タスク介入によるクラウドワーカの品質推定精度の改善

松田　義貴† 　鈴木　優† 　中村　哲†

† 奈良先端科学技術大学院大学　情報科学研究科　〒 630–0101 奈良県生駒市高山町 8916-5
E-mail: †{matsuda.yoshitaka.mq2,ysuzuki,s-nakamura}@is.naist.jp

あらまし　クラウドソーシングでは，能力が無いワーカや意図的に不適切な作業を行うワーカが存在するため，作業から排除もしくは指導することができれば，作業結果の品質向上や金銭的や時間的なコストの低下につながる．そこで，重要な課題となるのが自動的なワーカの品質推定である．本研究では，ワーカの品質を正しい作業結果を納品する確率と定義する．近年，ワーカの振る舞いからワーカの品質を推定する研究が行われているが，ワーカの振る舞いを取得することが難しいタスクに適用することは困難である．この原因として，ワーカの振る舞いのパターンが少ないために，良質なワーカと悪質なワーカの間に振る舞いの差が見られないことが挙げられる．そこで本研究では，タスクに対してワーカの振る舞いを取得する仕組みを追加し，多くの振る舞いを取得することによって，ワーカの品質をより高精度に推定する手法を提案する．提案手法では，低品質なワーカを再現率 0.84 で検出することができた．一方で，処理時間やワーカの離脱率とはトレードオフの関係があった．

キーワード　クラウドソーシング，ワーカの品質，振る舞い，タスク設計

1. はじめに

マイクロタスク型のクラウドソーシングは，安価に大量のタスクを発注できるというメリットがある．一方で，依頼者が期待するような結果を必ずしも得ることができるとは限らない．依頼者が監視できない環境下で，不特定多数のワーカがタスクに従事するというクラウドソーシング特有の労働環境によって，依頼者が期待するような結果を納品することができないワーカが存在する．例えば，タスクの難易度や分野によっては正しい答えを導くことができないワーカがタスクに従事すると，正しい作業結果を依頼者は得ることができない．また，金銭目的など悪意を持って不適切な作業を意図的に行うワーカが存在することも知られている [1]．本研究では，正しい作業結果を納品する確率をワーカの品質とし，正しい作業結果を納品する確率が低いワーカを低品質ワーカと呼ぶ．

低品質ワーカがタスクに従事することを禁止したり指導したりすることによって，できる限り多くの高品質ワーカを雇用することで，誤った作業結果が生成される可能性が低くなる．そこで重要な課題となるのが，各ワーカが低品質ワーカか高品質ワーカかを推定することである．しかし，依頼者が作業結果を一つひとつ確認し，ワーカの品質を算出することはクラウドソーシングを行うメリットがなくなるため，自動的にワーカの品質を算出する仕組みが必要となる．

近年，ワーカの振る舞いからワーカの品質を推定するアプローチが注目されている [2–5]．ワーカの作業画面上における振る舞いのログをプログラムにより自動的に取得し，機械学習によって品質を推定するアプローチである．既存の研究では，振る舞いの分析方法ばかりに焦点が当てられてきた．しかし，振る舞いの取得に関する議論はされておらず，ワーカが回答のために行うマウスやキーボード操作などの振る舞いの一部しか扱われていないため，2 値分類などの単純なタスクには適用が困難である．なぜならば，取得できる振る舞いの種類が少ないと，ワーカの品質がその取得した振る舞いと関係があるとは限らないためである．さらに，少数の振る舞いだけでは優秀なワーカの品質を正確に判断できない．例えば，処理時間だけでワーカの品質を判断しようとすると，早く正確に処理できるワーカの品質を低く見積もってしまう可能性がある．

そこで本研究では，タスクに介入することによって今まで扱われてこなかった振る舞いを取得する手法を提案する．つまり，振る舞いを取得するための仕組みをタスクに導入し，ワーカの品質推定精度の向上を目指す．ベースラインと比較し，提案手法では，低品質ワーカの再現率が最低でも 12%向上した．また，タスク介入によって取得可能となったコンテンツの閲覧時間や回数などの新しい振る舞いがワーカの品質推定に重要な役割を果たしていることが分かった．一方で，処理時間やワーカの離脱率とはトレードオフの関係が確認できた．本研究で得られた知見は，ワーカの品質推定精度とワーカへの負担のバランスを考えた今後のシステム設計の指針となる．

2. 関連研究

本章では，ワーカの品質を推定する三つのアプローチを紹介する．ここでは特に，ワーカの振る舞いを用いた研究を紹介する．

ワーカの品質を推定する手法は 1) ゴールドタスクによる測定，2) 作業結果の分析による推定，3) 振る舞いの分析による推定の 3 種類の手法が提案されている．一つ目のゴールドタスクによる測定では，あらかじめ答えの分かっているゴールドタスクをワーカに課し，ワーカの品質を測定する [6]．個々のタスクに対して，ワーカの品質を正しく見極めるゴールドタスクを作成する必要があり，タスク依頼者の手間とコストが発生する．二つ目の作業結果の分析による推定では，冗長化によって複数の作業結果を獲得し，その作業結果からワーカの品質を推

定する [7,8]．冗長化とは，複数のワーカに対して同じ作業を割り当て，複数の作業結果を統合し，最終的な作業結果を導く手法である [9]．依頼者にとっては，冗長化させるワーカ数に比例して，金銭的や時間的なコストが増大してしまう．そして近年では，ワーカの振る舞いからワーカの品質を推定する研究がなされている [2–5]．ゴールドタスクの作成や，冗長化を必要としないため，手間やコストを増大させない手法であると言える．また，アンケートや自由記述のような答えが一意に定まらないタスクにも適用可能である．また，これら 3 つのアプローチは併用することが可能であり，競合することはない．

Zhu ら [2] は，クラウドソーシングにおける作業中の振る舞いはワーカによって異なることを示した．ワーカの振る舞いをワーカの品質推定に初めて用いたのは Rzeszotarski ら [3] である．Rzeszotarski らは，マウスやキーボード操作のタイミングや回数，処理時間などをワーカの振る舞いとして用いた．他にも，Hirth ら [4] は，ページのスクロールやラジオボタンのクリック間隔からワーカがどの問題を解いてるかを推定し，問題ごとの処理時間をワーカの振る舞いとして用いた．そして彼らは，ワーカの振る舞いを入力とした機械学習アルゴリズムによってワーカの品質を推定した．Kazai ら [5] は，充分に訓練されたワーカと通常のワーカの振る舞いに差があることを示した．さらに，ワーカの品質推定のための機械学習モデルの学習時に，訓練者の振る舞いを用いることによってワーカの品質推定精度を向上させた．

これらの既存の研究では，ワーカの振る舞いの扱い方や分析方法ばかりに焦点が当てられてきた．例えば，機械学習への入力となる振る舞いの特徴量設計の工夫や，学習方法が工夫されている．しかし，振る舞いの取得に関する議論はされていない．つまり，振る舞いを取得する工夫はなく，簡単に取得できる振る舞いだけを使用してワーカの品質が推定されている．その理由としては，豊富な種類の振る舞いが存在するタスクが研究対象となっているためだと考えられる．しかし，マウスカーソルの動きやページのスクロールがほとんど発生しない単純なタスクでは，取得される振る舞いの種類が限定され，既存の研究をそのまま適用することは困難である．既存の研究で取得している振る舞いは，マウスやキーボード操作など表に現れる振る舞いだけであるが，実際には問題文を読んだり考えたりしている時間がある．そこで本研究では，既存の研究では扱われなかった振る舞いに着目し，ワーカの品質推定に与える影響を検証する．

3. ベースライン

本章では，本研究で行ったクラウドソーシングのタスクについて説明する．さらに，既存の研究に倣ったワーカの品質を推定する手法を説明し，その手法を本研究でのベースラインとする．

3.1 タ ス ク

本節では我々が行なったクラウドソーシングのタスクについて説明する．タスク内容は，与えられたツイートが以下の 2 点を満たすかどうかの分類である．

- 京都の様子が分かる．
- その様子は京都観光において有益である．

ワーカは，はい，いいえ，ツイートが表示されないのいずれかを選択する．我々の構築したクラウドソーシングプラットフォームでは，ワーカの作業時に twitter からツイート内容を WebAPI により取得するため，ツイート投稿者や twitter 社がツイートを削除した場合，当該ツイートを作業画面に表示することができない．そのため，ツイートが表示されないという選択肢を用意した．ワーカはツイートを読み，三つの選択肢から答えを選択する．送信ボタンを押すと，次のツイートが表示され，この作業を繰り返す．

3.2 品質推定手法

本節では，3.1 節で説明したタスクにおいてワーカの振る舞いからワーカの品質を推定する手法について説明する．手法の概念や特徴量の設計は Kazai ら [5] の手法を可能な範囲で模倣している．ワーカの品質推定には教師あり学習を用いる．つまり，ワーカの振る舞いを入力としてワーカの品質を出力する分類器を構築する．本研究では，分類の対象となるワーカを 100 回以上のタスク処理をしたワーカに限定する．

3.2.1 分類器の入力と出力

最初に，分類器への入力となるワーカの振る舞いと特徴量の抽出について説明する．我々はワーカのタスク処理をプログラムにより自動的に監視し，表 1 に示すワーカの振る舞いを取得する．これらの振る舞いはワーカが一つのツイートを分類するごとに取得する．つまり，100 回ツイートを分類したワーカにはそれぞれの振る舞いが 100 個ずつ存在する．ワーカごとに取得したそれぞれの振る舞いの平均値，中央値，最大値，最小値，標準偏差，エントロピーをそれぞれ計算し，ワーカの特徴量とする．さらに，ワーカ w のタスク総処理回数 N_w と総処理時間 T_w も特徴量として用いる．つまり，ワーカ w の振る舞い B_i の統計量 x_i を式（1）のように表すと，ワーカ w の特徴量 $\boldsymbol{x_w}$ は式（2）で表すことができる．

$$\boldsymbol{x_w^i} = [\, B_i^{Ave}\ B_i^{Med}\ B_i^{Min}\ B_i^{Max}\ B_i^{Std}\ B_i^{Ent}\,], \quad (1)$$

$$\boldsymbol{x_w} = \mathrm{concat}[\, N_w\ T_w\ \boldsymbol{x_w^1}\ \boldsymbol{x_w^2}\ \boldsymbol{x_w^3}\ \boldsymbol{x_w^4}\ \boldsymbol{x_w^5}\,] \quad (2)$$

次に，分類器の出力となるワーカの品質について説明する．ワーカ w の品質 Q_w は低品質（-1）か高品質（1）の 2 値とする．つまり，特徴量 $\boldsymbol{x_w}$ から品質 Q_w を推定する．これは，式（3）における f を機械学習によって推定することを意味する．ここで，$\boldsymbol{x}$ と Q はそれぞれ任意のワーカの特徴量と品質である．

$$f : \boldsymbol{x} \mapsto Q \in \{-1, 1\} \quad (3)$$

3.2.2 分類器の構築

分類器 f の構築方法について説明する．本手法では教師あり学習を用いているため，事前に複数ワーカの振る舞いを取得し，品質を算出しなければならない．まず，各ツイートに対して n 人のワーカを割り当て，多数決で各ツイートの正解ラベルを付与することによって，式（4）で表される正答率 r_w を求める．本研究では $n = 10$ とした．

表 1: 取得するワーカの振る舞い

ID	振る舞い	説明	サンプル
B_1	処理時間	作業ページがロードされてから送信ボタンを押すまでの時間（秒）	5, 10, 60
B_2	1 文字あたりの処理時間	処理時間をツイートの文字数で割った値（秒）	0.05, 0.1, 1
B_3	回答時間	作業ページがロードされてから回答を選択するまでの時間（秒）	5, 10, 60
B_4	1 文字あたりの回答時間	回答時間をツイートの文字数で割った値（秒）	0.05, 0.1, 1
B_5	回答回数	回答を変更した回数（回）	1, 2, 5

$$r_w = \frac{M_w}{N_w} \times 100 \tag{4}$$

ここで，M_w はワーカ w の総タスク処理回数のうち正しい答えを選んだ回数である．そして，ワーカ w の品質 Q_w を式 (5) で定義する．β は下位 $\alpha\%$ に位置するワーカの正答率である．

$$Q_w = \begin{cases} -1 & (r_w \leqq \beta) \\ 1 & (r_w > \beta) \end{cases} \tag{5}$$

分類器にはランダムフォレスト [10] を用いる．本研究において特徴量となるワーカの振る舞いには外れ値だと考えられる値が存在する．例えば，ワーカが作業画面を開いたまま作業を放置した場合，処理時間が 1 時間を超えることがある．ランダムフォレストはこのような外れ値に対して影響を受けにくいという性質がある [11]．さらに，ワーカの品質に関係がない振る舞いが特徴量に含まれていても影響を受けにくい [11]．ランダムフォレストのハイパーパラメータはグリッドサーチによって決定した．

サンプル数が不均衡なデータをそのまま学習に用いた場合，全てのワーカが多数派のクラスに分類されてしまう可能性がある．そこで，SMOTE アルゴリズム [12] を用いて比率が 1:1 になるように少数クラスのサンプルを仮想的に増やすことにより，学習サンプル数の不均衡を解消する．

4. 提案手法

本章では，ワーカの品質を推定する提案手法について説明する．まず，提案手法となるタスク介入アイデアについて述べる．次に，タスク介入の具体的な手法について説明する．最後に，ワーカの品質を推定する手法について説明する．

4.1 アイデア

本研究では，既存の研究では扱われていないワーカの振る舞いに着目する．まず，3.1 節で説明したツイート分類タスクにおける作業の流れを考える．ワーカの作業の流れの例を図 1 に示す．タスクを解くことだけを考えると，ワーカ A のようにツイートを閲覧し，回答するという流れになる．つまり，作業は 1) 閲覧，2) 回答の二つに分けることができる．従来の研究では，マウスやキーボード操作が表に現れる回答部分と全体的な処理時間が主な振る舞いとして扱われてきた．しかし，ツイート分類タスクのような単純なタスクでは，回答のために必要な操作の回数や種類が少なく，回答部分の振る舞いだけでは低品質なワーカと高品質なワーカの振る舞いに差異が出にくいことが考えられる．差異があったとしても，少数の種類の振る舞いだけではワーカの品質推定を誤る可能性が大きくなることが考えられる．例として，処理時間だけでワーカの品質を推定する場合を考える．意図的に不適切な作業を行う低品質なワーカは，ツイートをほとんど読まずに回答だけ行うことが予想される．そのため，処理時間は短くなる．また，慎重な作業を行う高品質なワーカの処理時間は長くなることが考えられる．しかし，早く正確に処理できる非常に高品質なワーカの処理時間は短くなる．このような非常に高品質なワーカの振る舞いは低品質なワーカの振る舞いと似ているため，品質を低く見積もる可能性がある．

図 1: ワーカの作業例

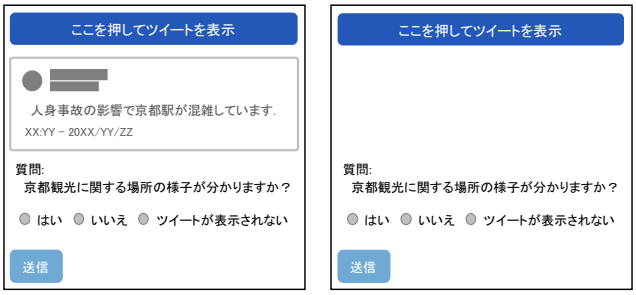
クラウドソーシングでは，誰にも監視されていない状況でインターネット上で作業ができるという特徴から，ワーカ B のように他のことを行いながら作業が行われている場合があることが予想される．例えば，パソコンで他の作業を行ったりテレビを見ながら作業を行ったりしていることが考えられる．そのため，作業ページが開かれてから作業結果を提出するまでの時間は必ずしも実際に作業を行った時間と一致しない．したがって，より正確な作業時間を獲得することが必要であり，閲覧部分の振る舞いを取得することがその一つとなり得る．

また，注意深いワーカはワーカ C のように回答を選択した後，ツイートをもう一度読み直すかもしれない．しかし，ワーカ A や B のように読み直さないワーカもいることが予測される．閲覧部分の振る舞いを取得することができれば，このような振る舞いの差も考慮することができる．

以上のようなことから，閲覧部分の振る舞いを取得することができれば，より高精度にワーカの品質を推定することが可能であると考えた．そこで，閲覧部分の振る舞いを取得する仕組みをタスクに導入することが本研究の提案手法である．本研究では，閲覧部分の振る舞いとして閲覧時間と閲覧回数を取得することをタスク介入によって実現する．従来の研究では扱われてこなかったこのような振る舞いがワーカの品質推定に与える影響を検証することが本研究の主題となる．

4.2 タスク介入手法

閲覧時間と閲覧回数を取得するための具体的なタスク介入手



(a) ボタンを押していない時　　**(b)** ボタンを押している時

図 2: 提案タスクのスクリーンショット

法について説明する．タスクの内容はツイートの分類であり，ベースラインと変わらない．しかし，提案手法では，初期状態ではツイートを表示しない．その代わりにツイートを表示させるためのボタンを設置する．作業画面を図 2 に示す．図 2b はツイート表示ボタンを押していない時の作業画面で，図 2a はツイート表示ボタンを押している時の作業画面である．ワーカがこのボタンを押下している時だけツイートが表示され，ボタンを離せばツイートは非表示となる．そうすることで，ツイートの閲覧時間や閲覧回数が取得可能となる．

4.3 品質推定手法

提案手法では，表 1 の振る舞いに加えて表 2 に示す振る舞いを取得する．ベースラインと同様に，各振る舞いの統計量をランダムフォレストの特徴量として用いて，ワーカの品質を推定する．つまり，ワーカの特徴量として用いる振る舞いだけがベースラインと異なり，特徴量の設計やワーカの品質推定は同様に行う．提案手法におけるワーカ w の特徴量 $\boldsymbol{x_w}$ は式（6）で表すことができる．

$$\boldsymbol{x_w} = \mathrm{concat}[\, N_w \;\; T_w \;\; \boldsymbol{x_w^1} \;\; \boldsymbol{x_w^2} \;\; \boldsymbol{x_w^3} \;\; \cdots \;\; \boldsymbol{x_w^{11}} \,] \qquad (6)$$

5. 評価実験

本章では，提案手法におけるタスク介入によって取得した振る舞いがワーカの品質推定に有益であるか検証するための評価実験について述べる．また，ワーカの品質推定とボタンを設置しタスクに介入したことによるワーカへの影響の関係を調査する．

5.1 実験手順

我々は独自のプラットフォームを作成し，ワーカは我々のプラットフォーム上で作業を行う．また，Javascript と jQuery ライブラリを使用してワーカの振る舞いを取得する．ワーカの募集と作業報酬の支払いは既存のクラウドソーシングのプラットフォーム（Crowdworks (注1)）を通して行う．

我々のタスクに応募したワーカは全員雇用する．つまり，我々は雇用するワーカを事前に選別はしない．

ワーカは 100 ツイート分類するごとに 30 円の作業報酬を得ることができる．平均的なワーカは 1 時間に約 450 ツイート分類した．つまり，ワーカは 1 時間で 135 円稼ぐことができる．これは，標準的なクラウドソーシングの賃金（1.38 ドル/時間）とほとんど等しい [13]．

ワーカは自身のタイミングで作業の中断や再開，離脱することができる．つまり，作業を行ったツイート数が 100 個以下であったため報酬を得ずに離脱したワーカや，1 万ツイート以上のタスクに従事したワーカも存在する．

5.2 収集データの概要

提案手法とベースライン手法で我々が雇用したワーカの数やツイートの分類総数などについて表 3 にまとめる．ここで，総ワーカ数とは我々のタスクに応募し 1 回でもタスクに従事したワーカの総数のことを指し，対象ワーカ数とは 100 回以上タスクに従事した本実験で扱うワーカを指す．総タスク処理回数とは，全てのワーカでのタスクの処理回数である．また，我々はこの全てのタスク処理においてワーカの振る舞いを取得している．ツイート数とはワーカが一人でも分類を行ったツイート数である．我々は一つのツイートに対して 10 人のワーカを割り当てた．ただし，ツイートが削除され，作業画面に表示できなくなった場合は，ワーカの割り当てを取りやめた．

5.3 ワーカの品質推定

本節では，ベースライン手法と提案手法それぞれのワーカの品質推定結果について説明する．また，提案手法でのツイート表示ボタンによって取得可能となった振る舞いの有効性を検証する．さらに，正しく推定できたワーカとできなかったワーカの性質を分析する．

5.3.1 推定結果

ランダムフォレストを用いて，ワーカの品質が高品質であるか低品質であるかの分類を 5 分割交差検証で評価した．すなわち，本実験で用いるデータを 5 分割し，四つのデータを学習データとして用い，残りの一つのデータで評価することを 5 回繰り返した．

まず，$\alpha = 10$ としてワーカの品質を推定した．すなわち，正答率が下位 10%のワーカを低品質ワーカ，それ以外のワーカを高品質ワーカと定義する．この時，低品質ワーカと高品質ワーカの境界となる正答率 β はベースライン手法で 65.5%，提案手法で 62.7%となった．

ベースライン手法と提案手法それぞれにおいて，5 回分の分類結果をそれぞれまとめた混同行列を表 4a，4b に示す．また，表 4c には，用いる振る舞いをベースライン手法と同じ振る舞いに限定にした場合の提案手法での分類結果も示す．以後，用いる振る舞いをベースライン手法と同じ振る舞いに限定にした場合の提案手法のことを限定的提案手法と呼ぶ．この分類結果を用いてワーカの排除や指導を行うことを考えた時，排除や指導に該当する低品質ワーカをできるだけ見逃さないことが重要である．なぜならば，低品質ワーカは作業結果の品質を低下させる原因だからである．つまり，低品質ワーカの再現率が重要であると言える．ベースライン手法では低品質ワーカの再現率が 0.72 であった．一方で，提案手法では 0.84 に向上した．また，限定的提案手法では 0.64 であり，提案手法が低品質ワーカをより高精度に検知することができた．また，適合率や F 値を見ても，提案手法が最も良い精度であった．しかし，高品質

(注1)：https://www.crowdworks.jp

表 2: 追加で取得するワーカの振る舞い

ID	振る舞い	説明	サンプル
B_6	閲覧回数	ツイート表示ボタンを押した回数（回）	1, 2, 5
B_7	閲覧時間	ツイート表示ボタンを押している間の合計時間（秒）	5, 10, 60
B_8	1 文字あたりの閲覧時間	閲覧時間をツイートの文字数で割った値（秒）	0.05, 0.1, 1
B_9	1 回あたりの閲覧時間	閲覧時間を閲覧回数で割った値（秒）	5, 10, 60
B_{10}	1 文字 1 回あたりの閲覧時間	1 回あたりの閲覧時間をツイートの文字数で割った値（秒）	0.05, 0.1, 1
B_{11}	閲覧割合	処理時間のうちの閲覧時間の割合	0.1, 0.5, 0.8

表 3: 収集したデータの規模

	総ワーカ数（人）	対象ワーカ数（人）	総タスク処理回数（回）	ツイート数（個）
ベースライン手法	439	250	132,594	–
提案手法	486	255	108,836	–
合計（重複無し）	793	439	241,430	26,713

ワーカを低品質と予測するケースが多く，改善の余地がある．

同様に，低品質ワーカを下位 5%と 15%としてワーカの品質の推定を行った．表 5 に β の値をまとめる．また，それぞれの β でそれぞれの手法で分類した時の分類結果を表 6 に示す．下位 5%と 15%のワーカを低品質ワーカと定義した時にも，提案手法が再現率だけでなく，適合率や F 値においても最も良い精度であった．これは，α によらず提案手法が効果的であることを示している．しかし，下位 10%のワーカを低品質ワーカと定義した時と同様に，高品質ワーカを低品質と予測するケースが多く存在した．

5.3.2 重要な振る舞い

ランダムフォレストでは，特徴量の重要度を計算することができる．本実験では 5 分割交差検証を行ったため，5 回の分類それぞれで重要度が計算される．ここでは，5 回の重要度の平均値をその特徴量の重要度として用いる．提案手法における重要な特徴量の上位 5 件を図 3 に示す．α の値，すなわち，低品質ワーカと高品質ワーカの境界に関わらず，重要度が高い特徴量の多くはタスクに介入したことによって得ることができた振る舞いの特徴量である．つまり，提案手法でなければ取得できない振る舞いが分類に重要な役割を果たしたと言える．

例えば，1 回あたりの閲覧時間の最小値は 3 種類の分類すべてで重要な振る舞いの上位 3 件に入った．1 回あたりの閲覧時間の最小値と正答率の関係を図 4 に示す．一つひとつの点はワーカを表している．1 回あたりの閲覧時間の最小値が大きいワーカは正答率が高い傾向がある．そのため，1 回あたりの閲覧時間の最小値から高品質なワーカを検出することは容易である．しかし，1 回あたりの閲覧時間の最小値が小さいからと言って正答率が低いとは限らないが，正答率が低いワーカは概ね 1 回あたりの閲覧時間の最小値が小さい．そのため，低品質なワーカを検知することもできるが，高品質なワーカを低品質と誤判別してしまう．これは，表 4 の混同行列にも表れている．

また，閲覧回数も分類に重要な役割を果たしていることが図 3 から分かる．高品質なワーカは回答を選択した後にもツイートを閲覧し直すなど，複数回閲覧していると考えられる．

5.3.3 推定結果の分析

提案手法によって正しく分類できたワーカと，提案手法でも正しく推定できなかったワーカの特徴を分析する．まず，最も正答率が悪いワーカに着目し，このワーカをワーカ D と呼ぶことにする．ワーカ D は限定的提案手法では高品質ワーカと誤って分類されたが，提案手法では低品質ワーカと正しく分類された例である．ワーカ D の処理時間の平均は約 6 秒であり，平均的なワーカに比べ処理時間が短い．しかし，5.3.2 項にも述べたのと同様に，処理時間が短くても高品質なワーカも存在する．そのため，ワーカ D を低品質なワーカだと断定することは難しい．しかし，ワーカ D は 9 割以上のタスクで 1 回もツイートを閲覧していない．さらに，閲覧した場合でも閲覧時間は 2 秒に満たず，ワーカ D は明らかに適切な作業をしているとは言えない．このように，提案手法によって取得した閲覧時間や閲覧回数によって低品質ワーカであると判断することが可能になったと考えられる．

次に，2 番目に正答率が悪いワーカに着目する．このワーカをワーカ E と呼ぶことにする．ワーカ E は提案手法でも高品質ワーカと誤って分類された例である．ワーカ E の閲覧回数は平均 1.5 回以上であり，閲覧時間の平均も約 12 秒である．平均的なワーカの閲覧時間の約 6.5 秒であり，約 2 倍ツイートを読んでいることになる．ワーカ E の正答率は低いが，振る舞いが不適切であるとは言い難い．そのため，提案手法でも高品質ワーカと分類してしまったと考えられる．ワーカ E は真面目に作業をしているが，タスクの内容を正確に理解できていない可能性がある．本手法では振る舞いのみでワーカを判断するため，このようなワーカを低品質ワーカと判断することはできなかった．正答率が悪い低品質なワーカを検知するためには，振る舞いだけでなく，作業結果も見る必要があると考えられる．

5.4 提案手法が及ぼすワーカへの影響

提案手法では，ツイートを読むためにボタンを押し続ける作業をワーカに対して課しているが，この作業自身はタスクの遂行に直接関係無い．これはワーカの品質推定のために行ったことであるが，作業結果の品質を低下させることや，ワーカが集まらないなどの問題があっては意味がない．そこで本節では，

表 4: 混同行列

(a) ベースライン手法

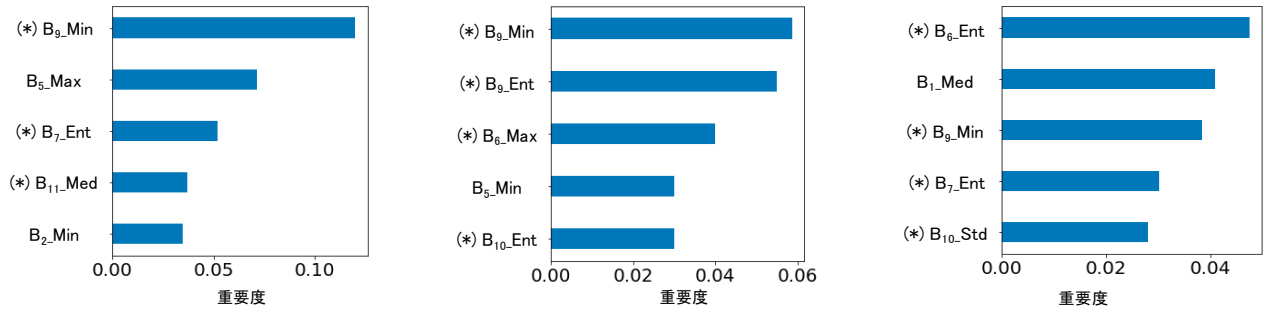
		予測	
		低品質	高品質
正解	低品質	18	7
	高品質	69	156

(b) 提案手法

		予測	
		低品質	高品質
正解	低品質	21	4
	高品質	72	158

(c) 限定的提案手法

		予測	
		低品質	高品質
正解	低品質	16	9
	高品質	65	165



(a) $\alpha = 5$ **(b)** $\alpha = 10$ **(c)** $\alpha = 10$

図 3: 重要な特徴量の上位 5 件 （*は提案手法で新たに取得した振る舞い）

表 5: 低品質ワーカと高品質ワーカの境界となる正答率

α	手法	境界となる正答率 β
5%	ベースライン	65.5%
	提案手法	62.8%
10%	ベースライン	73.0%
	提案手法	72.6%
15%	ベースライン	80.7%
	提案手法	78.7%

表 6: 分類結果

α	手法	適合率	再現率	F 値
5%	ベースライン	0.17	0.55	0.26
	提案手法	**0.21**	**0.75**	**0.33**
	限定的提案手法	0.16	0.58	0.25
10%	ベースライン	0.21	0.72	0.32
	提案手法	**0.23**	**0.84**	**0.36**
	限定的提案手法	0.20	0.64	0.30
15%	ベースライン	0.22	0.62	0.33
	提案手法	**0.32**	**0.79**	**0.45**
	限定的提案手法	0.27	0.71	0.39

提案手法がワーカに与える影響についての分析結果の説明と議論を行う．ワーカへの影響として，タスクの難易度，処理時間とモチベーションの三つの観点から評価する．

5.4.1 タスクの難易度

提案手法のタスクとベースライン手法のタスクでの正答率の違いと分布からタスクの難易度を評価する．まず，両方のタスクに従事した対象ワーカ 66 人を対象として，提案手法とベースライン手法でのそれぞれの正答率を図 5 に示す．横軸がベースライン手法での正答率，縦軸が提案手法での正答率を表し，一つひとつの点が各ワーカを表している．相関係数は 0.78 となっており，強い正の相関が見られる．

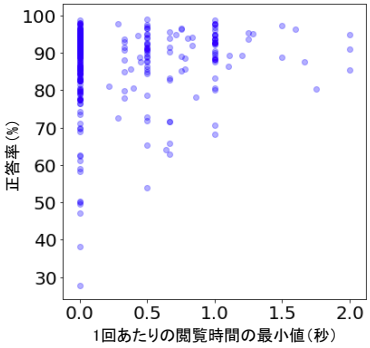


図 4: 1 回あたりの閲覧時間の最小値と正答率の関係

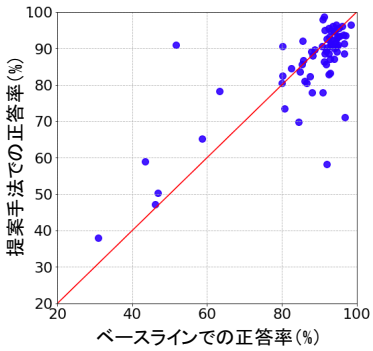


図 5: 両方のタスクに従事したワーカの正答率

さらに，両方のタスクに従事したワーカの正答率の分布を図 6 に示す．横軸が正答率，縦軸がその正答率のワーカ数である．提案手法とベースライン手法でのタスクのそれぞれの正答率をウィルコクソンの符号順位検定を用いて検定したが有意な差は認められなかった（$p > 0.05$）．つまり，タスクの難易度が異なるとは言えない．

提案手法とベースライン手法での正答率には強い正の相関が

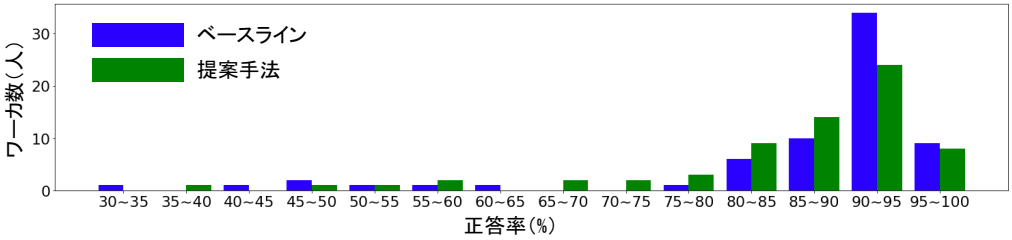


図 6: 両方のタスクに従事したワーカの正答率の分布

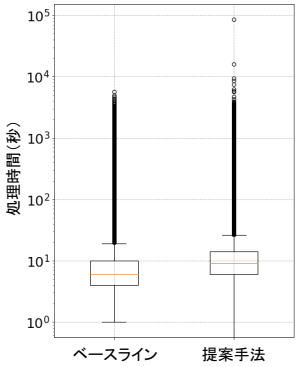


図 7: タスク処理時間の分布

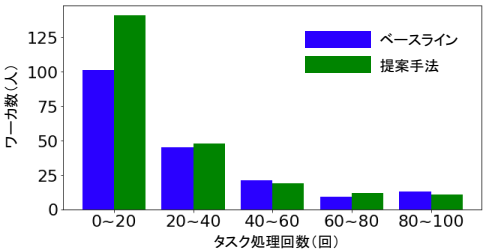


図 8: ワーカの離脱のタイミング

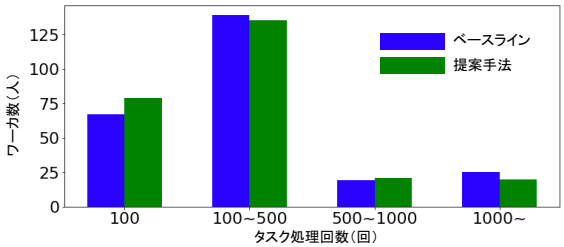


図 9: 対象ワーカのタスク処理回数の分布

あり，有意差が認められないことから，タスクの難易度の差は無い．さらに，正答率は，提案手法とベースライン手法のユーザインタフェースによって差がないことから，本実験のタスクにおけるワーカの品質と言うことができる．つまり，ワーカの品質はユーザインタフェースに依存するものではない．

5.4.2 処 理 時 間

提案手法では，ツイートを読むためにはツイート表示ボタンを押し続けなければならないため，ツイートを読む度にマウスカーソルをツイート表示ボタンへ移動させなければならない．その結果，ベースライン手法と比べ多くの処理時間を要することが予想される．

それぞれの手法の処理時間の分布の箱ひげ図を図 7 に示す．グラフの縦軸は処理時間であるが，対数目盛となっている．ベースライン手法では処理時間の中央値が 6 秒であった．対して，提案手法では 9 秒となった．我々の予想通り，提案手法では処理時間が 3 秒増加し，ワーカへ時間的な負担を与えることが分かった．さらに，タスク依頼者にとっても依頼する全てのタスクが処理されるまでにより多くの時間が必要となる．

また，ベースライン手法と提案手法の双方で，処理時間が 1,000 秒を超えるようなケースも存在する．Twitter の文字数は最大 140 字であり，これほど多くの時間が必要であるとは考えにくい．これは 4. 章で述べたように，ワーカは常にタスクに集中している状況ではなく，他のことを行いながら作業を行っていたり，休憩を挟んでタスクに取り組んでいたりすることが表れている．

5.4.3 モチベーション

ワーカのモチベーションを測定する指標として，作業単位である 100 ツイートに満たず作業から離脱した割合と，作業をどれだけ続けたかの処理回数を用いる．まず最初に，作業からの離脱について議論する．表 3 の総ワーカ数から対象ワーカ数を引いた人数が離脱したワーカの人数となる．つまり，ベースライン手法では 43.1%の 189 ワーカが途中で作業から離脱した．一方で，提案手法では，47.5%の 231 ワーカが途中で作業から離脱した．提案手法では 4.4%離脱率が増加した．

次に，離脱したタイミングを図 8 に示す．横軸がタスク処理回数，縦軸がそのタスク処理回数で離脱したワーカ数である．作業報酬額は同じであったが，提案手法では 10 回以内に離脱するワーカがベースライン手法よりも明らかに多いことが分かる．

図 9 に対象ワーカのタスク処理回数の分布を示す．提案手法とベースライン手法ではタスクの処理回数の分布には大差がない．以上の結果をまとめると，提案手法では作業開始直後にワーカが離脱してしまう最初のハードルを越えるとベースライン手法と同様にワーカを確保することが可能であると言える．

6. おわりに

本論文では，クラウドソーシングの単純なタスクにおいて，ワーカの品質を振る舞いから推定する手法を提案した．既存の研究では振る舞いの分析に焦点が当てられてきたが，我々は振る舞いの取得に焦点を当てた．提案手法では閲覧時間や閲覧回数などのより正確な振る舞いを取得するために，コンテンツを

表示するためのボタンを設置した．提案手法では，適合率を下げることなく再現率を向上させ，低品質ワーカをより高精度に検出することができた．特に，低品質ワーカの定義を正答率が下位 10%のワーカとした時，再現率は最も高い 0.84 となった．また，1 回あたりの閲覧時間や閲覧回数など，ボタンの設置によって取得可能となった振る舞いがワーカの分類に重要な役割を果たしていることが分かった．ボタンを設置したことによるデメリットとしては，処理時間が増えることや，ワーカの離脱率が上がってしまうことが挙げられる．

今後の課題としては，以下のようなことが挙げられる．一つ目の課題は，高品質なワーカを低品質と分類してしまう例が多いことや十分まじめに作業しているが正答率が悪いワーカを高品質と分類してしまうことである．つまり，素早く適切な作業を行えるワーカやタスクの内容を正しく理解していないワーカを誤判別しないように，振る舞いだけでなく作業結果を含めてワーカの品質を判断するなどの工夫が必要である．二つ目の課題は，提案手法の汎用性を検証することである．本論文では，一種類のタスクに対して実験を行った．我々は翻訳タスクや記事執筆タスクなどは対象外としているが，あるコンテンツを見て簡単な評価を与えるようなタスクには提案手法を適用することが可能であると考えている．例えば，画像のラベリングやアンケートなどが挙げられる．また，本研究で行ったタスクは多くのワーカの正答率が 90%程度あり，容易であった．ワーカの能力によって正答率にばらつきが出るタスクや全体的な正答率が低くなるタスクでも検証するべきである．三つ目の課題は，他のタスク介入方法を検討することである．提案手法ではコンテンツを表示するボタンを設置したが，より正確な振る舞いが取得でき，よりワーカに負担が少ない方法を検討することを今後の課題とする．

謝辞　本研究成果は，国立研究開発法人情報通信研究機構の委託研究により得られたものです．

文　献

[1] Aniket Kittur, Ed H Chi, and Bongwon Suh. Crowdsourcing user studies with mechanical turk. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 453–456. ACM, 2008.

[2] Dongqing Zhu and Ben Carterette. An analysis of assessor behavior in crowdsourced preference judgments. In *SIGIR 2010 workshop on crowdsourcing for search evaluation*, pp. 17–20, 2010.

[3] Jeffrey M Rzeszotarski and Aniket Kittur. Instrumenting the crowd: using implicit behavioral measures to predict task performance. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pp. 13–22. ACM, 2011.

[4] Matthias Hirth, Sven Scheuring, Tobias Hoßfeld, Christian Schwartz, and Phuoc Tran-Gia. Predicting result quality in crowdsourcing using application layer monitoring. In *Communications and Electronics (ICCE), 2014 IEEE Fifth International Conference on*, pp. 510–515. IEEE, 2014.

[5] Gabriella Kazai and Imed Zitouni. Quality management in crowdsourcing using gold judges behavior. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pp. 267–276. ACM, 2016.

[6] Gabriella Kazai, Jaap Kamps, Marijn Koolen, and Natasa Milic-Frayling. Crowdsourcing for book search evaluation: impact of hit design on comparative system ranking. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, pp. 205–214. ACM, 2011.

[7] Vikas C Raykar, Shipeng Yu, Linda H Zhao, Anna Jerebko, Charles Florin, Gerardo Hermosillo Valadez, Luca Bogoni, and Linda Moy. Supervised learning from multiple experts: whom to trust when everyone lies a bit. In *Proceedings of the 26th Annual international conference on machine learning*, pp. 889–896. ACM, 2009.

[8] Alexander Philip Dawid and Allan M Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied statistics*, pp. 20–28, 1979.

[9] Padhraic Smyth, Usama M Fayyad, Michael C Burl, Pietro Perona, and Pierre Baldi. Inferring ground truth from subjective labelling of venus images. In *Advances in neural information processing systems*, pp. 1085–1092, 1995.

[10] Leo Breiman. Random forests. *Machine learning*, Vol. 45, No. 1, pp. 5–32, 2001.

[11] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, Vol. 1. Springer series in statistics New York, 2001.

[12] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority oversampling technique. *Journal of artificial intelligence research*, Vol. 16, pp. 321–357, 2002.

[13] John Joseph Horton and Lydia B Chilton. The labor economics of paid crowdsourcing. In *Proceedings of the 11th ACM conference on Electronic commerce*, pp. 209–218. ACM, 2010.