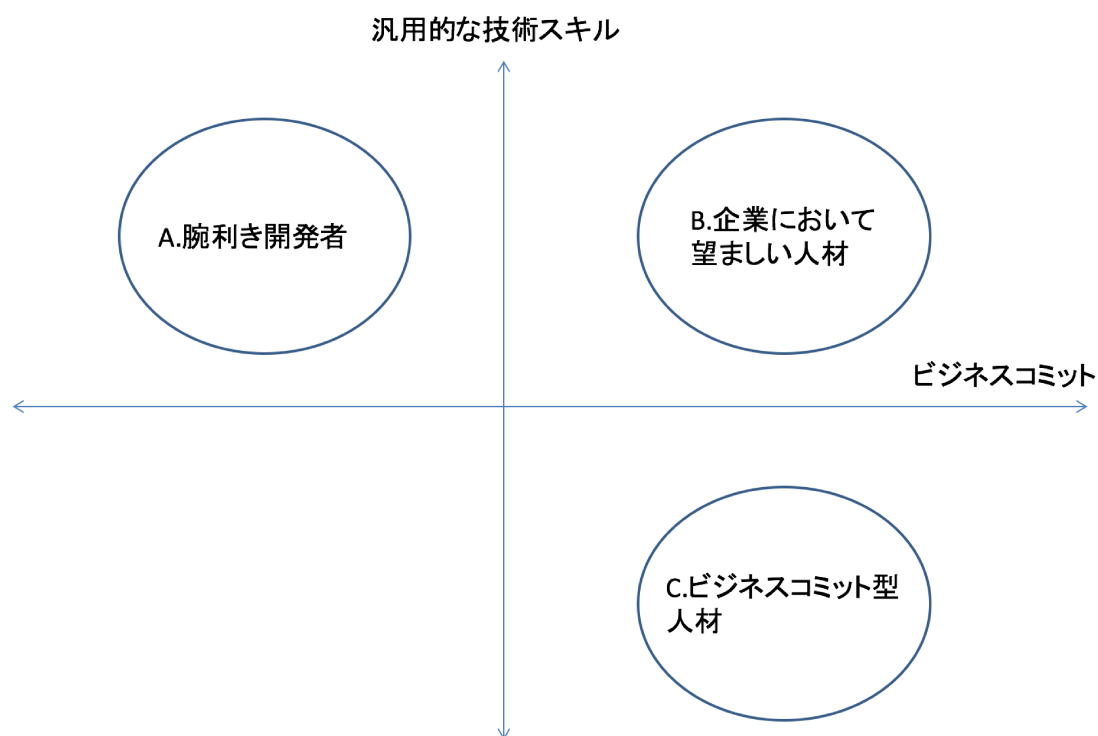


図 5.5 開発系技術者のキャリア特性

図 5.5 に示すインターネット業界のエンジニアの特性を示す。縦軸の汎用的な技術レベルとは、メディアを通じて人気のある開発言語やフレームワーク、ミドルウェアを取り扱うスキルである。横軸のビジネスコミットとは、ソフトウェア開発者として現在勤めている企業のビジネスにとってのコミットメントを示す。A. 腕利き開発者は、Github のパブリックリポジトリを通じてソフトウェアを公開したり、ブログで積極的に情報発信するなど知名度が高い。そのため転職しようと思えば引く手あまたに誘われる人材である。腕利きでかつ、本人の性格が利己的な場合は、会社の仕事はお金を稼ぐ手段であり、自分の興味は自分で管理しているソフトウェア開発であるケースが存在する。また、オープンソースへの貢献に大変興味があり、世界で活躍している開発者も存在する。このような開発者を、自社のビジネスにおいて重要ではあるが単調な仕事を頼むような扱い方をすると転職してしまったりと、ビジネスコミットの面では必ずしも高くはない人材が存在する。もちろん汎用的な技術スキルが高い人材のすべてがそのような人材ではなく、ビジネスコミットと両立できる人材も多々存在しており、それが「B. 企業において望ましい人材」ということになる。そのような人材は非常に貴重な人材であり、技術責任者候補として活躍することになる。その両者は、実際に話をしてみたり、自社のビジネスへの興味を強く持っているかで決まることも多く、判断は難解である。そのためパブリックリポジトリの公開情報からソフトウェア開発者としての能力を信頼して採用しても、必ずしも自社のビジネスとのフィットするとは限らず、期待はずれに終わることもある。

それに対して、C. ビジネスコミット型人材は、ソフトウェア開発者でありながら技術そのものに対して強い興味を持っているとは限らない。そのため積極的な情報交換をしている A. 腕利き開発者と比べるとスキル面や経験面では劣ることが多い。そのため夜や休日を割いてまでパブリックリポジトリを活用した情報発信はあまり行っていない。しかし、ビジネスの実現に対する関心は高いため、仕事についての努力を惜しまない人材であることが期待される。

Github の公開情報によるトラストスコアが高い人材を採用することを考えた場合、開発専門職としてのスキルが高いことは期待できるため、面接においては Github の活躍からは見えない部分、例えば自社のビジネスとの親和性、現状のチームメンバーとの相性などを重点的に確認することに専念することが望まれる。またトラストスコアが高なくても、自社ビジネスとの相性を重視して判断することの検討も必要である。トラストスコアが低い人材では、プライベートリポジトリにおいて活躍する興味関心を持っていれば、採用すべき人材であることも考

えられる。このように公開情報からのトラストスコアから判断可能な範囲には限りがある。

評価カテゴリはソフトウェア開発専門職の能力に対する信頼への期待を示すが、自社のビジネスへのコミットメントがあるか否かは実際に会って確認するしか手がなく、会社員としての能力を Github の情報から判断できるわけではないというのは把握が必要な部分である。

また本研究においては Github の Public リポジトリを元にスコアリングしているが、各企業が保有する Github のプライベートリポジトリを元にトラストスコアを設計、算出できるようになると、ビジネスコミット型人材における技術者の信頼への期待のされやすさをスコアリングすることの期待ができる。現状の転職市場においては、現状の企業内での評価や活躍は面接で本人からしか確認することができず、面接対応の良し悪しに影響されてしまうため、コミュニケーション能力は低い、ビジネスコミットが高く、転職しても活躍可能な人材の判断を見落としてしまうことがある。そのような人材のスキル判断にトラストスコアが適用可能になると活用の幅が広がることが期待できる。

5.5.3 Github から抽出できるスコアリングの設計

本研究では、Github から取得可能なユーザアクティビティを元にスコアリングの設計を行う。Github はアイデンティティであるユーザアカウントに紐づくリポジトリにソフトウェアが公開される。アイデンティティがリポジトリにソースコードをリポジトリにコミットすることで、そのプロジェクトは進行していく。それに対して、リポジトリに興味を持ったソースコードの閲覧者であるアイデンティティが Starring や Watch などの評価機能を通じて期待の意思は可視化され、以降の製品の発展が期待されるようになる。

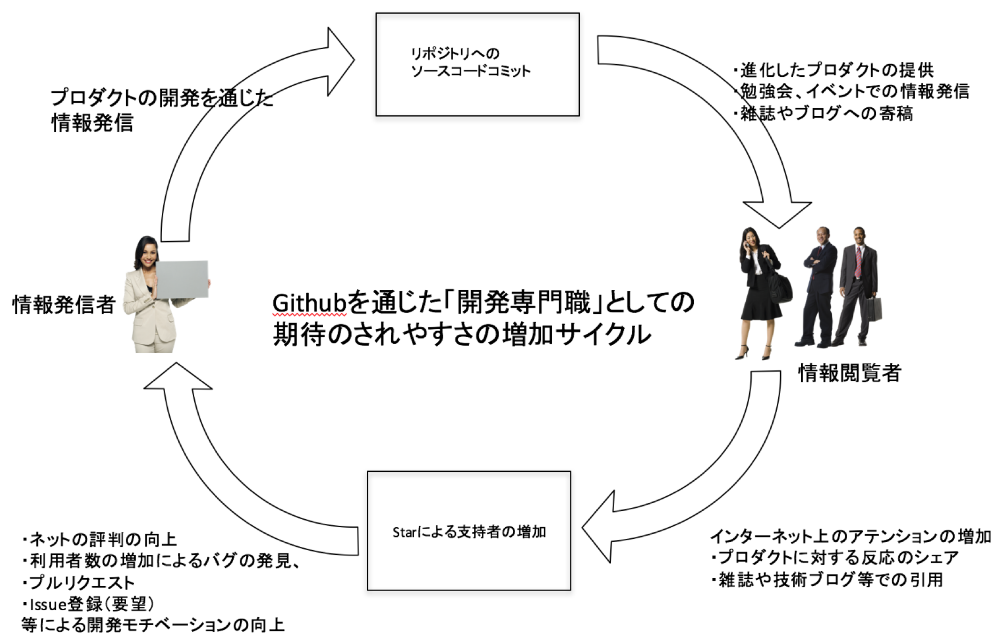


図 5.6 開発専門職におけるトラストスコア向上のサイクル

図 5.6 に示すとおり、アイデンティティである開発者は製品を開発し Github にてソースコードを公開する。公開後に認知向上のために開発者のイベントで発表したり、ブログに書いたり、またはファンとなってくれた利用者が有用性をブログに書いたりなど、製品の話題が口コミで広がっていくことで、新たにこのリポジトリに登録するアイデンティティが増える。注目度が上がることで製品の利用者が増え、注目が上がることで新たに利用者からの要望が出たり、バグ等の改善課題により新たなソフトウェア改善への期待が高まる。アイデンティティは改善への期待にしっかり応えていくことで、製品が発展し続ける。このサイクルを続けると情報発信としてのコミット数が増えていき、それをきっかけとして話題が生まれ、新たに評価するアイデンティティが増えていく。人に評価される製品は、このサイクルを継続し続けるモチベーションを得られることになり、継続的に製品を発展していくことができる。この情報発信のサイクルを続けられる開発者は、Github を通じて開発専門職としての信頼を得ていくこととなる。

5.5.4 トラストスコアの算出式

Githubにおけるトラストスコアの算出式について述べる。

リポジトリ毎のコミットの時期と数による情報を定期的な情報発信と定義し、リポジトリへの評価を算出する。情報発信およびアイデンティティへの評価情報については、2つの時間軸を考慮する。一つは情報発信を開始してからの蓄積された情報発信頻度および評価、また、短期的な一定範囲の期間についての情報発信の頻度および評価を考慮する。

長期的な情報発信量スコア Vlg 、長期的なアイデンティティへの評価スコア Elg 、短期的な情報発信量スコア Vst 、短期的なアイデンティティへの評価スコア Est とした時のトラストスコア Ts は、以下のように算出する。

$$Ts = \frac{Vlg + Elg + Vst + Est}{20}$$

各パラメータは以下の範囲の値を取る

$$0 \leq Ts \leq 5$$

$$0 \leq Vlg \leq 25$$

$$0 \leq Elg \leq 25$$

$$0 \leq Vst \leq 25$$

$$0 \leq Est \leq 25$$

各スコア Vlg, Elg, Vst, Es は25点を満点になるように正規化し、合計100点になる値を20で割り、四捨五入したものをトラストスコアとした。値の範囲は、0を含む1~5までの6段階評価になる。

以下の表5.2に、各指標大小の組み合わせによる想定されるスコアの範囲と、想定するユーザタイプを示す。

トラストスコアは、(1)過去の情報発信頻度、(2)アイデンティティへの評価、(3)直近の情報発信、(4)直近の評価、全てのスコアが高い場合に最大のスコアになるように設計する。これは過去の実績があるアイデンティティが高い評価になりすぎて、新進気鋭の若いアイデンティティが低評価になるのを避けるためである。過去の実績は高いが直近では活躍していないアイデンティティの評価が相対的に下がることとなり、新進気鋭のアイデンティティと同じく、トラストスコア2か3に入る可能性が高い。トラストスコア5と4は、長期的実績も短期的実績において

表 5.2 ユーザのタイプとトラストスコア

情報発信		評価		スコア	ユーザタイプ
長期	短期	長期	短期		
H	H	H	H	5	長期も短期も活躍
H	H	H	L	4	長期も短期も活躍
H	H	L	H	4	実績あるが主に短期で活躍
H	H	L	L	2~ 3	発信量が多いが、評価は高くない
H	L	H	H	-	該当なし
H	L	H	L	2~ 3	過去の実績者
H	L	L	H	-	該当なし
H	L	L	L	1~ 2	過去の情報発信者
L	H	H	H	-	該当なし
L	H	H	L	-	該当なし
L	H	H	L	2~ 3	新進気鋭
L	H	L	L	1~ 2	これから活躍？
L	L	H	H	2~ 3	評価が不自然に高い
L	L	H	L	-	該当なし
L	L	L	H	-	該当なし
L	L	L	L	0~ 1	発信評価共に低い

も申し分ない実績を出しているアイデンティティが入る。すなわち、2以上は、なんらかしら評価される要素を含んでいるアイデンティティである可能性が高い。

短期的な評価を算出する期間については、過去12ヶ月の情報発信と評価について算出を行う。Githubはソースコードによる情報発信のため、3ヶ月以内程度の短い期間では実績が出にくいのと、3ヶ月以内の短期で評価における、一瞬の人気の影響が出てしまうことを避ける。また、短期間であればトラストスコアを高くするために評価を自演することも考えられる。12ヶ月という設定は幾分主観的な設定ではあるものの、自作自演で高い評価を得るには、相応に長期間の行動が求められるものであり、本トラストスコアに対して相応のインセンティブがないと想定はしにくいと考える。もし、本スコアの社会的、ビジネス的影響が十分高くなっていれば、12ヶ月もの長期間でスコアを高くするためだけに活動する利用者が出

てくることは想定できるが，その場合は評価軸を増やしたりアルゴリズムを改良することで，次なる対処を考えることで対応が可能である．これは Google 検索エンジンに対する SEO 対策などで見られるいたちごっこの現象 [56] であり，本研究の結果がそのような状況になるのであれば，研究の成果そのものが社会的に価値があるものとして受け入れられたと考えられるため，評価方法をより深掘りをする研究を継続して進めればよいということになる．また，この期間は短ければ短いほど新しいアイデンティティを評価しやすくなるため，どのようにして短い期間でスコアを算出できるかを考えることが今後の改善点となる．

第6章

Githubからトラストスコアの算出

6.1. Githubのデータの取得

Github apiからのデータ収集には、GHTorrent[52]というプロジェクトが提供するデータを利用した。GHTorrentはGithubの全てのデータを収集し、Creative Commonsの非商用利用ライセンスにおいて研究開発用に公開しているプロジェクトである[54]。

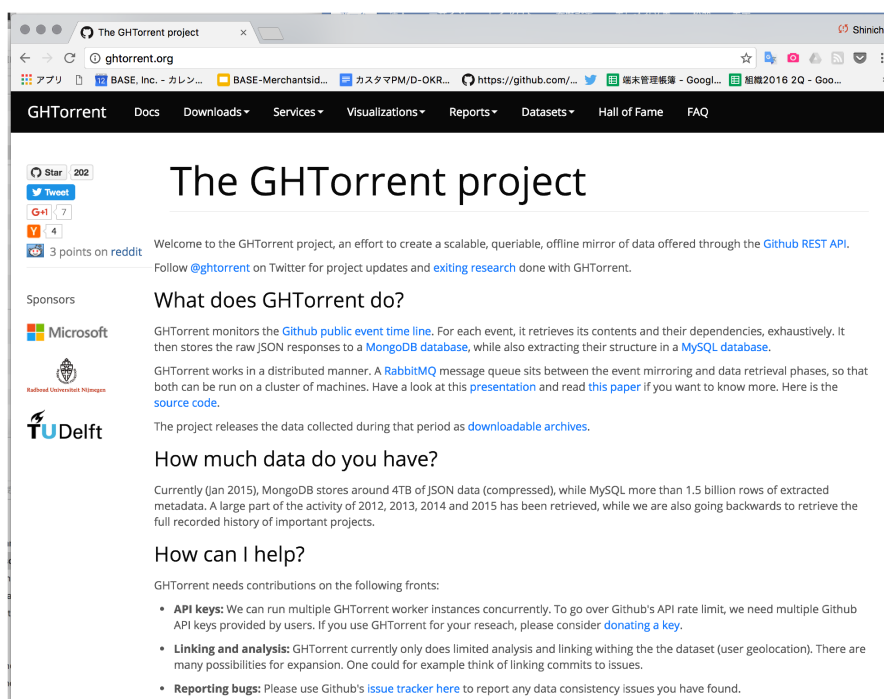


図 6.1 Ghtorrent の Web 画面

6.2. GHTorrent から取得できるデータ

GHTorrent は、Github Rest API の各データおよび、Github から時系列に発生しているイベント情報である Event stream を収集している。Event Stream には、時系列にそった Star などのアクティビティ情報が取得できトラストスコアの算出には不可欠な情報である。GHTorrent では、情報発信や評価に対して、時系列情報に近いデータ取得日時を組み合わせ MySQL[53] で利用可能な dump データが提供されており、再利用が可能になっている。

6.2.1 GHTorrent のデータベース

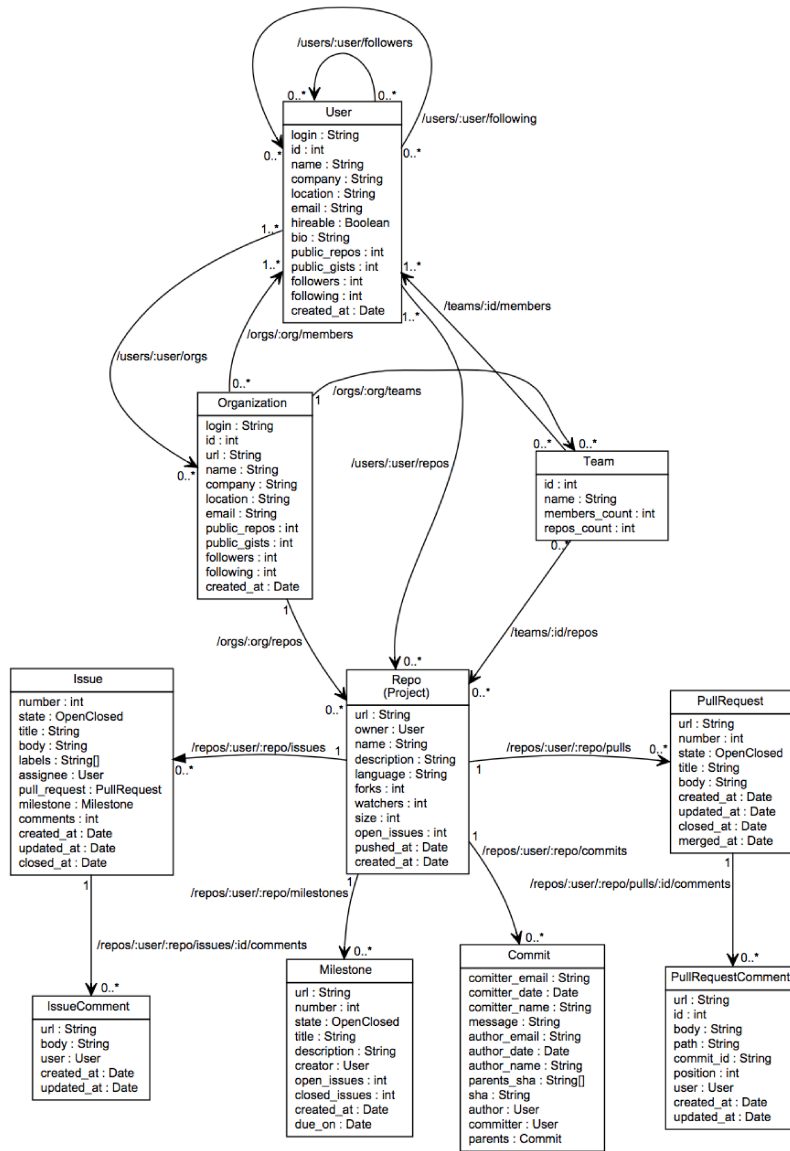
GHTorrent が提供しているデータベースのデータベーススキーマの概略図を図 6.2 に示す [55]。5.4 にトラストスコアの計測に活用する Github API の情報を示しているが、このデータベース構造を活用してトラストスコアを算出する。

図 6.2 に示したスキーマは概略図であるが、以下に詳細情報となるテーブル構造を示す。本研究では、GHTorrent の 2016 年 9 月 5 日に生成された MySQL の dump 情報を利用している。GHTorrent からのダウンロードしたダンプファイルのアーカイブファイルは、tar.gz の形式で、45,186MB である。

上記ファイルを展開して、MySQL データベースにインポートした各テーブルのレコード数およびデータサイズを表 6.1 に示す。

6.2.2 日本人ユーザの特定

アイデンティティ数を示す users テーブルは 14,345,804 アカウント存在するが、検証性の容易さから国単位で解析し、日本の Github ユーザを研究する。表 6.2 に国別のユーザ数のトップ 10 を示す。本情報は GHTorrent によって生成されており、各 github ユーザの公開プロフィールの location の情報を OpenStreetMaps のジオコーディング API を通じて国情報を算出している。日本在住でも Location の記述がないユーザは範囲に入らない。また記載文字列の揺れによるジオコーディングの不具合によって多少違うエリアの情報が含まれることもある。本研究においては、country code が JP と認識されているアイデンティティ 28,925 ユーザが所有する、56,617 のリポジトリをスコアリングの対象として研究を進める。



GHTorrent: Github 's Data from a Firehose[55],p13 より引用

図 6.2 GHTorrent のデータベーススキーマの概略図

表 6.1 GHTorrent が提供するテーブル一覧

Name	役割	Rows	Data length(MB)
Users	ユーザ情報	14,345,804	658.7
Projects	プロジェクト=リポジトリ	39,543,127	4905.5
repo_labels	リポジトリの名称	105,884,216	2484.1
project_commits	リポジトリへのコミット	4,747,346,073	40746.8
project_languages	リポジトリの言語設定	71,597,714	1706.7
project_members	リポジトリのメンバー一覧	7,736,279	270.4
Commits	コミット情報	500,517,968	32457.8
commit_comments	コミット時のコメント	3,418,218	477.2
commit_parents	コミット関連性情報	503,473,135	4321.3
Watchers	Star 情報	54,628,569	677.3
Followers	ユーザへのフォロー	11,616,754	144.0
Issues	イシュー情報	36,573,091	1046.4
issue_comments	イシューのコメント	62,320,790	1664.1
issue_events	イシューのイベント定義	52,911,966	2287.3
issue_labels	イシューのラベル	11,949,623	102.6
organization_members	組織メンバ	371,652	4.6
pull_requests	プルリクエスト (PR)	16,782,870	416.1
pull_request_comments	PR コメント	9,996,473	1338.7
pull_request_commits	PR コミット情報	84,067,780	721.6
pull_request_history	PR 履歴	39,932,014	1072.1

表 6.2 国別ユーザ数 上位 10 カ国

country code	国名	アイデンティティ数
us	アメリカ	384,414
gb	イギリス	76,979
cn	中国	74,338
in	インド	72,257
de	ドイツ	60,505
br	ブラジル	47,218
ca	カナダ	46,974
fr	フランス	44,133
ru	ロシア	32,796
jp	日本	28,925

6.3. コミット履歴を活用した情報発信量スコアの基礎評価

コミット履歴による情報発信頻度評価方法を検討するために、リポジトリ毎のコミット総数の分布を調査した。表 6.3 に、2016 年 9 月末時点で JP と判別されたユーザが所有するリポジトリのコミット数合計の分布を示す。

表 6.3 JP ユーザのコミット数別分類

コミット数	リポジトリ数	比率
0~ 9	28,655	50.6%
10~ 19	8,943	15.8%
20~ 29	4,604	8.1%
30~ 39	2,905	5.1%
40~ 49	1,933	3.4%
50~ 59	1,427	2.5%
60~ 69	1,106	2.0%
70~ 79	847	1.5%
80~ 89	687	1.2%
90~ 99	545	1.0%
100 以上	4,965	8.8%
合計	56,617	100.0%

この分布から求めるコミット数のボリュームゾーンは、コミット数 0-3 のリポジトリが 30%、コミット数が 100 以上のリポジトリは 8.8% である。約 91% のリポジトリがコミット数 100 以下で分布している。コミット数の多寡は、プロダクトの内容や、リポジトリの存続期間によっても、最大値に対する難易度は変わるので、個別の値をリポジトリ毎に比較することには有用性はない。また、多すぎるコミット数を評価することが目的ではないため、コミット数 100 を最大評価のラインとし、それ以上の値は最大値として丸めて扱うこととする。

この評価を元に、長期的な指標と、短期的な指標を計算するために、本表における分類を基礎評価と定義し、4 刻みで分解した 25 段階を用いて、コミット数に対する評価を算出する。

6.4. 月次コミット総数に関するスコアの算出

月次コミットの総数に関する指標化は、そのリポジトリの情報発信の頻度に対して行う評価である。リポジトリ毎の情報発信の合計コミット数を、表 6.3 で示す 25 段階の基礎評価を用いてスコアリングを行う。

6.4.1 コミット数による長期的な情報発信量スコアの算出

特定のリポジトリについての長期的な情報発信量に対するスコア Vlg は、以下の式で算出する。

$$Vlg(m) = \log_{10} \sum_{m=lm-sm}^{lm} cc(m) * 12.5$$

$$0 \leq Vlg \leq 25$$

lm : スコア測定月

sm : 本リポジトリの開始月

$cc(m)$: 該当月の合計コミット数

6.4.2 コミット数による短期的な情報発信量スコアの算出

短期的な月次コミット総数のスコアリングについては、測定月から過去 12ヶ月の範囲で計算を行う。過去の情報であるほど数値を減衰させて直近の情報発信量が重視されるようにスコアリングする。

特定のリポジトリについての短期的なコミット数に対するスコア Vst は、以下の式で算出される。

$$Vst(m) = \log_{10} \sum_{m=lm-12}^{lm} cc(m) * wt(m) * 12.5$$

$$0 \leq Vst \leq 25$$

lm : スコア測定月

$cc(m)$: 該当月のコミット数

$wt(m)$: 重み付け

重み付け $wt(m)$ によって、過去の月のコミット数を低く評価する

$$wt(m) = 1.0 - \frac{(12 - m)}{12}$$

6.5. Star 登録数を活用した評価スコアの基礎評価

GHTorrent を用いた評価の算出にあたっては、リポジトリに対する Star 数をアイデンティティへの評価とみなす。リポジトリの評価には、Star 数以外にもリポジトリへのコミットなどを購読する Watch という機能も存在するが、Star はソーシャルブックマークのように他人とも共有されている評価情報とみなせることと、GHTorrent では Watch の履歴が公開されていないので、Star 数のみを利用することとする。

リポジトリに付与されている Star 数による発信情報への評価値をスコアリング方法を検討するために、リポジトリ毎の Star が付与されている分布を調査した。表 6.4 に、2016 年 9 月末時点で JP と判別されたユーザが所有するリポジトリに付与された Star の数の分布を示す。大多数のリポジトリが獲得している Star の数は 100 以下である。ごく一部のリポジトリのみが Star 獲得数が 100 以上の評価を得られていることから、Star 獲得数 100 を最大値の評価として、0 から 25 段階で評価を行う。

6.6. 月次獲得 Star 獲得数によるスコアリング

Star は、そのプロジェクトの過去の Star 獲得の総数に対して行う評価である。リポジトリ毎の情報発信に対する評価としての合計 Star 数を、6.4 に示している 25 段階の基礎評価を元にスコアリングを行う。

表 6.4 JP ユーザの Star 数分布

Star 数	該当リポジトリ数	比率
0~ 9	48,566	85.8%
10~ 19	3,486	6.2%
20~ 29	1,285	2.3%
30~ 39	706	1.3%
40~ 49	452	0.8%
50~ 59	300	0.5%
60~ 69	216	0.4%
70~ 79	200	0.4%
80~ 89	139	0.3%
90~ 99	127	0.2%
100 以上	1,136	2.0%
合計	56,617	100.0%

6.6.1 Star 獲得数による長期的なりポジトリへの評価スコアの算出

特定のリポジトリについての Star 獲得数に基づく長期的な情報発信量に対するスコア Elg は以下の数式で求める。

$$Elg(m) = \log_{10} \sum_{m=lm-sm}^{lm} St(m) * 12.5$$

$$0 \leq Elg \leq 25$$

lm : スコア測定月

sm : 本リポジトリの開始月

$St(m)$: 該当月の Star 獲得数

6.6.2 Star 獲得数による短期的なりポジトリへの評価スコアの算出

短期的な月次の Star 獲得数のスコアリングについては、測定月から過去 12ヶ月の範囲で計算を行う。過去の情報はスコアへの影響を減衰させて直近の評価が重視されるようにスコアリングする。

$$Est(m) = \log_{10} \sum_{m=lm-12}^{lm} St(m) * wt(m) * 12.5$$

$$0 \leq Est \leq 25$$

lm : スコア測定月

$St(m)$: 該当月のスター獲得数

$wt(m)$: 重み付け

重み付け $wt(m)$ によって、過去の月の評価を減衰させる

$$wt(m) = 1.0 - \frac{(12 - m)}{12}$$

6.7. Githubにおけるトラストスコア

6.7.1 日本のアイデンティティにおけるトラストスコア

これまで示した Github のリポジトリへの情報発信頻度としてのコミット数におけるスコアと、リポジトリへの評価である Star 獲得数を、5.5.4 に示す算出式にあてはめてトラストスコアを算出した。各アイデンティティが所有するリポジトリ毎のトラストスコアは最も高いスコアを、そのアイデンティティのトラストスコアとしている。表 6.5 に算出したトラストスコアとアイデンティティの数の分布を示す。また、表 6.6, 表 6.7, 表 6.8, 表 6.9, 表 6.10 に、JP ユーザーのトラストスコア別の代表的な開発言語を示す。

表 6.5 日本における Github のトラストスコア

トラストスコア	アイデンティティ数	比率
5	127	0.63%
4	499	2.47%
3	930	4.60%
2	4544	22.48%
1	7864	38.91%
0	6246	30.91%

表 6.6 トラストスコア 5 のアイデンティティが使用している開発言語

言語名	アイデンティティ数	割合
Java	16	12.6%
JavaScript	16	12.6%
Swift	16	12.6%
Ruby	15	11.8%
C	7	5.5%
C#	7	5.5%
Go	7	5.5%
C++	5	3.9%
VimL	4	3.1%

表 6.7 トラストスコア 4 のアイデンティティが使用している開発言語

言語名	アイデンティティ数	割合
JavaScript	70	14.0%
Ruby	46	9.2%
Java	40	8.0%
Swift	34	6.8%
Python	34	6.8%
Go	26	5.2%
C++	25	5.0%
C	19	3.8%
Objective-C	15	3.0%

表 6.8 トラストスコア 3 のアイデンティティが使用している開発言語

言語名	アイデンティティ数	割合
JavaScript	123	13.2%
Ruby	91	9.8%
Java	68	7.3%
Python	57	6.1%
C++	51	5.5%
Shell	36	3.9%
Go	30	3.2%
C#	29	3.1%
HTML	29	3.1%
Objective-C	28	3.0%
Swift	28	3.0%
Emacs Lisp	28	3.0%

表 6.9 トラストスコア 2 のアイデンティティが使用している開発言語

言語名	アイデンティティ数	割合
JavaScript	478	10.5%
Ruby	332	7.3%
VimL	299	6.6%
HTML	289	6.4%
Java	238	5.2%
Shell	235	5.2%
Python	230	5.1%
C++	203	4.5%
PHP	162	3.6%
CSS	161	3.5%
Emacs Lisp	142	3.1%

表 6.10 トラストスコア 1 のアイデンティティが使用している開発言語

言語名	アイデンティティ数	割合
JavaScript	995	12.7%
Ruby	792	10.1%
Java	610	7.8%
PHP	428	5.4%
Python	411	5.2%
HTML	346	4.4%
C++	335	4.3%
CSS	331	4.2%
Shell	311	4.0%
VimL	301	3.8%
C#	251	3.2%

6.7.2 その他の国におけるトラストスコア

トラストスコアの分布の参考ために同じ計算式で算出した、その他の国におけるトラストスコアとアイデンティティ数の分布を示す。日本在住と識別できる Github のアイデンティティの数は、世界で 10 位の 28,925 アカウントであるが、1 位の米国 (384,414) を表 6.11 に、3 位の中国 (74,338) を 6.14 に、8 位のフランス (44,133) を表 6.12 に、日本からもオフショア先としてよく連携されている 29 位のベトナム (6,577) を表 6.14 にそれぞれ示す。また、比較のために 5 カ国のトラストスコアの分布状況を図 6.3 に示す。

表 6.11 アメリカにおける Github のトラストスコア

トラストスコア	アイデンティティ数	比率
5	1,063	0.43%
4	3,911	1.57%
3	8,530	3.43%
2	50,657	20.37%
1	101,059	40.64%
0	83,439	33.56%

表 6.12 フランスにおける Github のトラストスコア

トラストスコア	アイデンティティ数	比率
5	185	0.64%
4	591	2.05%
3	1,132	3.92%
2	6,159	21.31%
1	11,947	41.34%
0	8,883	30.74%

表 6.13 中国における Github のトラストスコア

トラストスコア	アイデンティティ数	比率
5	407	0.73%
4	1,319	2.38%
3	2,484	4.47%
2	12,071	21.75%
1	21,761	39.20%
0	17,468	31.47%

表 6.14 ベトナムにおける Github のトラストスコア

トラストスコア	アイデンティティ数	比率
5	6	0.14%
4	35	0.81%
3	67	1.55%
2	662	15.34%
1	1,739	40.29%
0	1,807	41.87%

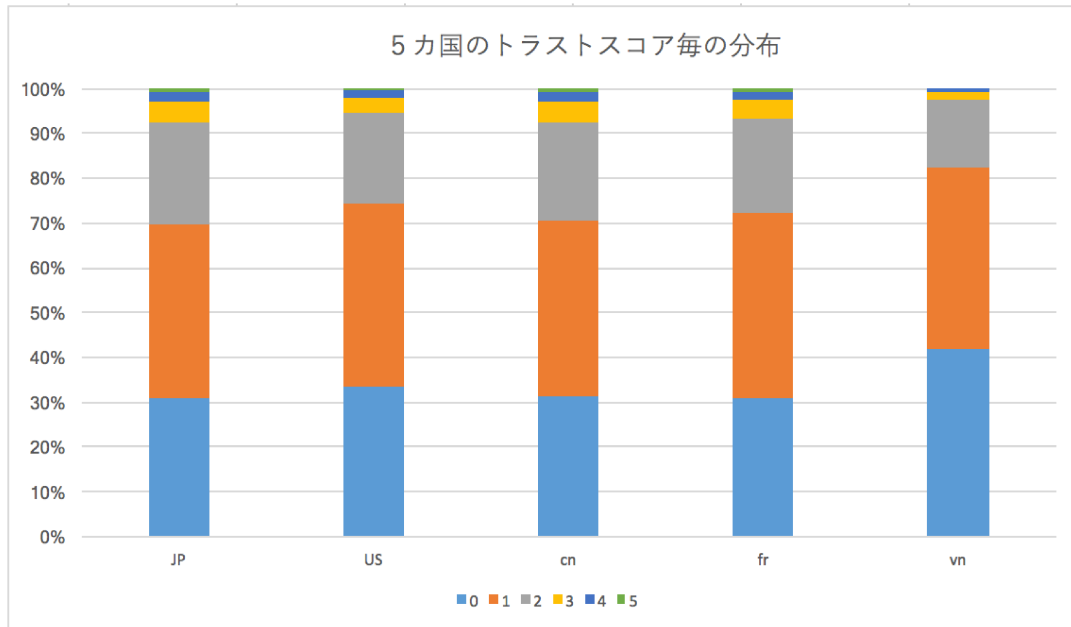


図 6.3 5カ国のトラストスコアの分布

6.8. スコアリングの考察

Github におけるトラストスコアが 5 になる日本人のアイデンティティは、わずか全体の 0.63% という結果になった。過半数のアイデンティティが評価 0 または 1 に該当する。表 6.4 に示した Star 獲得数において、100 以上の支持を獲得してい

るリポジトリは全体の2%程度しかないことから、更に、短期的な時間軸の評価を加えたトラストスコアの最大評価5がわずかしか存在しないのは妥当だと考える。この傾向は日本だけではなく、表6.3に示す世界5カ国のトラストスコアの分布においても、ほぼ同じような傾向になっている。トラストスコア5を獲得する開発者は全体の0.6%以下で、ほとんどのアイデンティティがトラストスコア0または1に属している。このアイデンティティは、Githubを少しだけ使ってみたというアイデンティティもいるが、会社のリポジトリなどの外からは見えないプライベートリポジトリに所属しているケースや、自分ではソフトウェアを公開していないものの他のプロジェクトにpull requestsを送るような貢献をしているケースも存在する。本研究のトラストスコアについては、それらのアイデンティティの計測はしていない。

また、トラストスコアの評価2~4において、30%弱のアイデンティティが存在する。トラストスコア2や3に対する判断の仕方であるが、本トラストスコアのコンセプトは、発信している情報と、その評価という情報でスコアリングをしているため、何かを公開しているという部分を尊重することができると、特に2以上のトラストスコアにおいても相応に活躍していると認めることは可能である。そのようなアイデンティティについては、面接を通じて性格や向上心などを判断することで、別の魅力を見つけることに期待して、一旦、トラストスコアにとらわれずに判断することが望まれる。

第7章

トラストスコアの評価

7.1. トラストスコアの評価

ここまで算出してきた Github を用いたトラストスコアを通じて研究の評価を行った。Github によるトラストスコアは開発専門職という評価カテゴリについてスコアを算出している。開発専門職のトラストスコアの活用例として、インターネットビジネス企業の人材採用における初期スクリーニングを想定しているため、トラストスコアの高いアイデンティティが、採用現場において実際に求められる人材であることが判断できるかを確認することで妥当性の評価が可能である。

評価方法は、現役のインターネットビジネス企業の技術責任者である CTO（最高技術責任者）級人材が、Github のトラストスコアが高い人材、もしくはスコアが低いアイデンティティに対して想定通りの評価を行うかを確認する。インターネットビジネスの最前線にいる企業の現役 CTO クラスの人材がトラストスコアの高いアイデンティティを高く評価するのであれば、トラストスコアを人材採用の初期スクリーニングとして活用することができると考えられる。

まずは予備実験として、数人の技術責任者クラスの人材に口頭でトラストスコアの説明を行い、実際のデータを閲覧しながら、Github を活用したスコアに関するインタビューを行った。すべてのエンジニアが Github を使っているわけではない上に、Github の使い方がバラバラであるという前提を踏まえながら、採用判断における適用可能範囲を確認することが狙いである。

次に予備実験のインタビューから得た知見を元に本実験を行った。Github のトラストスコアの傾向を用いて架空のペルソナを設定し、職務経歴書を作成。ペルソナに対するトラストスコアが想定通りに評価されるかについて、CTO クラスの人材 71 人に協力を得てアンケートおよびインタビューを行った。ペルソナの職務経歴書に対して、実際に面接してみたいか否かという評価を行うことで、トラストスコアの妥当性を分析した。この実験により量的、質的の両面からトラストス

コアの評価を行う。

7.2. トラストスコアのヒアリングによる予備実験

Githubによるトラストスコアの妥当性を検証する。Githubによるトラストスコアは、ソフトウェア開発専門職としての信頼への期待のされやすさを示す。まず予備実験としてWebサービスを運営する企業でCTOなどの立場で実際に人材採用に携わり、人材市場におけるスキルの判断軸や給与相場に実際に影響を与えている人達にヒアリングを行った。

7.2.1 インタビュー概要

表 7.1 インタビュー概要

内容	トラストスコアに関する印象をヒアリング
実施期間	2017/3/1 - 2017/3/2
場所	訪問先会議室
所要時間	1人あたり30分
インタビュー方法	対面におけるヒアリング
トラストスコア算定対象データ	2016/09時点のGithubデータ

7.2.2 インタビュー方法

質問項目は、トラストスコアの画面を共有しながら、以下のような設問をトラストスコアの趣旨や適用範囲については説明しながら感想をお伺いした。

1. 知り合いや会社の同僚を検索してみて、スコアリングをどう思うか。
2. スコアの趣旨を議論しながら、トラストスコアが表現するものについての感想。
3. 総括としての感想。

7.2.3 対象者の特性

インタビューは3組に分けて行った。各社とも採用面接、採用基準、給与水準の決定に携わる責任者クラスのエンジニアである。対象となる人物は、スタートアップと呼ばれる未上場のベンチャーまたは、東証1部上場企業の役員および役員経験者である。Webサービス企業の人材市場においてエンジニア採用の責任を持ち、採用基準、給与相場、人材評価基準の決定等に影響を与えている人達である。すなわち、彼らの判断が、国内のエンジニア採用市場に影響を与えている。そのためGithubを用いたトラストスコアが彼らの判断に沿うようであれば、Webサービス企業のエンジニア採用においては、正しく活用が可能であると考えられる。

- インタビュー1 C社 CIO O氏, CTO Y氏
C社は2014年に東京証券取引所マザーズ市場に上場している上場企業で、O氏は上場時のCTO、Y氏は同社の現CTOとして採用に携わる。
- インタビュー2 M社 プリンシパルエンジニア N氏
M社は、未上場で10億ドル以上の評価額を超える企業を示すユニコーン企業の一社であり、アメリカやイギリスなどにも進出しており優秀なエンジニアの採用に成功していることで業界内でも有名な会社である。その中で、N氏はサービスを維持成長させるコアエンジニアとして働いており、人材採用面接などもこなす。
- インタビュー3 L社 M氏, H氏, K氏, T氏
L社は、CTO経験者による技術者教育に関するコンサルティングを行う会社である。代表のM氏は、東証一部上場企業のM社のCTOを経験し、H氏は同M社の執行役員として技術責任者を経験、K氏は、L社に顧問として携わる傍ら、東証一部上場企業V社の現役CTOであり、T氏は、同社の取締役で、インターネットによるHR系サービス大手のB社の現CPO(Chief Product Officer)であるが、元CTOとして同社エンジニアの評価制度にも携わる。

7.3. インタビュー結果のまとめ

7.3.1 肯定的な感想

- トラストスコアの活用方法.
 - トラストスコアが高いアイデンティティは、有名で注目を集めることができる人だから、このようなエンジニアを採用すると投資対効果が高いと考えられる.
 - エンジニアの知名度やスキルを判断することができない人が使うのに良い.
 - トラストスコアでオフショア先のエンジニアを探す手がかりにはとても有用である.
 - 面接の段階で、スコアが2以上だったら優先的に面接スケジュールを組むなどを人事が判断できるのがよい.
 - 転職エージェントは、スコア2や1から良い人を探すとよい.
 - トラストスコアを活用して書類選考研修を作ることができる. githubのソースコードからスコアを予想することで、ソースコード評価の訓練ができる.
 - 公開しているソースコードの手がかりが無い人も探しているので、スコアリングされている方がよい.
- 上位のトラストスコアのアイデンティティについて.
 - トラストスコア4, 5は一流の開発者ばかりが並んでいる.
 - 上位は、よく知っているオープンソースプロダクトがあって納得感がある.
 - 上位のアイデンティティは採用したいと思うが、自社のサービス側にどれだけ関心があるかが重要で、面接で見なくてはいけないところ.
 - 上位の某氏は、ruby系で有名な会社にコミットしつつ、他社の技術顧問をやっている.
 - 上位は国内外の有名企業の社員が目立つ.

-
- 5の某氏はすごい人だが、本人がそういう役割だからと主張して、CTO職に加えてコードを書くことを重視して活動している。そのことがスコアに現れている。
 - トラストスコア5に、以前、業務委託で機能改善で活躍してくれた人がいる。今は某スタートアップのCTOをやっている。
 - CTOとして活躍してる人は3ぐらいが目立つ。
 - OSSのエンジニアとして有名な方で、3のところで目立つ。
- 低得点のトラストスコアのアイデンティティについて。
 - トラストスコア2の人のソースコードを調べているが、確かに会ってみたいと思う人と、そうでない人が混在している。
 - トラストスコアがついているアイデンティティであれば、小さいベンチャーなら全員会いたいと考える。
 - 某社のCTOが1にいる。仕事以外ではコードを書かないタイプなので納得できる。
 - トラストスコアの低いアイデンティティにも、よいエンジニアがいる。
 - トラストスコアの特性について。
 - HRの分野には大きくわけて、ハードスキルとソフトスキルというのがあるが、このスコアリングではハードスキルはわかる可能性がある。
 - githubもエンジニア自身の自己広報能力でソフトスキルも入っていると考えられる。
 - 見てて飽きない。
 - 自分のスコアが4だった。満足している。
 - 下位にはそこまで有名ではないが、Perlのcpan authorなど昔から実績がある人がいる。

7.3.2 否定的な感想

- スコアの悪用について。

-
- スターで評価されることがわかったら、スターを相互につける人達が生まれる.
 - スターが 100 や 300 だと記事になったり、瞬発力でつくことがありばらつきが多い.
- スコアの特性について.
 - I 氏, H 氏が 1 になっている. ネットで更新しなくなった人達の評価はわからない.
 - 上位の開発者は国際的な評価を受けている人は評価を受けやすそうな印象を受ける. 人気者が高評価になる印象.
 - 体感としては, 3 と 4 の差が少なく, 1 と 2 の差が大きい印象を受ける.
 - 注目を集めることと, 能力を高めることとは別のスキル. ソフトスキル. それだけのために評価をするのは危険ではないだろうか.
 - 50 人以上, ソースコードを見たが, 5 と 4 と 3 のコードのクオリティは曖昧に感じる.
 - ruby コミッタとしても事業にもコミットしてる優れたエンジニアを知っているが, トラストスコアは 2. スタッフとして Ruby の CI 環境を整えたり, コミッタ会議の事務局などをやっており, こういう人はスコアは低い..
 - フルタイム ruby コミッタで 3 の人がいて違和感. オーガニゼーションのコントリビュートも評価した方がいい.
 - いつも忙しい LINE 社のエンジニアの Y 氏のスコアが低い. 100%これだけで採用をやろうとするのは難しい.
 - スコアの算出について.
 - .dotfiles (vim の設定ファイルなどの共有) は除外した方がいい.
 - スターは外国人に見つかる多数ついたり, ブログの影響でノイズ的なスターが付与される恐れがある.
 - 暇になったエンジニアがランキングが上がる予想ができる.

-
- ソースコードを見る限り 1.5 の評価が欲しい。1.5 以上は面接したいと考える。

7.3.3 その他の感想

- Github の背景について。
 - 自社では社内制度でスターがついている OSS にコントリビュートしたら支援金を出す制度があり評価として重要視している。
 - 応募があると社内のメンバーで、twitter アカウントや github アカウント探して評価している。
 - github で公開されているソースコードを確認することで技術的に問題ないというスクリーニングを実際に行っている。
- ソースコードを読み込んで静的解析をやるとより高度な解析ができる。
- スマートコントラクトが発展し、仕事の取引が可視化されると、より高精度の評価ができると考える。

7.3.4 各自の感想

- O 氏
採用面では workable みたいな採用支援ツールが、関連情報を集めてスコアリングしてくれると便利だと考える一方で、github だけでも推し量れないし、個人リポジトリでも不足している。フォークしてるところのアクティビティなども見えることが望ましい。
- Y 氏
自分が知っている領域であれば、技術者の判断が可能であるが、自分が直接知らない領域をスコアリングできているのは価値がある。現状のスコアにおいても、4,5 のアイデンティティでも知らないものがある。更に何故評価されているのかという理由が見えるとよりよい。また、スコアがないことがネガティブではなく、会って話してみようと言う使い方にすることが重要である。

- M氏

トラストスコアを5に丸めたのは良い。3,4,5の差は少ない。2と1の間が広いという印象である。将来はトラストスコアと連動した人材レビューツールなどがあると面接の管理に使いそう。また、ベトナムのオフショアにおいて活用できると考える。

- H氏

開発者の一定ラインのスクリーニングをするのに向いている。重要な人材が来た時に見逃さないでであるとか、全く異なる文化圏の技術者を見逃さないようにするのに活用できる。上位の人材に違和感はないが、有名人が上位に出てしまう傾向は否めない。今よりもソフトウェアエンジニアの裾野が広がった時に、判断の一助として活用できる。

- K氏

エンジニアが使うツールではないという印象を持った。エンジニア以外の人があるような情報を収集する時に活用できる。エンジニアはこういうスコアではなく、直接リポジトリのソースコードを見る。トラストスコア5や4はよい開発者であるが、自分でソースコードを見れば判断可能である。

- T氏

もう一つ共通のデータが一つあれば、非常によくなりそう。HRの世界でのスペシャリスト採用の領域で活用できると考える。

7.3.5 予備実験のインタビューの考察

インタビューの結果から、以下の点に着目した。

1. トラストスコア1にも関わらず活躍している人は、Githubを重視していない可能性が高い
2. 上位は納得感がある。トラストスコアの上位の知人、同様に活躍している人を知っている
3. 上位は採用したいと思うが、これだけでは判断できない
4. 知らない人を判断するのに役に立つ

現状の人材市場において、トラストスコア下位のアイデンティティが示す Github を活用していない開発者にも有能なエンジニアは存在するため、トラストスコアが低いからと言って影響を受けてはいけないという指摘と、上位のトラストスコアについては、有能なエンジニアが実際にいるため加点要素として評価できるという指摘をうけた。また、トラストスコアが高いからと言っても、自社のビジネスやチームとのマッチングや、専門性において必ずしも望ましい人材とは限らないため、実際に会って判断することも不可欠であるという指摘を受けた。トラストスコアは信頼への期待のされやすさを示すという定義をしており、インターネットにおける情報判断の不確実性の高い状況における初期スクリーニングに活用することを意図しているため、トラストスコアが高かろうが低かろうが、採用面接では必ず会って判断しなくてはいけないのは想定通りである。つまり、トラストスコアが高いアイデンティティについては一定の評価を加点することで直接会ってみたいくなる動機が強くなり、トラストスコアが低い状況では加点できない分、不確実性が高いままになるため判断できない状態が継続することが自然であると考えられる。このことについて改めて実験を用いて検証することができれば、トラストスコアが持つ信頼への期待のされやすさを検証できるのではないかと考える。

表 7.2 に、トラストスコア検証のための仮説を示す。

表 7.2 トラストスコア検証の仮説

トラストスコア	情報判断の不確実性	会ってみる動機
高い	下がる	より会いたくなる
低い	不変	トラストスコアは影響せず

7.4. トラストスコアに関するアンケート調査

7.4.1 アンケート調査概要

表 7.2 に示した仮説を元にトラストスコアに関するアンケート及びインタビューを行った。調査対象は、インターネットビジネスを生業とするスタートアップ企業および大企業でエンジニアの採用に携わる CTO クラスの管理職もしくはエンジニアである。オンラインで人材採用に関するアンケートを行い、その後、Skype、もしくは対面にてアンケートの感想を聞く形でアンケート回答内容の深掘りを行った。

アンケートは簡単な職務経歴書の形で作成し、人材採用面接の書類選考を模擬する形式で、40人分の職務経歴書としてのペルソナを用意して、会ってみたいと思うか否かについて質問を行いトラストスコアによる影響を調べた。前半20人はトラストスコアが記載されていない職務経歴書、後半20人はトラストスコアが記載された職務経歴書である。なお、質問表の全文は巻末付録として記載する。

表 7.3 アンケート概要

内容	CTO クラス人材 50人以上への調査
実施期間	2017/7/1 - 2017/7/31
アンケート	Google フォームによるオンラインアンケート
インタビュー方法	Skype によるリモートインタビュー
インタビュー所要時間	1人あたり 30分

7.4.2 調査票の設計と分析方法

40人の職務経歴書としてペルソナの設計を行っている。高いトラストスコアを得るためには、GithubのPublicリポジトリにソースコードのコミットを長期的に行い、かつ、Starを継続的に取得していることが求められる。そのためにはインターネットでの認知を得られており、ニーズの高いソフトウェアを開発していることが必要である。ペルソナの設計においては、実際のトラストスコア上位のソフトウェアの内容やアイデンティティ本人の特徴を見ながらペルソナの設計を行っている。また、Web系企業やSI系企業、製造業などの所属する業界の違いや、Web系やデータ解析系などの実務経験の有無、多職種からの職種切り替えのタイミングの面接であるかなどを考慮している。これらの情報は多様なペルソナを構成するために活用しているが、それだけでトラストスコアの多寡を説明する要素にはしていない。また、外国籍であるかについても、評価者の反応を見るために加えている。

本調査の分析方法は、Githubのトラストスコアが高いアイデンティティが、面接へのたどり着きやすさに関して相関関係を示すことを分析する。結果としてGithubのトラストスコアがソフトウェア開発者としての信頼への期待のされやすさを示すことを検証する。

7.4.3 ペルソナの特徴

アンケートに用いたペルソナの設計として、トラストスコア無しの前半 20 人を表 7.4 に、トラストスコア有りの後半 20 人を表 7.5 に示す。なお、本アンケートの前提条件として、年齢は 32 歳、希望年収は 620 万円と設定しており、Web 系のベンチャー企業にとっては相応に条件が高い。一般的なエンジニアよりも高い採用難易度を設定することで、よりトラストスコアの裏付けとなるスキル面の判断を意識するように設定した。

表 7.4 ペルソナ前半 20 人

名前	経歴の特徴
1 さん	SI からの転職
2 さん	ネットビジネスからの転職
3 さん	スキル VBA
4 さん	ネット系、DB の経験が少ない
5 さん	プロプログラマーからの転職
6 さん	SES 出身、Android 開発経験
7 さん	SI 出身、サーバ管理
8 さん	Android、休職経験
9 さん	EC コンサル出身
10 さん	職業訓練校卒業、未経験
11 さん	SI 出身、PM
12 さん	自動車メーカー
13 さん	インフラエンジニア
14 さん	ソーシャルゲームディレクター
15 さん	メディアの編集長
16 さん	サポートエンジニア
17 さん	Google のエンジニア
18 さん	未踏経験者
19 さん	オフショア開発者、フィリピン出身
20 さん	Web 系言語処理開発

表 7.5 ペルソナ後半 20 人

名前	経歴の特徴	トラストスコア
21 さん	SI からの転職	2
22 さん	Web 受託からの転職	3
23 さん	営業支援	2
24 さん	広告系開発	3
25 さん	フリーターからの転職	2
26 さん	Windows 開発経験, Python 好き	4
27 さん	大企業 Web 運営	1
28 さん	動画サービス経験, 朝は強くない	5
29 さん	.NET の IT 講師	2
30 さん	ネット系営業職	2
31 さん	Web 受託	4
32 さん	家電の設計職	3
33 さん	SI 出身, ネットワークエンジニア	4
34 さん	EC 事業のディレクター	2
35 さん	教育系 IT 企業	1
36 さん	営業職	3
37 さん	データ解析エンジニア	5
38 さん	携帯キャリア企画	3
39 さん	オフショア, 画像処理開発, タイ人	5
40 さん	フリーランス, 独自で開発言語開発	5

7.4.4 調査結果と仮説についての検討

アンケートの結果

調査は 2017 年 7 月に行い, 結果として 71 人のアンケートの回答を得た. 71 人の内訳は CTO 職が 50 人, CEO 職が 5 人, 技術責任者, VPoE などの役割が 16 人である. また東証一部上場企業を含む上場企業での現役 CTO, もしくは CTO 経験者が 21 人である.

アンケート回答者の属性

分析対象のアンケート回答者の属性について、図 7.1 に採用への携わり方についての分布を、図 7.2 にアンケート回答者の役職の分布を、図 7.3 に、経験年数の分布を示す。

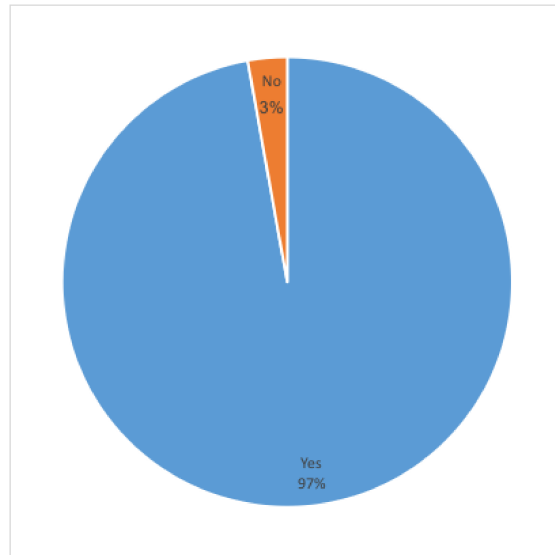


図 7.1 採用に携わっているかについての分布

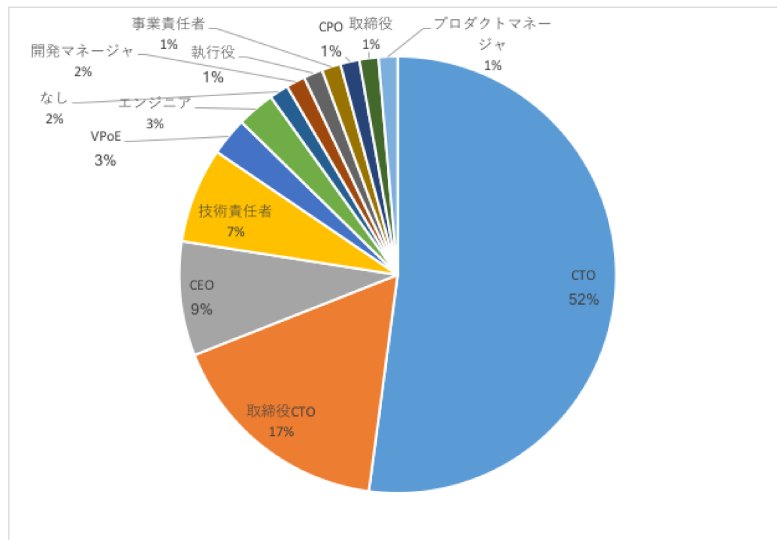


図 7.2 役職の分布

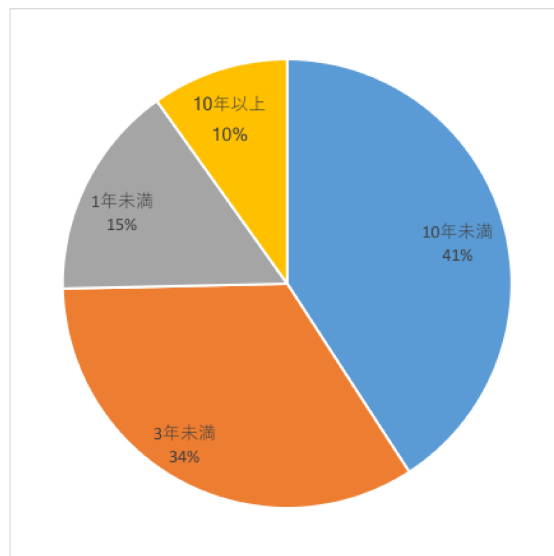


図 7.3 経験年数の分布

本実験のアンケート回答者は、大多数が採用に携わり、経営レベルでの意思決定を行う者で構成されている。採用に携わっていないと回答している者においても、現在、エンジニアの採用コンサルティングの仕事で起業している人や、元ベ

ンチャー企業のCTOであったが現在は大手企業のネットビジネスに携わっている方であったりとエンジニアの人材採用に関する知見や経験を持っている。役職について大多数はCTOもしくは取締役CTOで構成されている。CTO職は最高技術経営者と呼ばれる職業で各社の技術に関する責任者である。データ解析、AIなどの技術の専門企業やインターネットベンチャー企業などの技術を主体としたビジネスモデルの企業に存在する役職である。

ペルソナの回答

アンケート回答を表 7.6 に示す内容でスコアリングを行い、平均点を抽出する。

表 7.6 面接指標

選択肢	スコア
採用を前提に面接してみたい	+2
やや面接してみたい	+1
どちらでも良い（部下に聞いてみる）	0
あまり面接したくない	-1
面接しない	-2

アンケートの選択肢を面接指標として指標化し、アンケート結果から抽出したペルソナの評価の平均点と分散を以下に示す。なお、ペルソナ前半20人のペルソナにはトラストスコアはアンケートでは開示していないが設計上のトラストスコアを表に記載しておく。

表 7.7 ペルソナ前半 20 人

名前	経歴の特徴	想定トラストスコア	平均評価	分散
1 さん	SI からの転職	(1)	-0.12	1.28
2 さん	ネットビジネスからの転職	(3)	1.25	0.78
3 さん	スキル VBA	(1)	-1.49	0.91
4 さん	ネット系, DB の経験が少ない	(1)	0.39	1.29
5 さん	プロプログラマーからの転職	(1)	-1.11	1.11
6 さん	SES 出身, Android 開発経験	(2)	0.79	1.06
7 さん	SI 出身, サーバ管理	(2)	-0.46	1.21
8 さん	Android, 休職経験	(3)	-0.44	1.28
9 さん	EC コンサル出身	(1)	-0.93	1.08
10 さん	職業訓練校卒業, 未経験	(1)	-0.93	1.08
11 さん	SI 出身, PM	(1)	-0.93	1.11
12 さん	自動車メーカー	(3)	0.07	1.32
13 さん	インフラエンジニア	(1)	0.49	1.30
14 さん	ソーシャルゲームディレクター	(1)	-0.44	1.21
15 さん	メディア編集長	(1)	-0.83	1.15
16 さん	サポートエンジニア	(1)	-0.53	1.14
17 さん	Google のエンジニア, Github 未活用	(0)	1.60	0.62
18 さん	未踏経験者	(2)	1.61	0.96
19 さん	オフショア開発者, フィリピン出身	(3)	0.09	1.30
20 さん	Web 系言語処理開発	(4)	1.44	0.82

表 7.8 ペルソナ後半 20 人

名前	経歴の特徴	トラストスコア	平均評価	分散
21 さん	SI からの転職	2	0.02	1.29
22 さん	Web 受託からの転職	3	1.26	0.99
23 さん	営業支援	2	-0.93	1.13
24 さん	広告系開発	3	1.14	0.85
25 さん	フリーターからの転職	2	-0.63	1.14
26 さん	Windows 開発経験, Python 好き	4	0.86	1.11
27 さん	大企業 Web 運営	1	-0.79	1.16
28 さん	動画サービス経験, 朝は強くない	5	-0.39	1.42
29 さん	.NET の IT 講師	2	-1.0	0.84
30 さん	ネット系営業職	2	-1.12	0.90
31 さん	Web 受託	4	0.84	1.16
32 さん	家電の設計職	3	0.12	1.15
33 さん	SI 出身, ネットワークエンジニア	4	0.96	1.19
34 さん	EC 事業のディレクター	2	-1.05	0.94
35 さん	教育系 IT 企業	1	-0.54	1.07
36 さん	営業職	3	-0.54	1.07
37 さん	データ解析エンジニア	5	1.54	0.76
38 さん	携帯キャリア企画	3	0.54	1.13
39 さん	オフショア, 画像処理開発, タイ人	5	0.98	0.97
40 さん	フリーランス, 独自で開発言語開発	5	0.56	1.18

上記結果から、ペルソナ後半 20 人のトラストスコアと面接指標による平均評価との相関関係を表に示す。

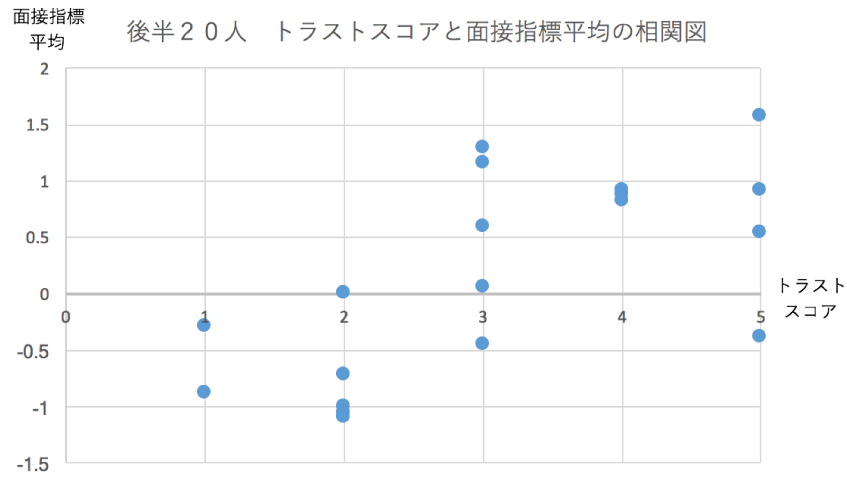


図 7.4 後半 20 人のトラストスコアと面接指標平均の相関図

トラストスコアと平均評価の相関係数は 0.67 となり，トラストスコアの大小と面接指標による平均評価との平均スコアとの間には正の相関が認められた。

なお，参考までにペルソナ前半 20 人の公開していない設計上のトラストスコアと面接指標による平均評価との相関関係を示す。

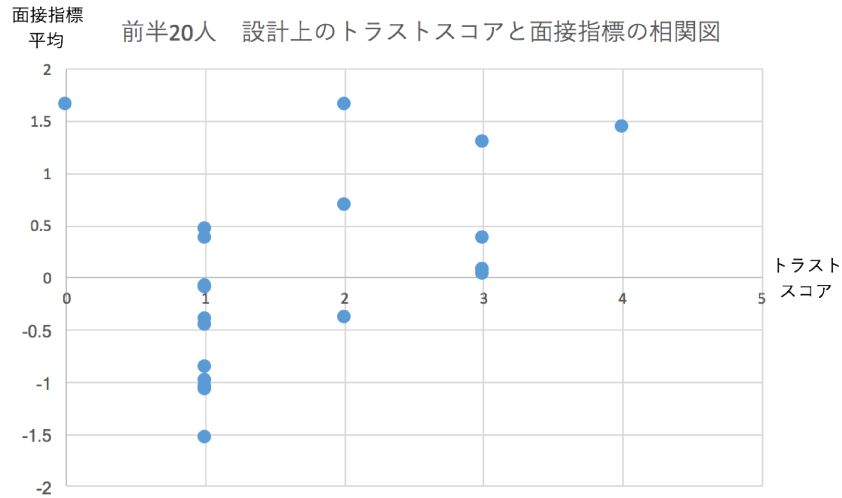


図 7.5 前半 20 人のトラストスコアと面接指標平均の相関図

こちらのペルソナは、Github を活用していてもトラストスコアが高くないケースか、非常に実務経験やスキルが高いものの Github を活用していないペルソナが混在している。そのため設計上のトラストスコアと評価の平均スコアの相関係数は 0.39 となり、後半 20 人よりは相関が低い。

7.4.5 アンケートに関するインタビューのまとめ

アンケートを実施した後にインタビューを行った。表 7.16 に示すインタビュー内容を Skype などの音声チャット、もしくは対面において対話する形で質問した。またインタビューを通じて、人材評価に対する意図などを確認した。

表 7.9 インタビュー内容

質問	内容
1	書類審査する場合に、どういう情報に着目して判断したか？
2	トラストスコアがあることでのイメージに影響しましたか？
3	トラストスコアが高い人は採用したくなりましたか？
4	トラストスコアがあるとネガティブな情報があっても会ってみようかと思うか？
5	トラストスコアが低い人（1， 2）の人はどう思いましたか？
6	海外の人のトラストスコアの有無は判断に影響しましたか？
7	その他、トラストスコアに感じたメリットや感想など

以下にインタビューの結果をまとめを記載する.

表 7.10 Q. 書類審査する場合に、どういう情報に着目して判断したか？

内容	回答人数
実務経験	30
自社技術との親和性	13
今後の成長可能性	12
即戦力と判断できること	9
高い技術力	8
個人でアウトプットしてる質	6
カルチャーフィット	5
チーム開発ができること	4
中堅どころにふさわしいバックグラウンドはあるか	3
Github でアウトプットすることになれているか？	2

表 7.11 Q. トラストスコアがあることでのイメージに影響したか？

内容	回答人数
影響した	32
影響してない	9
良いスコアは加点とする	7
ざっとイメージを掴むのには参考になった（これだけ判断しない）	6
スキル判断で迷った時に活用する	3
低いスコアはマイナス評価にならない	2
低いスコアはマイナス評価になる	1
文章記述の方で判断した	1

表 7.12 Q. トラストスコアが高い人は採用したくなりましたか？

内容	回答人数
高い人はプラスに作用する（会ってみたいくなる）	25
高いからと言って採用しないが会って判断する	12
判断材料の一つとして影響した	11
あまり意識しない	4
人による（他の記述と組み合わせて判断）	4
文章が悪いのにトラストスコアが高い場合は判断に迷う	1

表 7.13 Q. トラストスコアがあると、仮にネガティブな情報があっても、会ってみようかと思うか？

内容	回答人数
記載している内容次第で判断する（スコアよりも優先する）	24
トラストスコアが高いなら会ってみる	15
会ってみてから判断する（スコアはさほど影響しない）	12
会社のフェーズ、状況によって変わる	1

表 7.14 Q. トラストスコアが低い人（1， 2）の人はどう思いましたか？

内容	回答人数
ネガティブに働くことはない（判断できない）	41
良い印象にはならない	9
アウトプットしてるアピールなのに、スコアが低いとマイナス	3
サンプルコードだけ公開してる人はマイナス評価	2
何かに貢献していたり、動くものを自分で作ってれば気にならない	2
伸びそうなら育成とセットで考える	1
アウトプットしているという意味で好評価	1

表 7.15 Q. 海外の人のトラストスコアの有無は判断に影響しましたか？

内容	回答人数
海外の人だからと言って判断には影響しない	34
文化、スキルの判別がしにくいので、客観指標として有益	16
現在は海外の人を採らないので不可	3
オフショア経験者への印象がよくない（自律性、スキル）	3

表 7.16 Q. その他、トラストスコアに感じたメリットなどや感想など

内容	回答人数
客観指標・定量指標なのがよい	19
受験者をポジティブに判断できる要素が増えるのが良い	11
スコアが高い人は評価する	10
スコアが低い人は評価に使わない	6
足切りや候補者のプライオリティ付けなどに有用	6
技術面を見なくても良い、githubを見るべきか否かが判断できる	5
他の人に評価をお願いする時に使える	2
経験上、スコアが高すぎる人が信頼できるわけではない	1
他社では採用されにくいが高スキルの高い逸材を見つけるには有用	1

7.4.6 アンケートおよびインタビューの考察

このアンケートで設定した前提条件の難易度は高く設定されている。まず、年収希望額を620万円を設定しているが、アンケート対象のインターネット系のベンチャー業界においては決して低い給与水準ではない。加えて32歳のエンジニアというのも決して若い方ではないため、チームを技術面またはマネジメント面からリードしていきける即戦力であることが求められる役割として設計した。トラストスコアを前提としたペルソナは、1から5まで分布させることが必要なため、全体的な厳しい結果になることは致し方ないが、その分、面接判断についてしっかり考えさせることを想定して設計した。結果は、トラストスコアの大小とペルソナを元にした評価の平均スコアとの相関は0.67となり正の相関が認められた。

今回のペルソナ設計について、トラストスコア上位のペルソナは、必ずしも充実した職務経歴を書いていない。特にトラストスコア5のペルソナについては、昨今ニーズの高いデータ解析エンジニアが1名いるが、それ以外のペルソナは、フルタイムでチームをリードしていくエンジニアへの期待に対して、健康面に不安のあるペルソナや、全ての企業が採用しているわけではない外国人のエンジニア、また、技術にこだわりが強くチームの一員としては少し難しいと判断されるエンジニアのペルソナを設定している。その結果として、トラストスコア5のアイデンティティの平均評価はあまり高くなく、ばらつきが大きく判断が割れている。そ

の結果として、全体の相関係数が低くなる要因となっている。

インタビューによると、面接の後に最終的に採用をする際には技術力だけで採用しているわけではないことがわかった。採用について最も重視するのは実務経験の有無であり、また自社技術の親和性などのフィット感が重視されている。即戦力としての期待も高く、速やかに現場の組織や開発に合流して欲しいという意図が強い。そのためあくまでも面接で会ってから採用を考えるという回答が多かった。

それでもトラストスコアが高いアイデンティティについては加点評価となり、実際に会ってみたいくなるという意見が非常に多かった。多少人物評価に難しいところがあっても、実績となる技術力が高いのであれば、どうにか採用できる人物であるかを実際に会ってから確認してみたいと思うCTOは決して少なくなかった。高いトラストスコアにおいて、初期スクリーニングの評価に一役買っていることがわかる。

次にトラストスコアの有無が判断に与える影響を考える。アンケートによると外国人のトラストスコアの有無は判断には影響しないという意見が多かった。ネット系の会社は外国人の採用を積極的に行っている会社もあるため、そのような会社であれば特別視しないという意見である。ペルソナにおいて外国人は、19番と39番が該当する。19番はフィリピンのエンジニアでトラストスコアは表記されておらず評価平均は0.28、分散は1.34、設計上のトラストスコアは3である。それに対して39番はタイ出身のエンジニアでトラストスコアは5、評価の平均は0.91、分散は1.05である。プロフィールは違うが、境遇にはさほど大きな違いはない。しかし19番は、前半のトラストスコア3が設定されている他のアイデンティティの評価平均0.44よりも低い。39番は、後半20人のトラストスコア5の平均0.66よりも高い評価を得ている。39番は確かにトラストスコア5に見合うだけの実績を持っているが、平均よりも高い評価である。これだけで判断するのは早計だが、39番はトラストスコア5が判断に与える影響にバイアスをかけていた可能性は否めず、トラストスコアが非公開の19番は平均よりもやや低い評価を得ていた可能性は否めない。こちらについては一つの考察として留めておく。今回は母数が少ないことと、今後トラストスコア自体の認知が高まることでこのような結果を誘発することも想定できる。

表 7.17 外国人の評価比較

内容	18 番	39 番
国籍	フィリピン人	タイ人
プロフィール概略	オフショア開発者	オフショア, 画像処理開発
トラストスコア	3 (非公開)	5
平均評価	0.28	0.91
分散	1.34	1.05

最後に、トラストスコアが低いアイデンティティについては、トラストスコアによる判断をするのではなく、別の要素をしっかりと見たいとする回答が多かった。これはトラストスコアが高いアイデンティティに対する不確実性は減少し、トラストスコアが低いアイデンティティについては、不確実性が高い状況が続くことから、インターネットからの情報だけでは判断ができないという判断を下したと考えられる。これは、表 7.16 に示す仮説と一致する。

第8章

全体考察

8.1. トラストスコア設計に関する考察

本研究の目的は、インターネットという情報の非対称性を前提とした不確実な状況において、ユーザ行動や発信している情報から、その人をどれだけ信用していいのかを判断することで、インターネットをより安心して使えるようにする方法論の研究である。

インターネットの特徴はアイデンティティが発信したいと思った情報のみが発信されることであり、これが情報判断の制約として存在する。アイデンティティの内面にある負の側面を正しく測定することができれば、この人は悪いことをするかもしれないという尺度を元に信頼を測定できるかもしれないが、インターネットの公開情報から判断することは不可能である。

この制約から信頼について考えていくと、2.5節に示した情報の非対称性という不確実な状況において必要とされる信頼の概念に辿り着く。信頼とは、相手が目的とする行為を適切に遂行できそうだという能力もしくは意図に対して期待をし、期待通りの結果になった場合に抱くものである。

この信頼の概念に基いてトラストスコアが目指したものは、インターネット上のコミュニケーションにおいて、コミュニケーションの手がかりとなる「この人なら信頼できそう」という期待の可視化である。これを信頼への期待のされやすさとして指標化することで、人々がコミュニケーションを試したくなる手がかりを提供し、その結果生まれる信頼関係の構築に結びつけることを目指した。

そのための指標としてトラストスコアを算出した。ただし、特定のトラストスコア自体では、判断可能な信頼はその特定された対象に限定される。トラストスコアの算出は、特定のウェブサービスで活躍するのに必要な能力を評価カテゴリと定義し、そのカテゴリに連動する情報発信量と評価情報をシンプルに組み合わせ、評価カテゴリを頼りにしたスコアを算出していることが特徴である。

そして、本研究の検証に用いた Github のトラストスコアでは、リポジトリへのコミットを情報発信とみなし、情報発信に伴う増加する評価を可視化するものとして Starring を評価情報と設定し、長期および短期の時間幅に対してスコアリングを行っている。

このスコアが示すものは、オリジナルソフトウェアの開発専門職としての能力に対する期待のされやすさという極めて限定的な範囲に関して信頼への期待のされやすさの指標である。Github で高い評価を得ているリポジトリのソフトウェアが信頼できることは Web 系の開発者であれば共通認識として識別可能であるが、同じくトラストスコアが高いアイデンティティは、Github への信頼を下地としてソフトウェア開発専門職としての信頼を寄せることが可能になるという考え方となる。

また評価の時間幅について、長期的な評価については、リポジトリ開始時からの蓄積をスコアリングし、短期的な評価については、過去 12ヶ月間の推移を採用し、過去の情報の評価を減衰させて直近の評価の重み付けを増やして算出している。これにより過去に実績はあるものの直近の活躍はしていないアイデンティティよりも、直近で活躍しはじめた新進気鋭のアイデンティティの方が評価が高くなるようにしている。

トラストスコアの特徴はマイナス値が出るようには設計していないことにある。トラストスコアは、インターネットに存在する情報からどれだけ信頼への期待のされやすさを抱けるかについてスコアリングしており、行動の継続性や評価を積み上げていくアプローチとなっている。それはインターネットの情報は不確実性が高いということ前提にしたものであるのに対して、高いスコアであればあるほど不確実性が減っていき、特定の評価カテゴリの範囲に限り信頼への期待のされやすさが向上すると考えられる。

その結果、トラストスコアが高いアイデンティティに対して、その評価カテゴリに関することであれば取引を持ちかけたり、コミュニケーションが成功する期待を抱くことができる。それが初期スクリーニングに活用できる理由である。その期待に対してコミュニケーションを取ってみた結果、思った通りの成果が得られれば、アイデンティティに対する信用が生まれることになる。このアプローチで、インターネット上に情報を発信している見知らぬ人に対して信用判断の手がかりを提供することになる。

トラストスコアは信頼への期待を積み上げていくアプローチのため、トラストスコアの最小値は 0 であり、対象となるアイデンティティの信頼に関して「判断

できない」ということを示す。それに対して、オークションやフリーマーケットの Web サービスの評価システムのような、瑕疵がなければ最大評価を得られ、瑕疵があると減点されることで、相手は信用できないことを識別する減点法とはアプローチが異なる。

また、上位のアイデンティティをひたすら高く評価するのではなく、スコアが高くなりにくい中間層のアイデンティティにおいても、一定の信頼への期待を持てるように条件をつけて設計していることも特徴である。大量の情報発信をしていて、それに伴い評価情報の蓄積も大量になっているアイデンティティを基準にすると、中間層のアイデンティティが評価されにくくなるため、上位の評価には上限をつけて、中間層のあぶり出しができるようにしている。

情報の管理にデータベースを有する Web サービスでは、上位のアイデンティティの活躍を見つけ出すのはデータベースに蓄積された情報をクエリで集計することで簡単に分析できるが、一方で、中間層やこれから出てくる新進気鋭のアイデンティティを見つけ出すのは難しい。トラストスコアについては、できるだけ多くの人が信頼への期待のされやすさを見いだせるようにと、上位のスコアの上限を抑える形で、中間層を見つけやすくしている。

なお、本研究においては周囲の評価情報を適切に判別できないタレントなどの有名人を対象外としている。そのようなアイデンティティは社会的認知の高さから信頼への期待のされやすさが最初から高くなってしまうため、正しい情報発信と評価の組み合わせからは正しく判断できないためである。情報発信の質を問わず評価されやすいという状況を避けるのが目的である。

Github によるトラストスコアの検証で実施したインタビューにおいて、まず予備実験として実際に人材市場に影響を及ぼしている上場企業の最高技術責任者や、人材スキルの基準を作り出しているスタートアップのエースエンジニアにインタビューを行った。彼らの判断が実際に人材市場の序列や業界としての給与水準を作り出している責任者クラスの人材であり、もしトラストスコアが肌感覚として受け入れ可能であれば、実際の業務においても活用可能と考えてインタビューを行った。この予備実験から導き出された仮説として、全体的に不確実性が高い人材判断の中で、トラストスコアが高いことは、アイデンティティへの能力判断に対する不確実性を低くする効果があるのではないかと考えた。つまり、トラストスコアが高いことが信頼への期待のされやすさの影響を与えることで、実際に会ってみるためのハードルが下がるという仮説である。これが検証されることで、Github などのトラストスコアが、実際の行動判断に影響を与えるということを実証でき

ると考えた。

これを検証するための本実験として、50人を超えるCTOクラスの人材に、40人のペルソナに対して書類審査を行うアプローチでアンケートを行った。前半20人にはトラストスコア無しのペルソナを用意し、後半20人にトラストスコア有りのペルソナを用意し、それぞれの書類審査を行う形のアンケートである。その後、skype面談を行い詳しく質問した。この実験からわかったことは、トラストスコアだけで採用判断をしようとは思わない反面、トラストスコアが高い人材については確実に会ってみたいと思う結果となった。トラストスコアのペルソナには、必ずしも会ってみたいと思うとは限らないであろう癖の強い人材なども入れてある。当然、そのような人材への評価は分かれるものの、トラストスコアの低い人材と比べて確実に会ってみたいという評価をいただいた。つまりトラストスコアの裏付けとなるスキルが高いのだから、実際に会ってみて採用可能な人材かどうかを見極めたいという動機が存在している。

またトラストスコアの低い人材に関しては、トラストスコアで判断するのではなく、トラストスコア以外の要素において面接可否について判断したという意見が多数あった。アンケート結果からはトラストスコアが低い場合は、情報判断の不確実性を下げる効果がなかったことを示す。アンケート参加者の期待は、あくまでもトラストスコアに依存するのではなく、自分自身の目で判断したいと言う結果であった。現状の職場の経験からGithubを生かしていない人材でも優秀な人材が存在することを知っていることが、Githubだけに依存したトラストスコアだけで判断しない理由である。

一方で、実験結果の分析によると評価とトラストスコアの間には相関があることから、今後、トラストスコアそのものに対する信頼性が高くなると、トラストスコアが低いアイデンティティに対する判断の仕方も変わってくる可能性がある。更にトラストスコアが普及することで、アイデンティティ自身が、トラストスコアを意識して行動することで、長期的な情報発信と評価を受けられるようなアウトプットを意識した活動をし始めることが期待される。そのような行動変容が起きることで、ソフトウェア開発者の全体的なスキルの底上げも期待できる。

8.2. トラストスコアの貢献と限界

本研究の貢献は、Webサイトにトラストスコアを導入した場合、そのWebサイトで活躍するために必要な能力を評価カテゴリと定義し、情報発信頻度と評価情報からスコアリングをすることで、評価カテゴリに限った情報発信者を信頼する手がかりを示すことにある。その結果、評価カテゴリという限られた範疇であるものの、不確実性の高いインターネットの中でも相手を信頼する機会を生み出し、コミュニケーションを促進させ期待通りの結果になることで、相手に対する信用を生み出すことになる。

そのような積み重ねで、インターネットを活用するユーザーが継続的な情報発信によりアイデンティティを継続し、評価を得るという行為を重要視し始めることで、さらなる信頼性の構築に繋がっていくことができる。

本研究の検証において、Githubによるトラストスコアがソフトウェア開発専門職の能力に対する信頼への期待のされやすさを導き出したのと同じく、違う能力を求めるWebサイトでは、必要な能力を評価カテゴリとしたトラストスコアを前提としたサービス設計にすることで、その道のプロの登竜門にもなれるような存在になることが重要である。トラストスコアの考え方を踏襲した様々な評価カテゴリのWebサイトが成功することで、人々の個性がよりインターネットに投影されやすくなる。その結果、よりインターネットがメリットある形で活用されることが期待できる。

一方でトラストスコアを既存のWebサイトに導入する場合の限界は、まずそのWebサイトで活躍する能力が明確に定義できる必要があることが挙げられる。フリートークを前提としたSNSなどではその能力が社会的に評価されている可能性が低く、そのようなWebサイトではトラストスコアの定義ができない。しかしながら、インスタグラマーやYoutuberのようにそのSNSで活躍すること自体が評価カテゴリとして定義できるならトラストスコアを導入することは可能である。また、その他の限界として、情報の発信頻度や評価情報が一定期間以上測定できることが必要になることが挙げられる。そして、評価情報の存在が適切にユーザの情報発信へのモチベーションに影響しているWebサービスであることもトラストスコア算出の前提条件となる。それはWebサービスを活用するアイデンティティが、サービスで活躍する能力を持ち、継続的に情報発信を行い、それに対して評価を得ているということがトラストスコアのコンセプトになっているからである。

また、昨今、コミュニケーションの敷居が低いWebサービスであればあるほど、

他人と軋轢を起こすとすぐにアカウントを削除して、新たに作り直すアイデンティティが後を絶たないが、そのような Web サービスでは、継続的な情報発信を計測できないため、トラストスコアを導入するのには不適切である。一方、本研究の検証に用いた Github においては、何か問題が起きたらすぐにアカウントを作り直すような機会はあまり見られないため、トラストスコアの適用には有用な Web サービスである。

トラストスコアを通じて、そのような考え方を是正していき、アイデンティティの継続にメリットを提供することで、インターネットでコミュニケーションを行う信頼性を高めていくのがトラストスコアの役割でもある。そのような価値観の醸造が今後、重要な課題となる。

これは Web サイトを新規にプロデュースする方法論の視点においても、トラストスコアを意識したサービス設計にすることで、Github のように多数の利用者を抱えるサービスを作ることができる可能性を示している。今後、トラストスコアを成功する Web サービスのデザインパターンとして確立することが重要である。近い考え方としてゲーミフィケーション [57] と呼ばれる Web サービスやスマートフォンのアプリにおいて、ユーザ行動の結果、アイテムやバッジがもらえるようなモチベーション向上策を通じて継続率を高める設計論が存在するが、特定の Web サービスの優等生となるべきアイデンティティが高いトラストスコアを得るように情報発信と評価を得ることができ、中間層のアイデンティティも相応のトラストスコアを得るようなサービスデザインをすることで、多くのユーザが定着する Web サービスの設計法則を確立することが重要な課題であると考えられる。

8.3. Github のトラストスコアの活用に関する考察

検証で活用した Github のトラストスコアの活用についての考察を述べる。トラストスコアを活用してエンジニアを採用に活用する際に、トラストスコアの上位層だけを狙うとトラストスコア 4 以上で全体の上位 1% 程度の数少ない開発者の取り合いになってしまう。まだ資金力もブランド力も高くないスタートアップ企業などが、このユーザ層に目を向けてしまうのは難易度が高く必ずしも合理的ではない。それよりもトラストスコア 1 や 2 程度のアイデンティティの中から、共に成長できそうな人を見つける方が望ましい。今回アンケートに協力いただいたような CTO クラスの人材が書類選考をせずとも、人事担当が、この人の面接を優遇す

るや否やを判断して、優先的に面接のスケジュールを組むなどの用途が考えられる。優れたエンジニアに限らず人材採用は対応スピードが重要とも言われており、トラストスコアによる判断の有無によって採用の成否にも影響することが期待できる。

8.4. Githubのトラストスコアの活用範囲

近年、オープンソースのソフトウェアを公開したり、開発情報に関するブログ記事などを積極的に発信する開発者を英雄視するメディアが増えたことで、インターネット系の開発者は、休日も勉強して情報発信をしないといけないのであるとか、オープンソースにコントリビュートしなくてはならないのではないのかということについてストレスを感じている開発者が増えている。しかし、実際は仕事が忙しかったり、結婚したり子供が生まれたりすると家族のケアなどに時間を取られ、休日や深夜に活動することが難しい現実がある。このことで自分はダメなエンジニアではないのかという閉塞感や専門職の継続に対する将来への不安を抱えている開発者は少なくない。

本研究において、Githubの公開情報とその評価からトラストスコアを開発することで、そのスコアの限界について述べた。インターネットで公開されている情報だけでは、開発専門職という職業人の良し悪しを判断することはできないことを示した。トラストスコアが付与されているアイデンティティについて、ソフトウェアエンジニアとしての信頼への期待を抱くことはできても、それが自社のビジネスとうまく折り合うかは判断できない。特にインターネットビジネスにおいて、ビジネスの内容や方向性、プロジェクトの進め方について、その人がやりたいと思うか、思わないかについては、スキル以上に影響がある。検証でインタビューを行ったCTOクラスの人達は、公開されている情報だけが全てではないということを実感的に認識しており、検証作業においても本研究におけるトラストスコアの限界として指摘を受けている。

本研究から明確に判明したことは、インターネットに公開されている情報から判断できる範囲には限界があるということである。公開情報だけで開発者の評価をしてはいけないという部分は、もっと知らしめていく必要がある。Webサービスの開発者は日常的にオンラインの情報と接しているため、どうしてもインターネット上の注目度を意識せざるを得ない状況にある。そのため、オンラインで目

立つエンジニアを過度に評価してしまいがちであるが、自身が日常的に携わっている仕事への貢献も同じように評価できないと、特に若年層の開発者であればあるほど視野が偏ってしまうケースがある。今日、インターネット関連の開発者を悩ませているエンジニアの評価の得られ方について、自社のビジネスに集中して貢献することもしっかり評価されることが重要であるということを示し、開発者のキャリアパスが正しい方向に向かうようにする必要がある。一方で、オープンソースライセンスでソフトウェアを公開することは、たくさんの人に自分のソフトウェアを使ってもらい社会を豊かにしていく方法論であり、同時に開発者としてのプレゼンスを向上させることについては、重要な選択肢の一つであるということを示す必要がある。

8.5. 今後の展望

本研究の検証に用いた Github のトラストスコアにおいて、アイデンティティ自身が発信しているパブリックリポジトリへの情報発信を用いてトラストスコアを算出した。オリジナル作品に対する開発能力を示したとも言える。

Github は、今回の評価カテゴリ以外にも活用方法が存在する。最も大きな用途が、Ruby や Ruby on rails などの数多くの人携わっている著名な言語やフレームワークが保存されている Github プロジェクトへの貢献である。このような特定のプロジェクトの貢献は、今回の研究では対象外とした。仮にそれぞれのトラストスコアを算出する場合は、各プロジェクトで重要視されているユースケースを元にスコアの算出方法を設計するべきである。例えば、Github の issues をよく使うプロジェクトや Github wiki, プルリクエストのみを活用するプロジェクトなど、その価値観はプロジェクトによって異なる。特に著名プロダクトであればあるほど、その評価の恩恵を受けるアイデンティティの数は増えていくため、個別に算出する価値は高いものになる。

また、将来的に Github の公開リポジトリからは取得不可能なプライベートリポジトリにおけるトラストスコアが測定できると、各社の開発者が日常的な仕事で携わっている領域に対するパフォーマンスが見えてくる。現実的にはそのような情報は企業秘密であり、仮に実現しようとする測定システムのあり方、トラストスコアの取扱い、セキュリティに対する考え方も含めて、実現すべき課題について難易度は高い。

仮にトラストスコアを公開したり，他社と基準をあわせて共有することができれば，技術者の転職などで行われている，前職の企業に問い合わせ人間性やスキルの確認をするリファレンスを取るような活動において，より効率的な状況が生まれる．仮にエンジニアのパフォーマンス評価サービスとして自動化されていると，自分自身の評価がわかりやすくなることで，より低コストに転職しやすくなるエンジニアが増えることが予想される．

そのようなサービス設計を工夫して，企業内の情報取得を自動化し，エンジニア自身の活動がオープン，クローズ問わずに指標化できると，仮に今の仕事がつまらないと感じても，スコアメイキングを目的にひとまず今の仕事を頑張るという選択肢も期待できる．クレジットスコアが信用のプラットフォームになったように，トラストスコアが特定の能力における信頼のプラットフォームとして活用されることは望ましい未来である．また，企業側においても，自分たちの部下の生産性が把握できるようになることで，パフォーマンスが落ちていて退職懸念があることを検知できれば，面談でケアするようなきっかけを提供するなどの応用例が考えられる．

Githubに限らずトラストスコアそのものの今後の展望は，インターネットに発信されている情報から特定の能力に対する信頼への期待のされやすさの第一歩を作り上げることができれば，Webサービスにおいて，利用者間の信頼について相手を期待する機会はより増えていき，アイデンティティに対する信頼に配慮したインターネットの世界を構築することが期待できる．それによってアイデンティティの継続性を促すことができたり，より円滑なコミュニケーションができるようになることでインターネットの価値はより向上することが期待できる．

また，複数のWebサイト間においてトラストスコアを測定し，単一のアイデンティティの評価をまとめる方法を研究することで，多数のWebサイト間でのトラストスコアを融合することが可能になる．そのようにすることでインターネットから切り取り可能なアイデンティティの特性を広く見つけていくことで，その人の人となりを集めていくこともできると考える．一つのWebサイトでは限られた評価カテゴリへの信頼への期待しか見えなくても，多数のWebサイトのトラストスコアを集めていくことで，その人の良い点における能力の投影になったり，もしくは，弱い点が浮かび上がっていくことが考えられる．現状，多数のメディアサービスを運営するベンチャー企業を買収している大企業が存在するが，現時点のビジネスモデルでは広告ビジネスにおけるシナジー効果に限定されている．そのような企業においてトラストスコアを活用しアイデンティティの評価を行える

ようになることで、新たなビジネスチャンスを生むと同時に、インターネットを活用するユーザ同士のコミュニケーションに、より高度な結びつきを提供することが期待できる。

そして、トラストスコアの設計を前提とした Web サービス設計において、利用者が長期間定着するようなサイト設計手法を確立したいと考えている。Web サービスの企画で、漫然とした情報発信機能やシェア機能などを企画してしまう企画者は多い。とりわけ評価情報についてサービスの根幹であることを意識して設計する方法論はゲーミフィケーションなどに限られている。結果的にサービスをリリースしてから再訪問施策のための評価機能を考えているようなケースは少なくない。それは新規の Web サービスが継続的に利用してもらうことの方が難易度が高いため、その後のことまで設計していないことが考えられる。

それに対してトラストスコアを用いた設計論では、企画の段階から Web サービスを情報発信と評価が時間軸で絡み合っていく生態系と見立て、利用者が情報発信を楽しめる仕組みを考えると同時に、評価情報によるサービスの優等生がアクティブユーザとして心地よく活動しつづけるという考え方を実現したいと考えている。サービス利用者の上位層だけでなく中間層やカジュアルユーザが情報発信とその評価に刺激を受けて継続的に Web サービスを活用することが重要である。そのためにはトラストスコアの要素である情報発信と評価が Web サービスで活躍するための能力と連動して設計されていることが求められる。その結果、アイデンティティを継続することが、Web サービスを楽しむ方法論として認知させ、インターネットを介した情報発信者と情報の受益者の間で信用を抱くことができる世界の実現に寄与していきたいと考えている。

第9章

結 論

本研究では、インターネットの情報の非対称性を前提とした不確実な状況において、ユーザが発信している情報とその評価を用いたトラストスコアを算出することで、トラストスコアが指し示す評価カテゴリについての信頼への期待のされやすさを検証した。実施例としてGithubの公開情報から算出したソフトウェア開発専門職の能力を示すトラストスコアについて有識者にインタビューを行うことで、トラストスコアの高い人材像は期待されやすい人材であるとの相関を示した。

トラストスコアの設計にある、情報発信頻度とその評価方法および評価カテゴリの設計については、多数のユーザを獲得しているWebサービスに兼ね備わっているユーザインターフェースに依存している。トラストスコアは、Webサービスで活躍するのに必要な能力を示す評価カテゴリと、Webサービスにおいて理想的なユーザ行動を評価する仕組みをWebサービスが提供していることが前提になっている。そのような要素が利用者間で肯定的に活用されているからこそ、そのWebサービスで活躍するのに必要な能力からアイデンティティの能力に対する信頼への期待をもたせることができるようになっていく。評価指標によってWebサービスの利用者はより活性化し、サービス設計の意図通りにユーザが活動していることがトラストスコア成立の前提条件となる。

また本研究の特徴は、インターネットにおいてはマイナスの評価値は測定できないという特徴を導き出し、炎上さえもプラスの評価を得るための方法論として扱っているところである。炎上において情報発信の継続性を阻むものは、情報発信者が情報発信を継続することへの意思を失ってしまうことや、社会的責任によって、自分自身が情報発信を止めてしまうことにある。しかし、インターネット上で情報発信を継続して、評価がマイナスになるなどが要因で全員から無視されたりなどで情報発信の継続機会を失った人は存在しない。情報発信の継続性は、何よりも評価を得るために必要なことであることを示した。このような抽象化によって、さまざまなインターネットの情報発信における信頼への期待のされやすさを

算出できるようになった。

トラストスコアの目的が、本来心の内側にある部分を推し量るために不確実な状況における信頼への期待のされやすさを指標化するものであるため、企業に対する従業員のロイヤリティなどを行動履歴や周囲の関係性からスコアリングするという応用など幅広い分野に流用することも可能であると考える。今後は、Webサービス上のアイデンティティ間のコミュニケーションはもちろん、Webサービスをビジネスプラットフォームとみなした場合のサービス運営者とアイデンティティとの間の関係性においても、社会的な信用の担保が設定される前の段階において、発信されている情報を元に信頼への期待を抱き、特別な機能を提供したり、フィンテックとしての金銭の融資におけるオンラインによるスクリーニングへの応用も期待できる。[58]

昨今のWebサービスの成功法則として、厳密な与信審査等を通じた契約関係とは別に、Webサービスの利用規約程度の契約で、敷居はできるだけ低く緩い関係性において、金銭を伴った社会活動にまで繋がるような、サービス間もしくはユーザ間の信頼関係をどのように紡いでいくかが求められている。そのためにWebサービス上の行動履歴等を解析することで、そのユーザがどのように信頼できるかを推測するような機能は今後主流となる開発手法として期待されている。

そのようなWebサービスの設計方法において、継続的な行動履歴と周囲の評価を元に算出するトラストスコアのモデルは広く応用が可能で有用性があると考えられる。また、トラストスコアを軸としたWebサービスの設計論まで昇華させることで、利用者が長く情報発信を行い、相互に評価しあい、信頼し合えるWebサービスの設計論を開発することで、トラストスコアがサービス設計の核となるフレームワークとして活用ができるように持っていきたいと考えている。つまり、Webサービスの設計手法にユーザ間の継続的な信頼への期待のされやすさの醸造という概念を持ち込む。今後もより成熟したコミュニケーションサービスを提供する方法論への発展を継続的に模索していきたい。

謝 辞

本研究の指導教員であり，幅広い知見からの確な指導と暖かい励ましやご指摘をしていただきました慶應義塾大学大学院メディアデザイン研究科の砂原秀樹教授に心から感謝します。

研究の方向性や論文執筆について様々な助言やご指導，暖かい励ましをいただきました，慶應義塾大学大学院メディアデザイン研究科の加藤朗教授に心から感謝いたします。

研究指導や論文執筆等，数多くの助言をいただきました，慶應義塾大学大学院メディアデザイン研究科の古川享教授に心から感謝いたします。Github の活用について助言いただいた時には霧が晴れるように視野が広がりました。

研究指導や論文執筆等，数多くの助言をいただきました，慶應義塾大学大学院メディアデザイン研究科の岸博幸教授に心から感謝いたします。岸教授のご指導において本研究の生かしどころが明確になり，研究がスムーズに進むようになりました。

審査委員をお引き受けいただき，また数多くの助言やご指摘，丁寧なご指導をいただきました，関東学院大学の折田明子准教授に深く感謝いたします。お忙しい中，細かい箇所のご指摘など丁寧にご指導をいただき，50 人以上の CTO 級エンジニアへのインタビューが実現できたことを感謝しております。

さまざまな視野から研究活動を支えていただきアンケート，インタビューにもお答えいただいた慶應義塾大学大学院メディアデザイン研究科 NetworkMedia Project の皆様に心から感謝いたします。お忙しいにも関わらず研究評価のアンケートとインタビューのお願いに快くご協力いただいた Facebook の CTO's グループに参加するスタートアップ各社の CTO 級エンジニアの方々に心から感謝いたします。

最後に，研究活動に関するご理解をいただき，生活面において支援していただいた妻に心から感謝いたします。

参 考 文 献

- [1] Web2.0 と科学技術コミュニケーション, 石村源生, 科学技術コミュニケーション = Journal of Science Communication, 2007.
- [2] 不安意識調査におけるネットサービスのカテゴリ別差異, 山本 太郎, 関 良明, 高橋 克巳, コンピュータセキュリティシンポジウム 2013 論文集, 4, 239 - 246 , 2013-10-14
- [3] 消費者向け Web サイトを取り巻く環境の変化と備えるべき基本要件, 渡邊充隆, UNISYS TECHNOLOGY REVIEW 第 110 号, NOV. 2011
- [4] ソーシャルメディアにおける信用形成とプライバシー, 折田 明子, 計測と制御 55(1), 47-52, 2016
- [5] 本音を言うのは別名で、「サブアカ」を使い分ける中高生の思い, 高橋暁子, 日経 IT pro <http://itpro.nikkeibp.co.jp/atcl/column/14/537662/111400017/>
- [6] モイ株式会社. TwitCasting. <http://twitcasting.tv> , 2009.
- [7] 広告コミュニケーションにおける有名人広告の効果と課題, 朴 正洙 , 産業経営 (44), 21-37, 2009
- [8] ISO/IEC 24760-1:2011 , <https://www.iso.org/standard/57914.html> , 2011
- [9] アイデンティティ管理技術解説 , 独立行政法人 情報処理推進機構 技術本部 セキュリティセンター , <http://www.ipa.go.jp/security/idm/> , 2013
- [10] インターネット時代の法常識, 伊藤博文, 2002
- [11] 信頼の構造 - こころの社会の進化ゲーム, 山岸俊男, 東京大学出版会, 1998.
- [12] ”インターネットトラブルの分類方法の提案.”, 田代光輝, 情報社会学会誌 6.1 (2011): 101-114.

- [13] ソーシャルメディアの共感が購買行動に及ぼす影響, 泉水清志, 育英短期大学研究紀要 31, 1-14, 2014
- [14] リンクを含むつぶやきに着目した Twitter の分析, 吉田光男, 乾孝司, 山本幹雄, March 15-16, 2010
- [15] ブログ空間上のコミュニケーション発生メカニズムの分析, 山本 仁志, 諏訪博彦, 岡田 勇 [他], 山本 浩一, 日本社会情報学会学会誌 20(1), 29-39, 2008-09-30
- [16] ソーシャルリスニングの有効性と限界: 書き込みの信頼性と安定性に関する検証実験から, 長島 直樹 - 経営論集, 2015/03.
- [17] なぜウェブで炎上が発生するのか: 日本のウェブ文化を手がかりとして, 平井 智尚, 情報通信学会誌 29(4), 61-71, 2012-03-25
- [18] ネット炎上の発生過程と収束過程に関する-考察-不具合に対する嫌がらせと決着による収束-, 田代 光輝, 折田 明子, 研究報告電子化知的財産・社会基盤 (EIP) 2012-EIP-57(6), 1-6, 2012-09-06
- [19] インターネットにおける「ブログ炎上」に関する一考察, 平井智尚, 慶應義塾大学大学院社会学研究科紀要, 2007.
- [20] 炎上マーケティング, <https://ja.wikipedia.org/wiki/炎上マーケティング>
- [21] first monday. The Attention Economy and the Net, Michael H. Goldhaber, 1997.
- [22] Understanding Participatory Hashtag Practices on Instagram: A Case Study of Weekend Hashtag Project, Changhoon Oh, Taeyoung Lee, Yoojung Kim, SoHyun Park, Bongwon Suh Seoul National University, Seoul, South Korea, 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems Pages 1280-1287
- [23] 本人確認基盤と公開 ID の提案 - Web 横断的な匿名本人確認と OpenID の有用性について -, 岡下綾, 電子情報通信学会技術研究報告. SITE, 技術と社会・倫理 108(459), 55-60, 2009-02-26

- [24] プロバイダの発信者情報開示義務， 前田泰， 群馬大学社会情報学部研究論集 18, 227-241， 2011-03-31
- [25] トリップ (電子掲示板)， wikipedia <https://ja.wikipedia.org/wiki/トリップ>
- [26] インターネット異性紹介事業を利用して児童を誘引する行為の規制等に関する法律， law.e-gov.go.jp ， <http://law.e-gov.go.jp/htmldata/H15/H15HO083.html>
- [27] Retty 株式会社 Retty ， <https://retty.me/> ， 2010
- [28] ヤフー株式会社 ， <https://auctions.yahoo.co.jp/> ， 1999
- [29] 株式会社メルカリ ， <http://mercari.jp/> ， 2013
- [30] 日本の SNS 利用者の書き込み記事に対する主観的信頼度の調査， 石渡 僚， 桑原恒夫， Science Journal of Kanagawa University, 21: 13-16, 2010
- [31] 消費者金融サービス市場と個人信用保証機関， 藤原七重， 消費者金融サービス研究学会年報， 2002
- [32] アメリカの無担保社債金融， 正木久司， 同志社商学 27(5)， 55-80， 1976
- [33] 「信用力」格差社会， 岩田昭男， 東洋経済新報社 ， 2008
- [34] アメリカ生活とパーソナル・ファイナンスの総合情報サイト， <http://www.fiplanning.com/credit/credit3.html>
- [35] Klout, Inc. , Klout ， <https://klout.com> ， 2008
- [36] Klout Score: Measuring Influence Across Multiple Social Networks, Adithya Rao, Nemanja Spasojevic, Zhisheng Li, Trevor DSouza , 8 pages, 2015 IEEE International Big Data Conference - Workshop on Mining Big Data in Social Networks , 2015
- [37] Twitter,Inc. , ツイッター,<https://twitter.com/> , 2006
- [38] Kakaku.com,Inc , 食べログ ,<https://tabelog.com/> , 2005
- [39] GitHub Inc. ,GitHub,<https://github.com>, 2008

-
- [40] Cookpad,Inc ,クックパッド ,<https://cookpad.com/> ,1998
- [41] ミシュランガイド ,<http://www.michelin.co.jp/JP/ja/guides-goods/michelin-guide/about.html> , 2008
- [42] <https://git-scm.com/> , 2005-12-21
- [43] Quality and Productivity Outcomes Relating to Continuous Integration in GitHub, Bogdan Vasilescu , Yue Yu , Huaimin Wang , Premkumar Devanbu , Vladimir Filkov, ESEC/FSE 2015 Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering , Pages 805-816, 2015-08-30
- [44] ruby/ruby:The Ruby Programming Language , <https://github.com/ruby/ruby> , 1993
- [45] Python: Git Source Code Mirror , <https://github.com/python> , 1991
- [46] rails/rails: Ruby on Rails , <https://github.com/rails/rails> , 2004
- [47] <https://about.gitlab.com/> , 2011
- [48] <https://bitbucket.org/> , 2008
- [49] 複数のオープンソースプロジェクトに参加する開発者による貢献の分析, 坂口 英司, 伊原 彰紀, 尾上 紗野, 畑 秀明, 松本 健一, 研究報告セキュリティ心理学とトラスト 15, 1-4, 2014-05-08
- [50] Understanding "watchers" on GitHub, Jyoti Sheoran, Kelly Blincoe, Eirini Kalliamvakou, Daniela Damian, Jordan Ell, MSR 2014 Proceedings of the 11th Working Conference on Mining Software Repositories , Pages 336-339 , 2014-05-31
- [51] 組込み技術者向けキャリアガイドの開発 , 井上 克郎 , SEC journal Vol. 8 (2012) No. 2 SEC journal No.29 p. 85-88 , 2012-07-24
- [52] <http://ghtorrent.org/> , 2012
- [53] <https://dev.mysql.com/> , 1995

-
- [54] Lean GHTorrent: GitHub data on demand, Georgios Gousios, Bogdan Vasilescu, Alexander Serebrenik, Andy Zaidman, MSR 2014 Proceedings of the 11th Working Conference on Mining Software Repositories , Pages 384-387,2014-05-31
- [55] GHTorrent: Github 's Data from a Firehose, Georgios Gousios, Diomidis Spinellis, MSR '12 Proceedings of the 9th IEEE Working Conference on Mining Software Repositories , Pages 12-21 , 2012-06-02
- [56] 検索エンジンの概要, 山名早人, 村田剛志. 情報処理 46.9 981-987, 2005
- [57] Adaptive Approach for Gamification Optimization, David Codish, Gilad RavidUCC '14 Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing Pages 609-610, 2014
- [58] フィンテックスタートアップのビジネスモデル, 藤川真一, 情報処理学会誌 : IPSJ magazine 57(9), 870-876, 2016-09

研 究 業 績

国内研究会

1. スマートフォンを「鍵」としたウェブサイトへの自動ログイン機構の提案.
， 藤川真一, 山内正人, 砂原秀樹. インターネットと運用技術シンポジウム 論
文集 2013 (2013): 53-57, 2013

その他

1. フィンテックスタートアップのビジネスモデル, 藤川真一, 情報処理学会誌
: IPSJ magazine 57(9), 870-876, 2016-09

付録A

トラストスコアのアンケート

A.1. アンケート表紙

研究協力をお願い

(実験実施者：慶應義塾大学大学院メディアデザイン研究科 藤川真一)

博士論文の研究の一貫として、エンジニア採用判断の実験を行います。実験は、(1) アンケート および (2) Skypeによるインタビューの形式で実施します。

得られたデータは、氏名や企業名を伏せ、個人を特定できないように処理した上で、博士論文の内容に利用する他、ブログでの公表などを予定しています。上記の条件で実験に協力いただけるのであれば、ページ下部からのアンケートにご回答をお願いいたします。

なお、実験にご協力いただいた方には、後日結果をご報告いたします。

<説明>

このアンケートでは40名分の「トラストスコア」および職務経歴を読んでいただき、「採用面談してもよい」「採用面談したくない」の判断をしていただくものです。

「トラストスコア」とは、Githubの個人アカウントが所有するパブリックリポジトリへのソースコードのコミットの頻度や期間と、Star数を勘案して、【0から5】の6段階のスコアとして算出しているものです。

一定以上のコミットが行われていれば【1から5】のスコアをとります。5が最高点で、1が最低点となります。

Githubの記述がありますが、「ソースコードを見て判別できないから回答不可」とはせずに、このアンケートの文面のみから、エンジニアの視点から印象としてどう判断するかを選んでいただけますと幸いです。

回答内容を元に後日 skype mtg でいくつかの質問をさせていただく予定です。skype などのオンラインでインタビュー（10分程度）できる情報をお教えてください。

※注意：本アンケートに登場した「トラストスコア」については、口外しないようお願いいたします。

A.2. 基本質問

1. お名前
2. 会社名
3. 役職
4. 業務上、人事採用に関わっていますか？
5. ご自身の職歴（現在の会社でN年）
6. skype-id など（最後のビデオチャットインタビューの連絡先）

A.3. 共通の選択肢

ペルソナ判断の選択肢

- 採用を前提に面接してみたい
- やや面接してみたい
- どっちでも良い（部下に聞いてみる）
- あまり面接したくない
- 面接しない

A.4. 職務経歴書アンケート 1（最初の 20 人）

まず最初に 20 人の職務経歴書を選択ください。

共通条件として、希望年収は全員年収 600 万円で、年齢は 32 歳とします。希望職種はオープンポジションではありますが、ソースコードを書く【開発専門職】への応募としてご判断ください。

1 さんの職務経歴書

大学を卒業して、某上場企業の SIer で IT システムに関するコンサルタント兼アーキテクトをやっています。官公庁の Web サービスに必要な複雑な業務設計と Azure や AWS を活用したアーキテクチャ設計は得意です。

ただし現状は大きなアーキテクチャ設計までしかやっておらず、実際にサーバを構築するチームとの温度差があり歯がゆい経験もしてきたためもっと下のレイヤーまで自分でやりたいと思い転職を決意しました。

今回は、自分でクラウドを活用したアプリケーションのプログラムを書きたいと思って転職活動中です。Github では、プログラムの勉強に使った Ruby on rails のアプリケーションが公開されています。

2 さんの職務経歴書

大学を卒業後、1 部上場の SNS 系 M 社に技術の企画職として入社し、その後、起業して自社サービスを運営していました。残念ながら会社はそれほどスケールしなかったため、サービスを売却して会社を畳みました。

次は、開発のマネジメントには自信がありますが、改めて現場から開発できたらと思っています。

技術としては Perl、PHP、Ruby は使えます。自社サービスは PHP の Laravel もしくは Ruby on rails で動いているサーバの両方を開発していました。

Github には、前職の会社で開発していた Perl によるツールを公開しています。

3 さんの職務経歴書

大学を卒業してから、派遣で経理事務をやってきました。
Excel のピボットテーブルや VBA によるマクロのコードを書くことができます。

経理の現場でずっとやっていくつもりでしたがプログラムが楽しくなってきたことと、もっと大量のデータを取り扱えるようになりたいと思い開発者になる決意をしました。

開発の仕事は未経験ですが、腰を据えて働きたいのでよろしく願いいたします。

Github には、自分で開発した pivot テーブルを活用するマクロのスク립トを公開しています。

4 さんの職務経歴書

大学を卒業して、新卒入社した大手ネット企業でオークションの開発をやっていました。

主に基盤チームが作った api を活用した UI の開発です。

強みは、Web のパフォーマンスを考慮した開発や、UX を意識した開発です。

弱みとして、基盤チームが深いデータベース周りの技術を担ってい

たので、自分自身が、もう少し SQL を活用する部分からの開発をしたいので転職先を探しています。

Github には、SQL を勉強するのに作ったサンプルソースコードが置いてあります。

5 さんの職務経歴書

大学を卒業してからプロブロガーをやってきました。
毎日、ブログでガジェットのニュースを追っており、IT の話題には詳しいです。

ブログは 100 万 PV/月を達成して目標を達成して、一人で続けていく限界が見えてきたため、今後はチームで働きたいと思っています。

自分でサービスを作る力をつけたい！と思って転職を希望していません。

開発業務経験はありませんが、Ruby 言語を勉強しており、Github には RSS を読み込むクローラを公開しています。

6 さんの職務経歴書

大学を卒業して、SIer のソフトウェアハウスから派遣される SES をやっていました。

お客様には大変恵まれており、携帯電話会社で Android 携帯に搭載するツールの開発をやっていました。

Android OS に関する深い知識を得られたので、今度はもっとアプリケーションの部分で、ユーザーさんの顔が見える仕事をしたいと思い、転職を決意しました。

自分の Github には、Github でフォークした Retrofit という Android

用通信ライブラリに機能を足したものを公開しています。

7 さんの職務経歴書

大学を卒業してから、商社の子会社でサーバ管理をやってきました。

Windows サーバのオペレーションが得意です。ソースコードは、ActivePerl を使った管理スクリプトなどを作ったことがあります。

開発経験は未経験ですが、協力会社 10 人に指示を出しながらのプロジェクト管理を行い監視業務の効率化のためのシステム構築経験があります。

その時には、要件定義、詳細設計、開発ディレクション、受け入れテストなど一連の開発経験を行っています。

その経験で、業務プロセスの設計をすることに目覚めて、さらに自分でコードを書いて幅を広げたいと思いまして、個人でも Ruby の勉強もしてきました。

今後、よりスキルを高めたいと思い、転職を決意しました。

Github では、AWS SDK を使ってサーバインスタンスの増減を自動コントロールを簡単に行う Ruby スクリプトを公開しております。

8 さんの職務経歴書

大学を卒業してから、スタートアップで Android を書いており、公開したアプリが Google から年間大賞をいただきました。その後、体調不良で半年間休職し、復帰できる準備が整いました。

体調は元気になりました。

どうにかがんばりたいと思い転職先を探しています。アプリ開発で

はお力になれると思っています。

Github には、Java で開発したデスクトップ向けの Twitter 投稿アプリを公開しています。

9 さんの職務経歴書

大学を卒業してから販売員を経験した後、職業訓練校に通って Android を学びました。

Android の育成コースでは脱落する人が続出して続けるのが大変でしたが、無事に主席で修了しました。

卒業制作でアプリを作ってみたので、面接の際には是非見ていただきたいと考えております。

今後は Android の開発を通じて仕事をしていきたいと思っています。Github では、卒業制作のアプリを公開しております。

10 さんの職務経歴書

大学を卒業してから、R 社で EC コンサルタントとしてお店の営業をしてきました。

EC コンサルタントの立場でしたが、お客様のお店を作るお手伝いもやっておきておまして、HTML や CSS は Web 標準レベルで理解しております。

ただ EC サイトでは SEO を重視するなどの限界があり、オーソドックスな JavaScript などしか使えない状況にあります。

今後は開発力をつけて、Web サービスでユーザさんに貢献したいと思っています。

開発は未経験ですが、エンジニアと一緒に仕事をするのは得意でし

た。

Github では、勉強のために書いた node.js と React.js を使ったプログラムを公開しています。

1 1 さんの職務経歴書

大学を卒業して、一次請けの大手 SIer 企業で SE として働いていました。

プロジェクトの PM も経験するなどの経験があります。
開発者 5 人を含む 10 人程度のチームを率いて、お客様の案件をこなしておりました。

業務では協力会社がコードを書いておりますが、私は業務でコードは書いておりませんが、自分でもメンテナンスできるように開発者の書いたコードはテストやコードレビューを行っておりました。

Github には、友人と開催したイベント申し込み用の cgi スクリプトを公開しています。

1 2 さんの職務経歴書

大学を卒業し、自動車メーカーで機械設計のエンジニアをやっていました。

CAD で図面を設計して、製造工程に落とし込む部分の調整を行っておりました。

自動車のメーターや操作パネルの設計をやっていたので、特に UI 面の使い勝手に対しては意識が高いと思います。

ただし、ソフトウェアの実務はやったことないですが、家で勉強したり、ハッカソンに参加してチーム開発をするなどソフトウェア開

発の経験を積んできました。

某ハッカソンアワードでチーム優勝をしたこともあります。

Github では、iOS アプリのさまざまな使い勝手のボタンを作ることができる UI ライブラリを公開しています。

1 3 さんの職務経歴書

大学を卒業して、データセンターの会社で、ネットワーク設計を行う部門でネットワーク、インフラエンジニアとしてやってきました。会社では、シスコのルーターを設定するだけでなく、自動構成管理を行うスクリプトなどを書いていました。

AWS などのクラウドはライバル企業であることもあり、趣味で触る程度ですが、自分でアカウントを取得し、teraform を活用して、EC2 / RDS / Redis / ECS などを活用した Web サービスの自動構築などをして勉強しています。

将来を見据えて AWS がいじれる環境で実務経験を積んでいきたいと思ひまして転職を決意しました。

Github では、ネットワーク設定に関する設定資料のドキュメントを公開しております。

1 4 さんの職務経歴書

大学を卒業して、ソーシャルゲームの会社で制作のディレクターをやっていました。

毎月のキャンペーンの運用と、新規ゲームの開発ディレクションを行っています。

20 人のチームで、毎月サーバの改善なども含めた業務に携わっています。

もっと手を動かして、サービスを自分で作りたいと思いました。
JavaScript や Unity を勉強したのでゲームや Web サービス問わず、
フロントエンドエンジニアとして転職先を探しております。

Github には、JavaScript による簡単なゲームを公開しています。

15 さんの職務経歴書

大学を卒業してから、出版も営む IT 系メディア企業に入社し、スタートアップ情報メディアの編集長をやっていました。
メディアのメンバーはインターンあわせて 10 人ほどで記事の制作を毎日回していました。

少数精鋭ではありますが、実際にスタートアップの代表に取材し、しっかり記事を書くことでは業界内では定評があります。

今回は、編集長ではなく開発者として一から出直したいと思っており御社を志望いたしました。

Github には、勉強のために作った Go による RSS リーダーアプリを公開しています。

16 さんの職務経歴書

US のスタートアップでサポートエンジニアをやっていました。アメリカ人ですが、子供の頃、日本の学校にも通っていたので日本語は問題ありません。

アニメが好きなので日本に来て就職したいと思っています。

開発の実務経験としてはゼロからの開発はありませんが、サービスの技術的なトラブルシューティングは得意です。

開発言語は自分で PHP、Python、SQL などを勉強しています。

Github には、アニメの情報サイトから画像をスクレイピングする Python スクリプトを公開しています。

17 さんの職務経歴書

大学を卒業して、日本の Google でエンジニアをやっておりました。広告ターゲティング技術の開発をしており機械学習が得意です。

開発技術は主に Java、Python などです。仕事柄 GCP などは日常的に使っています。

今回、もう少し違うサービスで AI 開発を行いたいと思い転職を志望しています。

個人では特に Github にソースコードの公開はしてはしません。

18 さんの職務経歴書

学生時代から未踏プロジェクトに参加し、今までにない Web サービスの開発を行っていました。

大学を卒業してからは、R 社に入社し飲食店の予約のサービスを開発しています。

将来の目標であるアーリーステージの会社で CTO を目指せるような仕事ができればと思います。

メインは Java でサーバサイド開発をしています。Ruby や Python も普通に書いています。データベースは Oracle と MySQL を使っています。

Github には、友人と開催した開発技術のイベントの申込みサイトの Rails によるソースを公開しています。

19 さんの職務経歴書

フィリピン在住、フィリピン出身のエンジニアです。
大学は地元の大学を卒業し、主に Java を使ってオフショアでの開発をおこなってきました。
ブリッジ SE もやっており日本語には自信あります。

この度、日本で働きたいと思って就職を希望しています。
Github には、ImageMagick をラップした画像ライブラリを公開しています。

20さんの職務経歴書

大学で自然言語処理を学び、大手 Web 系大手の Y 社に新卒入社し、今年で 10 年目です。検索エンジンのサービス開発と研究業務に従事してきました。

今後は自然言語処理のスキルを活かして自動化 bot や音声 AI、リコメンドエンジンなど、日本語データ解析を活用した仕事を探しています。

Github では、プライベートで作っている Python 言語のバージョン切替や環境変数を簡単に変更できるアプリケーションを Github に公開しています。

A.5. 職務経歴書アンケート 2（残りの 20 人）

ここまでお疲れ様でした！あとちょっとだけお付き合いください！

残りも同じく、共通条件として、希望年収は全員年収 600 万円で、年齢は 32 歳とします。

希望職種はオープンポジションではありますが、ソースコードを書く【開発専門職】への応募としてご判断ください。

2 1 さんの職務経歴書

大学を卒業して、大手 SIer で開発を行っておりました。
普段は、Java + Oracle で金融系のシステムを開発しています。
ミッションクリティカルなサービスを素早く作るのは得意です。

金融の現場はどうしてもスピードが遅いので、もっとスピード感を持った仕事をしたいと思っており、Web サービスに携わりたいと思っています。

Github では、5 年前に公開した掲示板スクリプトをずっとメンテナンスしており、トラストスコアは【 2 】です。

2 2 さんの職務経歴書

大学卒業後、Web システム開発の受託の会社で開発をやってきました。15 人のチームの開発マネジメントも行っていました。

受託なので、さまざまな技術を使っていますが、一番得意な開発言語は、Python です。

今回、継続的に開発し続ける Web サービスに携わりたいと思って応募致しました。

Github には、Go 言語で書いた chat bot を公開しています。

ソースを公開している chat bot は珍しいのでメディアにも取り上げられ、トラストスコアは【 3 】です。

2 3 さんの職務経歴書

大学を卒業してから、住宅メーカーの営業支援をやってきました。
仕事の傍ら、Wordのマクロを使って自動的に営業文章を作るツールを作ったりしました。
営業の仕事も楽しかったのですが、ITで人を支援する仕事の方に興味がでてきまして開発者になりたいと思い志望しました。

仕事での開発経験は未経験ですが、VBAに限らず、iOSでswiftも勉強しております。
技術情報を学ぶことに自信はあります。

Githubには、営業文章を自動生成するWordのマクロを公開しております。gigazineにも取り上げられて好評でした。
トラストスコアは、【2】です。

24さんの職務経歴書

大学を卒業して、新卒入社した会社で、広告サーバの開発とプロジェクトマネジメントをやっていました。
主にやっていたのは、広告周りのエンジニアマネージャ、新規事業の企画をやっていました。

DSPの開発をしていたのでスケーラビリティ周りの苦勞とUIの開発については経験が深いです。
今回、改めて自分自身でコードを書く開発をやりたいと思い開発職を志望しています。

Githubには、Ruby on railsで作ったマストドンの機能拡張を公開しております。
国内のマストドンサーバの定番プラグインになっており、トラストスコアは、【3】です。

25さんの職務経歴書

大学を卒業してからフリーターをやってきました。
今は、渋谷の109のお店で店員をしながら仕入れも任されています。

売り上げについても、毎日お店の改善を続けたことが認められてグループ内表彰もされました。
地道な改善が得意です。

仕事での開発は未経験ですが、正社員として就職したいので転職を希望しています。

Githubには、KotlinでAndroid用に開発したファッションアプリのソースコードを公開しています。
公開以降、沢山のエンジニアの方々から支援していただき、トラストスコアは、【2】を獲得しております。

26さんの職務経歴書

大学を卒業して、SIerの中でパッケージの開発をやっていました。
.NETとMS SQL Serverを使って、e-learningのCMSを開発していました。

e-learningのソフトウェアは主に企業への導入をしておりトップダウンで使われるものなので、あまり使われている感が得られなかったというのがあります。

今回の転職では、C向けのWebサービスでボトムアップで、沢山のユーザーさんに使われるサービスを作って、もっと手応えを得られないかと思い転職活動をしております。

今回、LL言語による仕事は初めてですが、家ではPythonの勉強をしてリアルタイムの掲示板サービスを公開しています。Python以外にも、さまざまなサーバサイド言語を勉強してみたいです。

Githubには、Pythonのflaskとnode.jsを使ったリアルタイム掲示板のWebアプリケーションを公開していて、モダンな掲示板スクリプトは今時、珍しいので結構使われています。トラストスコアは、【4】です。

27さんの職務経歴書

大学を卒業してからビールメーカーのWeb制作室でフロントエンドエンジニアをやってきました。

業務ではキャンペーンページを作ったり、特集等のLPの制作と、コーポレートサイトのメンテナンスを行ってきました。

ユーザーが直接触る画面を作るのが好きで、自分でもSwiftの勉強を行いアプリケーションを作ってきましたが、一人で勉強するには限界があると思い、もう少しスキルを上げたいと思い応募いたしました。

Githubでは、勉強のために書いたSwiftのソースコードが公開されています。トラストスコアは【1】です。

28さんの職務経歴書

大学を卒業してから、しばらくニートをした後、動画サービスで有名な某D社でスマホのアプリをつくっていました。

朝が苦手な午前中の出社は難しいですが、夜働くのは平気です。

アプリの開発に専念できる転職先を探しています。

Githubでは、自分で開発し公開した動画ストリーミングのライブラリが人気で、トラストスコアは【5】です。

29さんの職務経歴書

大学を卒業してから、ITエンジニアを養成する学校の講師をやってきました。

Microsoftの資格も持っています。得意なサービスはAzureです。学校のイントラネットの構築、メールサーバ、ファイルサーバの管理などを行っていました。

カリキュラムでは、C#.net, Python, SQLなどを教えていたので仕様は十分理解しております。

開発業務は経験は未経験ですが、GithubではLinux上で動かしたcronのバッチが、正しく動いたことを監視するサーバを公開しており、トラストスコアは【2】をいただいております。

30さんの職務経歴書

大学を卒業して、上場しているアフィリエイト企業の営業職をしてきました。

営業という立場ですが、お客様のWebサイトのコードを瞬時に理解して安定的にタグを埋め込む作業などもやってきました。

開発現場の経験はありませんが、顧客要望を伝えるなど開発者と一緒に仕事をするのは得意です。

今回は未経験であるものの一歩進んで開発者になりたいと思って転職を決意しました。

プライベートではGo言語などでプログラミングの基礎を学習しています。

Githubには、自作のGo言語によるプログラムを公開しており、トラストスコアは【2】です。

3 1 さんの職務経歴書

大手 SIer 企業の関連会社である Web 制作会社に新卒入社をし、お客さん向けの開発をしてきました。

最初はソースコードを書いております、ライト案件では PHP、大きめなバックエンド連携が伴うものは Java, Struts2 や Spring フレームワークを使った開発を行っておりました。

途中から会社の方針が変わりオフショア開発が増えブリッジ SE の役割がふえて、あまり日常的にソースコードを書く機会が減ってきたため、もう少し現場でコードを書いていたいため転職を決意しました。

Github では、個人で PostgreSQL のクライアントアプリを Github に公開しておりトラストスコアは【 4 】です。

3 2 さんの職務経歴書

大学を卒業して、某家電メーカーで製品設計をやっていました。企画部と連携して、実際に製品化するための図面設計をしたりプロトタイプを作る仕事です。

基板やファームウェアの開発は外注でしたが、協力会社と連携する UI の設計や画面デザイン、アイコンなどは自分の方で担当しておりました。

また、家電機器の付属アプリなども外注に指示する形で設計と開発ディレクションを行っていました。

今回は開発現場で、Web を通じて沢山の人のサービスを提供したいと思い転職を決意しました。

Objective-C、Swift は勉強しており、Github では、個人で、iOS アプリのブラウザライブラリを公開しており、大変好評で、トラストスコアは【 3 】を獲得しております。

3 3 さんの職務経歴書

大学を卒業して、SIer でネットワークエンジニアをやっていました。

案件ごとにデータセンターに籠ってサーバのラッキングを行い、ネットワーク設定をするところを主に担当していました。

普段から AWS を使った環境構築はやっております。Chef を使った Nginx などの構成管理を行っておりまして、そのスクリプトも書いていました。

お客様だけの環境だとあまり構成管理を求められないケースもありまして、お金をいただけないと構成管理ができない状況ではなく、自社サービスで継続的に構成管理をしていけるような場で働きたいと思い、今回転職を決意いたしました。

Github では、AWS のオートスケールのようなサーバの増減を行う処理を、もっときめ細やかなトラフィックの変動などをトリガーに、できるかぎり素早く行うライブラリを公開しております。

実際に、多くのソーシャルゲーム会社さんなどに使われており、トラストスコアは【 4 】です。

3 4 さんの職務経歴書

大学を卒業して、某 R 社で EC 事業のディレクターをやっていました。

毎月 PDCA を回すマーケティング施策にあわせて、LP を作ったり、特集を作る仕事をやっていました。

もっと手を動かして自分で新しいものを作りたいと思って、HTML5を勉強しています。

Githubには、iframeとAjaxでネットショッピングの時に使う、アマゾンや楽天市場の価格を比較するツールを開発しています。トラストスコア【2】です。

35さんの職務経歴書

大学を卒業してから、ITによるオンライン教育の企業に入社し、教材作成等の編集をやっていました。

今回は、開発者として一から出直したいと思っており御社を志望いたしました。

もともと開発言語は趣味で触っておりRubyは特に得意です。

Ruby on railsのリソース翻訳と日本語処理周りのコミッタとして活動もしております。

オリジナルのソースコードはほとんど公開していないのでトラストスコアは【1】です。

36さんの職務経歴書

大学を卒業して、上場企業であるアフィリエイト企業の営業職をしてきました。

開発現場の経験はありますが、顧客要望を伝えるなど開発者と一緒に仕事をするのは得意です。今回は、未経験ですが開発者になりたいと思って転職を決意しました。

技術については、Python を自分で勉強し、Github にソースコードを公開しています。

個人で開発したアルゴリズムで動く株の自動購入ソフトウェアを公開しており、トラストスコアは【 3 】です。

37 さんの職務経歴書

大学を卒業して新卒で入社した C 社で、ずっとデータ解析に関する開発を行っていました。

Ruby on rails による Web アプリケーション、Python を使ったデータ解析処理が得意です。

今後は、新しい AI の開発に力をいれていきたいと思っています。

個人ではデプロイ時の特定環境からのテスト用の環境設定切り替えなどを便利する Ruby Gem による Rails プラグインをメンテしており、先日、トラストスコアが【 5 】になりました。

38 さんの職務経歴書

学生時代から積極的にハッカソンに参加し、チーム優勝を何度もやってきました。

大学を卒業してからは、携帯キャリアに就職し、提携する Web サービスとの企画をやってきました。

開発言語は、Ruby on rails でよく書きます。

仕事で Web サービスを作る立場になりたいと思って転職を希望しています。

Github には、Web サービスの情報収集をするための bot システムを公開しています。

Ruby on rails を活用してクローリングしています。トラストスコアは【3】です。

39さんの職務経歴書

タイ在住、タイ出身のエンジニアです。

大学は地元の大学を卒業し、主にオフショアでの開発をおこなってきました。

お客さんは日本のSIやゲーム会社で実績があるため日本語によるコミュニケーションは問題ありません。

日本で働きたいと思って就職を希望しています。

Github では、ImageMagick よりも軽量な画像圧縮ツールを開発し公開しています。トラストスコアは【5】です。

40さんの職務経歴書

大学を卒業して、フリーランスで開発者の仕事をしています。

フリーランスは、友人とプロジェクトチームを組んだり、お客様の組織に常駐して仕事をするので、チーム開発には慣れています。

また、独自で開発したQ言語を公開しており、普及に命をかけています。

今回、もう少しチーム開発にコミットしたいと思ったので、転職先を探しています。

Github で公開したQ言語のソースコードは、おかげさまでトラストスコア【5】をいただいております。

A.6. 回答集計

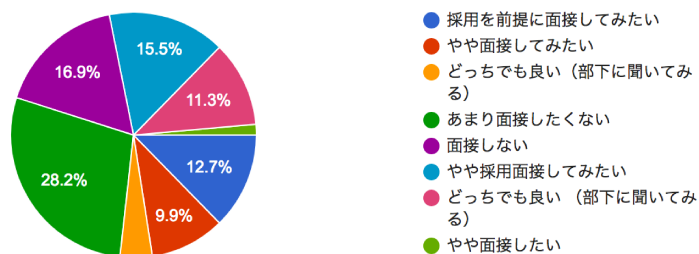


図 A.1 1 さんの評価結果

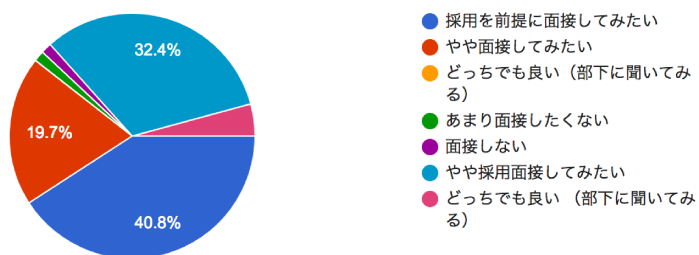


図 A.2 2 さんの評価結果

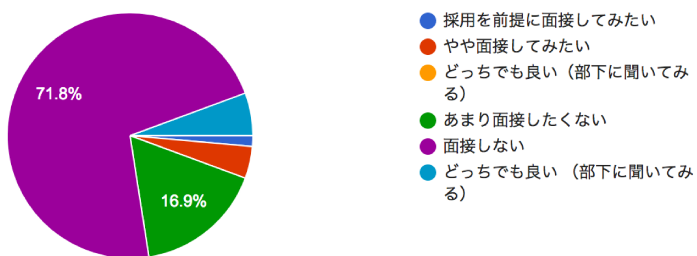


図 A.3 3 さんの評価結果

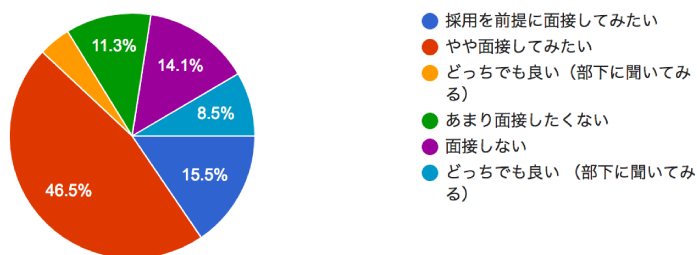


図 A.4 4さんの評価結果

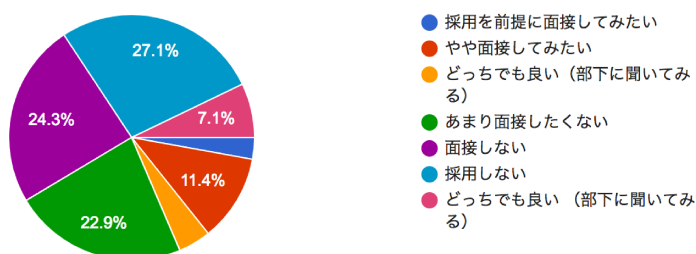


図 A.5 5さんの評価結果

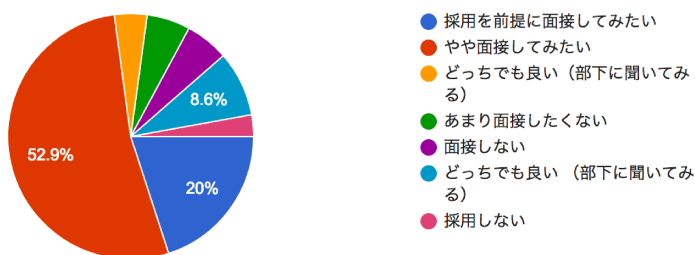


図 A.6 6さんの評価結果

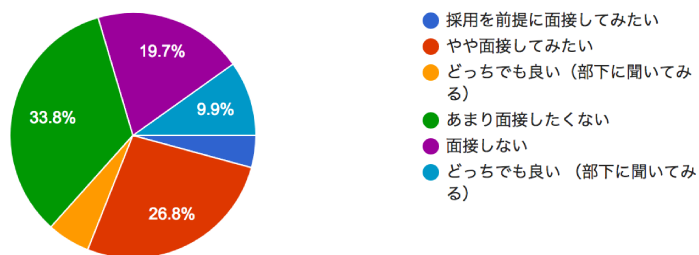


図 A.7 7さんの評価結果

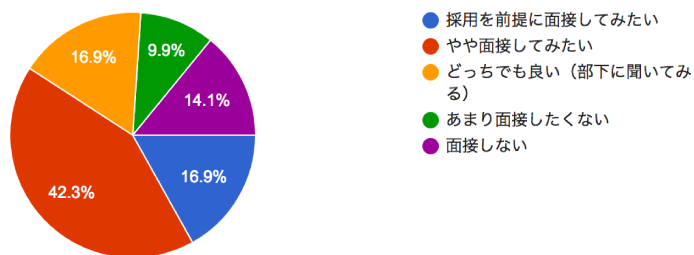


図 A.8 8さんの評価結果

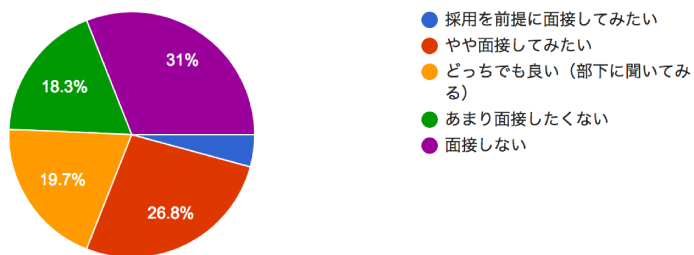


図 A.9 9さんの評価結果

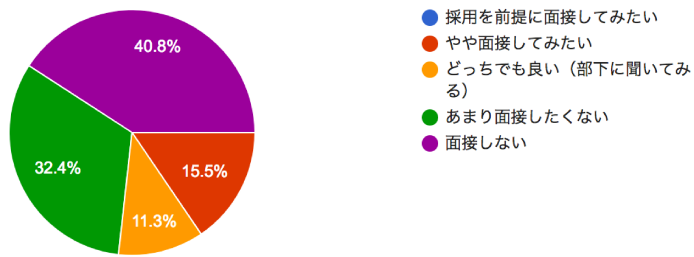


図 A.10 10 さんの評価結果

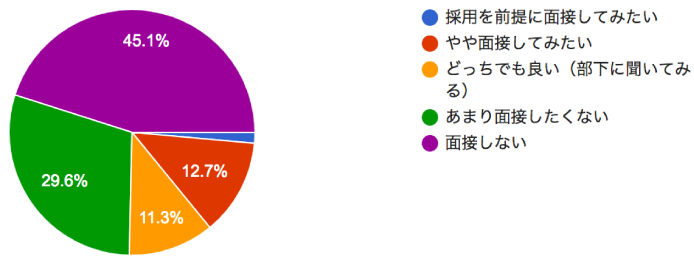


図 A.11 11 さんの評価結果

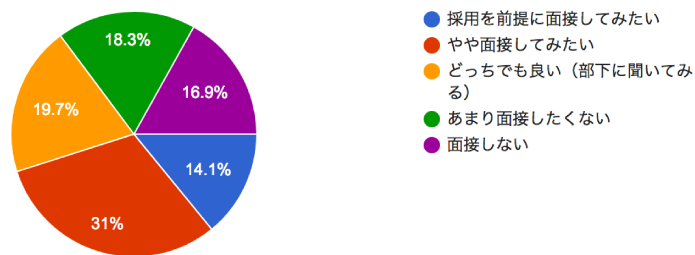


図 A.12 12 さんの評価結果

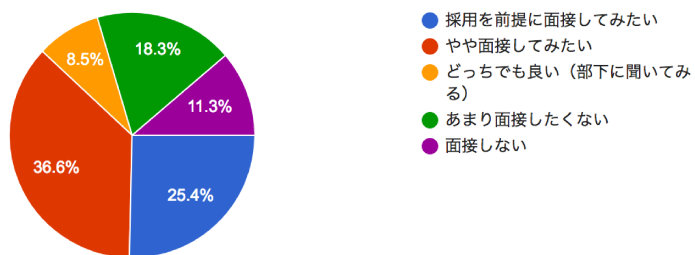


図 A.13 13 さんの評価結果

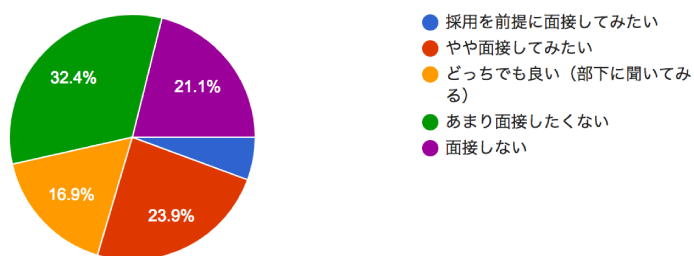


図 A.14 14 さんの評価結果

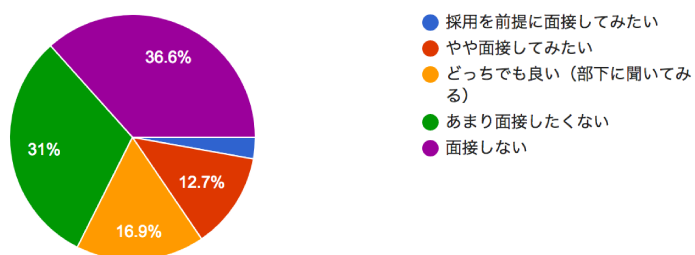


図 A.15 15 さんの評価結果

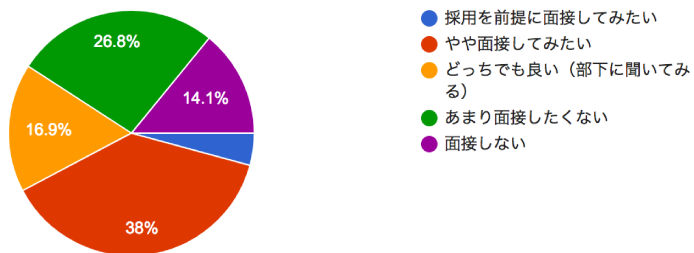


図 A.16 16 さんの評価結果

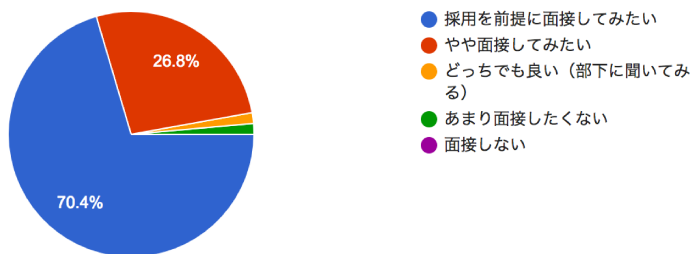


図 A.17 17 さんの評価結果

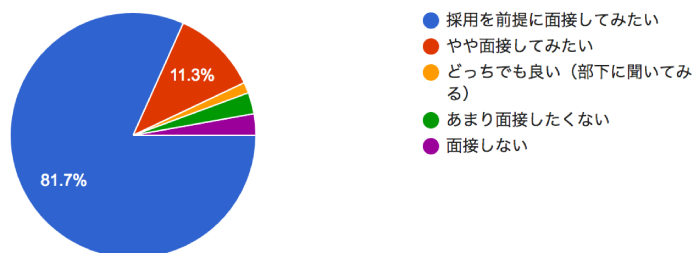


図 A.18 18 さんの評価結果

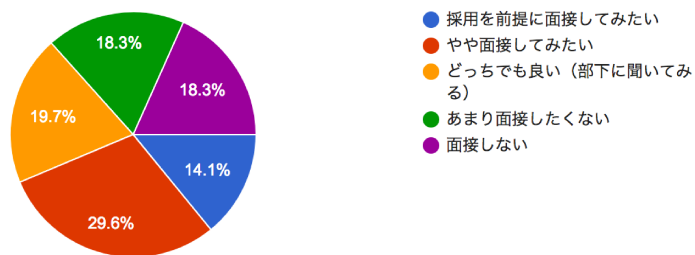


図 A.19 19 さんの評価結果

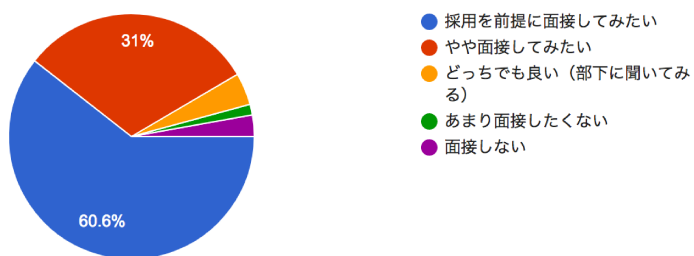


図 A.20 20 さんの評価結果

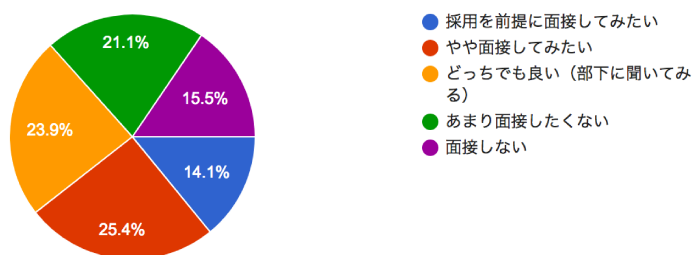


図 A.21 21 さんの評価結果

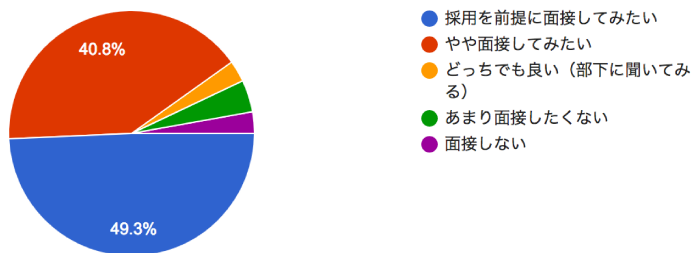


図 A.22 22 さんの評価結果

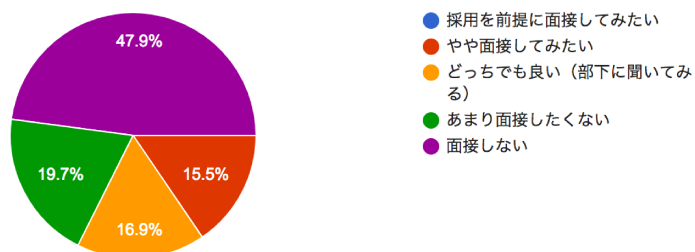


図 A.23 23 さんの評価結果

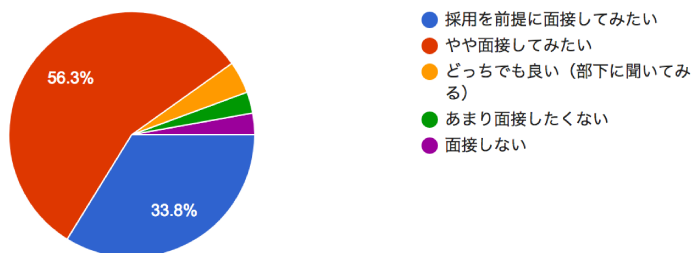


図 A.24 24 さんの評価結果

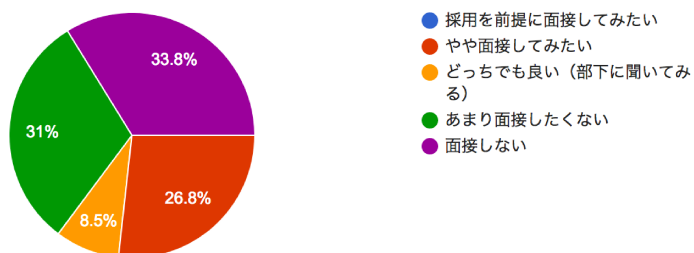


図 A.25 25 さんの評価結果

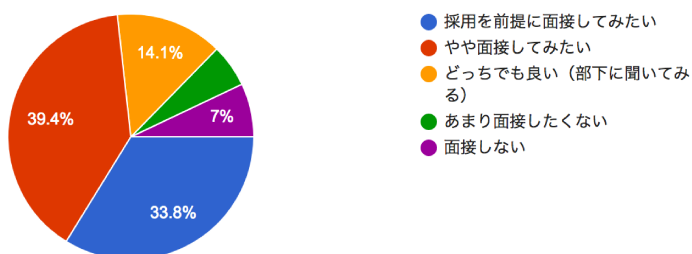


図 A.26 26 さんの評価結果

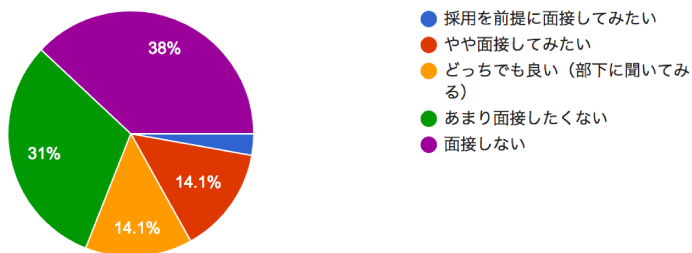


図 A.27 27 さんの評価結果

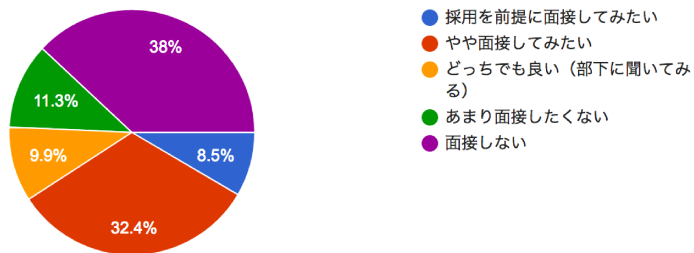


図 A.28 28 さんの評価結果

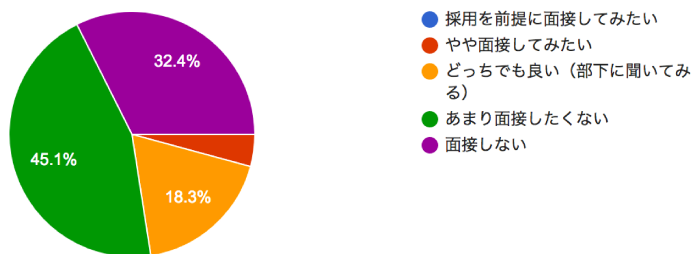


図 A.29 29 さんの評価結果

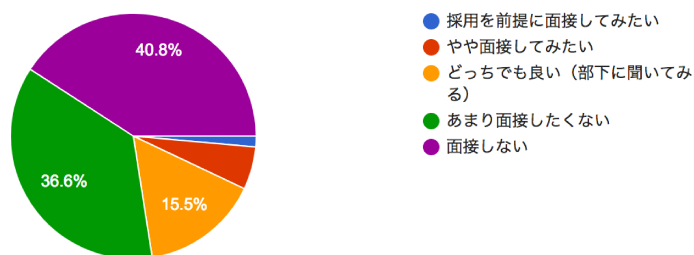


図 A.30 30 さんの評価結果

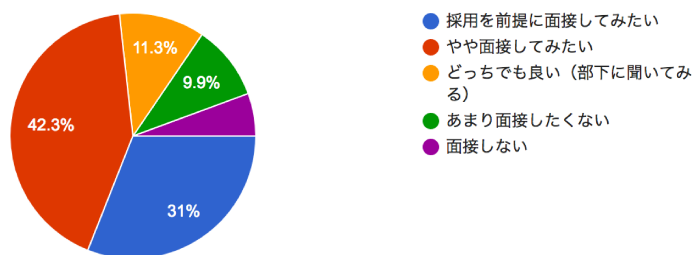


図 A.31 31 さんの評価結果

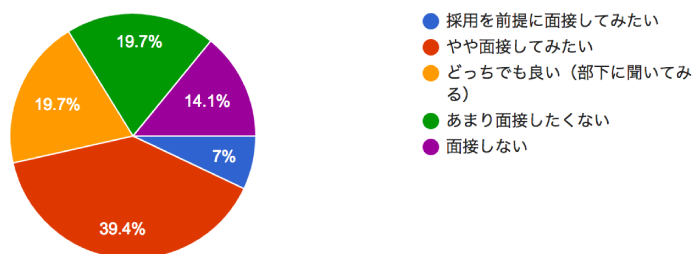


図 A.32 32 さんの評価結果

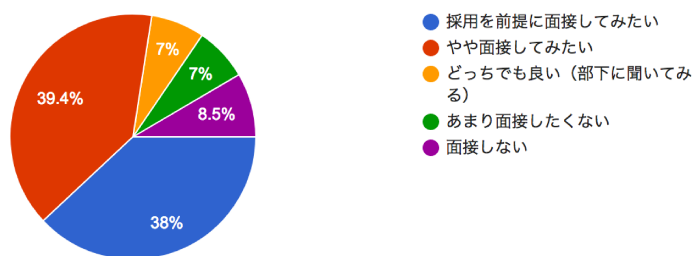


図 A.33 33 さんの評価結果

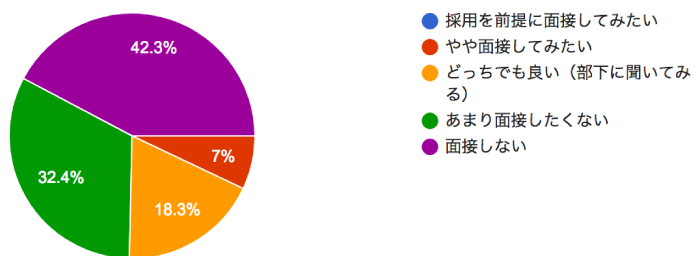


図 A.34 34 さんの評価結果

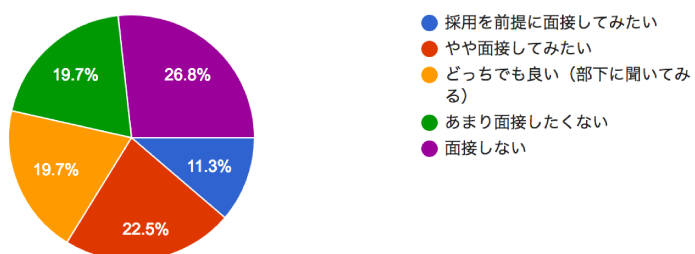


図 A.35 35 さんの評価結果

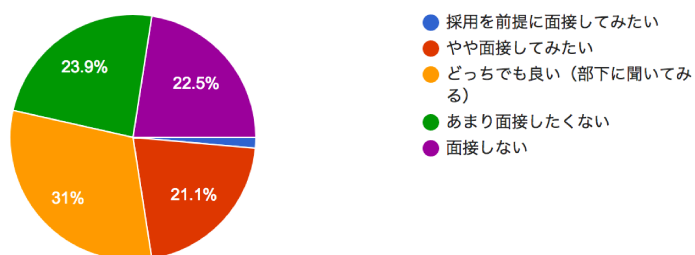


図 A.36 36 さんの評価結果

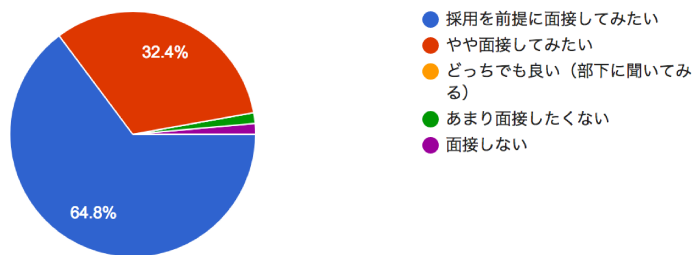


図 A.37 37 さんの評価結果

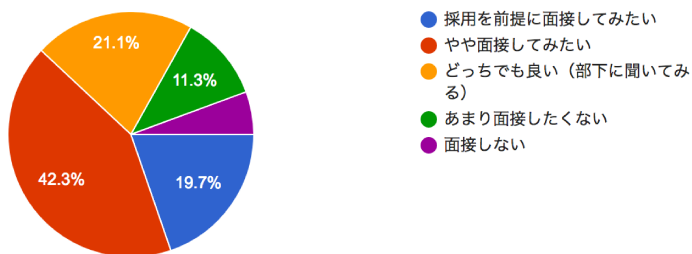


図 A.38 38 さんの評価結果

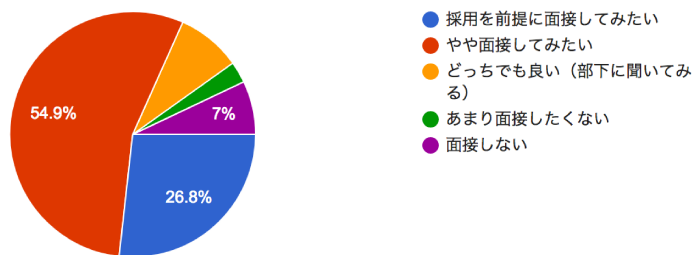


図 A.39 39 さんの評価結果

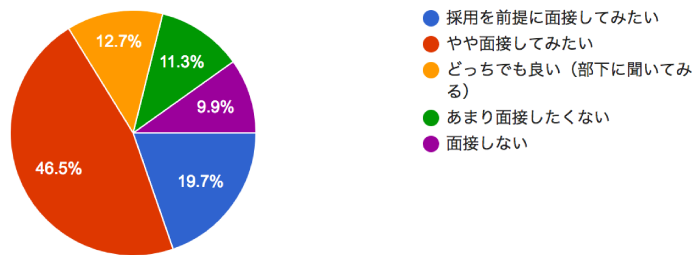


図 A.40 40 さんの評価結果