3K3-OS-20a-1in

クラウドソーシングにおける階層的分類タスクの品質管理手法

Quality Control for Crowdsourced Hierarchical Classification

大谷直樹　馬場雪乃　鹿島久嗣
Naoki Otani　Yukino Baba　Hisashi Kashima

京都大学大学院情報学研究科知能情報学専攻
Department of Inteligenec Science and Technology, Graduate School of Informatics, Kyoto University

In this study, we focus on crowdsourcing, which has become widely used to annotate large datasets. One of the major concerns in the use of crowdsourcing is the quality of results. As crowdsourcing workers are non-experts, they may have a large variance in their abilities and motivation. Therefore labels annotated by them are often unreliable. To cope with this problem, one of the widely adopted approaches is to make redundancy, which is to request a single task to multiple workers and aggregate their responses to obtain more reliable annotations. The main aim of this research is to develop an accurate aggregation method for hierarchical classification tasks on crowdsourcing. Though it is a typical classification task, there is no method to exploit its hierarchical structure for controlling the quality of labels. By using the structure information, it is expected to reduce the complexity of classification and enable us to infer the true labels more accurately. This paper gives a novel method to model worker's annotation process in hierarchical classification tasks. We demonstrate that our methods aggregate labels more effectively than existing methods on real crowdsourced hierarchical classification tasks.

1　序論

クラウドソーシングとは、主にインターネット上で不特定多数のワーカーに作業を依頼するプロセスである。クラウドソーシングは、人間の認知能力を大規模かつ比較的簡単に活用する道を開いた。例えば、従来少数の専門家によって行われていた画像や文書のアノテーションがクラウドソーシングで行われるようになっている [Callison-Burch 10]。また、計算機にとって困難な問題を解くためのアプローチとして注目を集めており、計算量に課題の大きいタンパク質の立体構造予測問題が、クラウドソーシングで行われ成功を収めた例がある [Cooper 10]。一方で、クラウドソーシングでは計算機と異なりいつも正確な結果が得られるとは限らない。

品質管理はクラウドソーシングにおける重要なテーマである。クラウドソーシング上のワーカーは専門家とは異なり能力や意欲に幅がある。したがって得られる結果はしばしば信頼性を欠く。そこで、安定して高品質な結果を得るための仕組みが求められる。広く採用されているのが、単一のタスクを複数のワーカーに依頼して冗長性を上げるというアプローチである。集めた回答群を統合して正解を推定する方法には、多数決のほか、ワーカーの回答プロセスをモデル化して真のラベルを推定する手法も数多く提案されてきた。例えば、能力の高いワーカーの意見を重視する方法がある [Dawid 79, Whitehill 09, Welinder 10]。

本研究では、階層分類タスクにおける品質管理手法を提案する。階層分類は物や情報を整理する上で有用かつ一般的な方法である。例として、図書館の図書分類やオンラインマーケットの商品分類がある。階層分類はクラウドソーシングの典型タスクの 1 つである。タスク設計 [Bragg 13] やプロセス設計 [Kamar 15] に関する研究はあるが、分類の階層構造に焦点を当てた回答統合手法はこれまで提案されてこなかった。階層分類は多クラス分類と見なせ、いくつかの既存手法で回答の統合を行うことができる。本研究はその手法を階層情報を用いるよう拡張する。これにより多数のクラスの複雑性を減らすことができ、より効果的に統合が行えると期待できる。

連絡先: 大谷直樹, otani.naoki.65v@st.kyoto-u.ac.jp

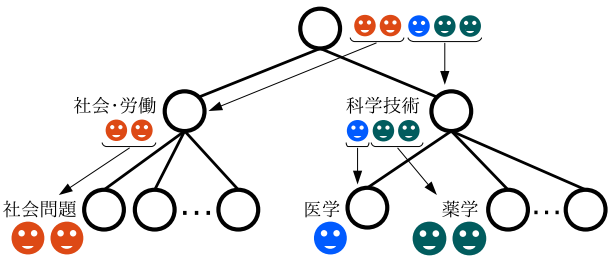


図 1: 階層分類タスクの例: 10 人のワーカーが 1 冊の図書を分類する。

クラス数が非常に多い図書分類タスクを例に取る。5 人のワーカーにある図書の分類を依頼し、そのうちの 2 人が「社会問題」2 人が「医学」に残りの 1 人が「薬学」に分類したとする (図 1)。多数決を取ると「社会問題」と「医学」が同じ信頼性を持つことになる。一方、親クラスを見れば「科学技術」に 3 票「社会・労働」に 2 票が与えられている。ゆえに、「科学技術」の子である「医学」がよりふさわしいと予想できる。

この洞察に基づき、本論文では階層構造を加味した結果統合手法を提案する。階層分類でのワーカーの回答プロセスをモデリングするために、項目反応理論 (Item Response Theory; IRT) の Steps モデルのアイデアを用いる。1 つの階層分類作業を小問が連なるテスト問題と見なし、分類作業が成功する確率をワーカーの能力と小問の難しさの関数として表現する。本研究は Whitehill らが提案した GLAD [Whitehill 09] を階層分類に拡張した形になるため、我々は提案モデルを Steps GLAD と名付ける (3 章)。パラメーターと正解は EM アルゴリズムを用いて推定する (4 章)。続いて、提案手法をクラウドソーシングで行った階層分類に適用し、提案手法の有効性を確認した (5 章)。

本研究の貢献は以下の点にまとめられる

- 階層構造の情報を活用した回答プロセスのモデルを構築した
- 実データを用いた実験により提案手法が精度の向上に有効なことを確認した

2 問題設定

n 個の階層分類タスクを m 人のワーカーに依頼する。本論文ではワーカーを $i \in \{1,\ldots,m\}$ とし、タスクは $j \in \{1,\ldots,n\}$ で表す。

クラス集合を $\mathbf{K}$ と表記する。クラスは s 個のカテゴリーの連なり $\boldsymbol{k} = (k_1,\ldots,k_s)$ で表される。例えば、(動物, 犬, 柴犬) のように上位のカテゴリーから下位カテゴリーに向かって分類基準が詳細化されていく。階層を表す添字は $h \in \{1,\ldots,s\}$ とする。なお最上位が $h = 1$ である。階層 h のカテゴリーの集合を $\mathbf{K}_h$ とする。それぞれの階層におけるカテゴリーは 1 個または 0 個の親に属する。つまり親カテゴリーが決まれば、その次に選ばれ得る子カテゴリーの集合が定まる。親カテゴリーを k としたときの子カテゴリーの集合を $\mathbf{K}_{kh}$ と書く。クラス $\boldsymbol{k}$ の要素は次を満たす: $k_1 \in \mathbf{K}_1, k_h \in \mathbf{K}_{k_{h-1}h}$ $(h = 2,\ldots,s)$ クラス $\boldsymbol{k}$ の階層 h のカテゴリーを含むクラスの個数を N_{kh} と定義する。

タスク j に取り組んだワーカーの集合を $\mathbf{I}_j$ で表記する。ワーカー $i \in \mathbf{I}_j$ がタスク j に与えた回答を $\boldsymbol{l}_{ij} = (l_{ij1},\ldots,l_{ijs}) \in \mathbf{K}$ とする。ただし、$\boldsymbol{l}_{ij}$ の要素は次を満たす: $l_{ij1} \in \mathbf{K}_1, l_{ijh} \in \mathbf{K}_{l_{ij(h-1)}h}$ $(h = 2,\ldots,s)$。

タスク j の真のクラスを $\boldsymbol{z}_j = (z_{j1},\ldots,z_{js})$ と表す。ワーカー i が正しい答えを与える（$\boldsymbol{l}_{ij} = \boldsymbol{z}_j$）ためには、上位から順にすべての階層で真のクラスの要素と一致する必要がある。

本研究で取り組む問題は、回収した回答の集合 $\mathbf{L} = \{\boldsymbol{l}_{ij}\}_{i=1,\ldots,m,j=1,\ldots,n}$ から、すべてのタスクの真のクラス $\mathbf{Z} = \{\boldsymbol{z}_j\}_{j=1,\ldots,n}$ を推定することである。

3 階層分類における回答プロセスのモデル

Steps モデル [Verhelst 97](3.1 節) のアイデアをクラウドソーシングに応用して、階層的分類タスクにおけるワーカーの回答プロセスをモデリングする (3.2 節)。

3.1 Steps モデル

はじめに、IRT の Steps モデルを説明する。これは複数の階層で構成されている項目に対する受験者の振る舞いのモデルである。対象とする項目の例として $\sqrt{16/4}+2=$ という計算を考えよう。正しく計算するためには、(1)16/4 を計算し、(2) その計算結果の平方根を計算し、(3)2 を足す、という 3 段の処理が必要になる。ここでの特徴は、ある階層での結果が後続の階層での選択に影響を与えるということである。この仮定は我々が対象とする階層分類と一致する。

項目 j の階層 h において受験者 i が正答する確率を q_{ijh} とおく。また、受験者 i が項目 j に正解した段数を r_{ij} とする。Steps モデルによれば、受験者が r_{ij} 回だけ正答する確率は次のように定義される:

$$\left(\prod_{h=1}^{r_{ij}} q_{ijh}\right) \cdot \left(1 - q_{ij(r_{ij}+1)}\right)^{1-\delta(r_{ij},s)} \tag{1}$$

ここで、$\delta(r_{ij}, s)$ は r_{ij} が s に等しいとき 1、それ以外のときには 0 を取る。

3.2 Steps GLAD

前節で説明した Steps モデルをクラウドソーシングでの階層分類に適用する。そのためには、項目と受験者をそれぞれタスクとワーカーに読みかえればよい。クラウドソーシングの場合はタスク j の正解クラス $\boldsymbol{z}_j$ が非観測の確率変数となる。回答の系列 $\boldsymbol{l}_{ij} = (l_{ij1},\ldots,l_{ijs})$ が生成される確率は $\boldsymbol{z}_j$ に関する条件付き確率として式 (1) から次のように定義される:

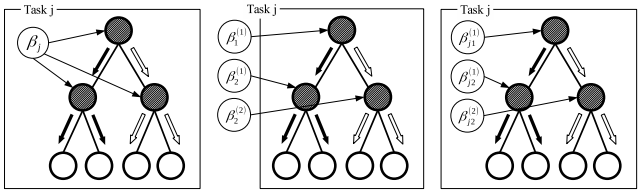
$$\mathrm{P}(\boldsymbol{l}_{ij}|\boldsymbol{z}_j) = \left(\prod_{h=1}^{r_{ij}} q_{ijh}\right) \cdot \left(\frac{1 - q_{r_{ij}+1}}{N_{z_{jr_{ij}}} - 1}\right)^{1-\delta(r_{ij},s)} \tag{2}$$

ただし、階層 $r_{ij}+1$ で判定を誤ったときには、正解 $z_{jr_{ij}}$ と同じ親を持つ不正解クラスから一様な確率で $\boldsymbol{l}_{ij}$ の値が選ばれるとする。

ワーカー i がタスク j の階層 h で正しい判定を与える確率 q_{ijh} はワーカーの能力と判定の難易度の関数で表されるとする。本研究では次に示す GLAD モデル [Whitehill 09] を用いる:

$$q_{ijh} = \frac{1}{1 + \exp(-\beta_{jh}^{(k)}\alpha_i)}$$

$\alpha_i \in (-\infty, \infty)$ はワーカーの能力を表すパラメーターであり、大きい値を取るほど正答確率が高くなる。判定の難易度を表すパラメーターは $\beta_1^{(1)} \in [0,\infty), \beta_h^{(k)} \in [0,\infty)$ $(h = 2,\ldots,s, k \in \mathbf{K}_{h-1})$ で表す。大きい値を取るほど正答確率が高くなる（判定が易しくなる）。難易度パラメーターの設定については 3 つの方法を考える。それを以下で述べる。



(a) タスクごと　(b) 判定箇所ごと　(c) 両方

図 2: タスク j に関する難易度パラメーターの設定方法

3.2.1 難易度をタスクごとに定める場合

まず、各タスクが固有の難易度を 1 つだけ持つとして正答確率を定義する (図 2(a))。このとき

$$q_{ijh} = \frac{1}{1 + \exp(-\beta_j\alpha_i)} \tag{3}$$

となる。回答の生起確率（式 (2)）は Whitehill らの GLAD を階層数分掛け合わせたものになる。

3.2.2 難易度を判定箇所ごとに定める場合

一方で、各判定箇所で難易度が異なるという考えもあろう。つまり、難易度がタスクに関わらず判定箇所ごとに 1 つ定まるとする (図 2(b))。このとき

$$q_{ijh} = \frac{1}{1 + \exp(-\beta_h^{\left(z_{j(h-1)}\right)}\alpha_i)} \tag{4}$$

となる。ただし $\beta_1^{\left(z_{j0}\right)}$ は最上位の判定箇所の難易度である。

3.2.3 難易度をタスクと判定箇所ごとに定める場合

上の 2 つの考え方を組み合わせて正答確率を定義することも考えられる。タスクそれぞれが判定箇所ごとに異なる難しさを持っているとする (図 2(c))。このとき

$$q_{ijh} = \frac{1}{1 + \exp(-\beta_{jh}^{\left(z_{j(h-1)}\right)}\alpha_i)} \tag{5}$$

ただし $\beta_{j1}^{(z_{j0})}$ は最上位の判定箇所の難易度である。これを式 (1) に代入すると

$$P(\boldsymbol{l}_{ij}|\boldsymbol{z_j};\alpha_i,\boldsymbol{\beta}_j) = \frac{\exp(\alpha_i \sum_{h=1}^{r_{ij}} \beta_{jh}^{(z_{j(h-1)})})}{\prod_{h=1}^{\min(s,r_{ij}+1)}(1+\exp(\beta_{jh}^{(z_{j(h-1)})}\alpha_i))} \cdot \left(\frac{1}{N_{z_{jr_{ij}}}-1}\right)^{1-\delta(r_{ij},s)} \quad (6)$$

となる。

4 正解とパラメーターの推定

ワーカーから集めた回答から、真のクラスと 3 章で定義したモデルのパラメーターを EM アルゴリズムによって推定する。

E ステップ それぞれのワーカーの能力は他のワーカーから独立であり、同様に判定の難易度は他の判定箇所から独立であるとする。タスク j の真のクラス $\boldsymbol{z}_j$ の事後確率は、

$$\begin{aligned}\mathrm{P}(\boldsymbol{z}_j|\mathbf{L};\boldsymbol{\alpha},\boldsymbol{\beta}) &= \prod_{i\in\mathbf{I}_j}\mathrm{P}(\boldsymbol{z}_j|\boldsymbol{l}_{ij};\alpha_i,\boldsymbol{\beta}_j)\\ &\propto \mathrm{P}(\boldsymbol{z}_j)\prod_{i\in\mathbf{I}_j}\mathrm{P}(\boldsymbol{l}_{ij}|\boldsymbol{z}_j;\alpha_i,\boldsymbol{\beta}_j)\end{aligned}$$

と計算される。また、Q 関数はこの事後確率を使って

$$\begin{aligned}Q(\boldsymbol{\alpha},\boldsymbol{\beta}) &= \mathrm{E}[\ln\mathrm{P}(\mathbf{L},\mathbf{Z};\boldsymbol{\alpha},\boldsymbol{\beta})]\\ &= \sum_j\sum_{\boldsymbol{z}_j} p_{\boldsymbol{z}_j}\ln\mathrm{P}(\boldsymbol{z}_j)\\ &\quad + \sum_{j,i\in\mathbf{I}_j}\sum_{\boldsymbol{z}_j} p_{\boldsymbol{z}_j}\ln\mathrm{P}(\boldsymbol{l}_{ij}|\boldsymbol{z}_j;\alpha_i,\boldsymbol{\beta}_j)\end{aligned}$$

となる。なお $p_{\boldsymbol{z}_j} = \mathrm{P}(\boldsymbol{z}_j|\mathbf{L};\boldsymbol{\alpha},\boldsymbol{\beta})$ である。

M ステップ Q 関数を最大化する $\boldsymbol{\alpha},\boldsymbol{\beta}$ を数値最適化で求める。5 章で示す実験では共役勾配法によって最適化した。例としてワーカーの回答の生成確率 $P(\boldsymbol{l}_{ij}|\boldsymbol{z}_j;\alpha_i,\boldsymbol{\beta}_j)$ が式 (6) に従う場合を考える。パラメーターの勾配ベクトルは以下のようになる:

$$\frac{\partial Q}{\partial \alpha_i} = \sum_{j\in\mathbf{J}_i}\sum_{\boldsymbol{z}_j} p_{\boldsymbol{z}_j}\left(\sum_{h=1}^{r_{ij}}\beta_{jh}^{(z_{j(h-1)})} - \sum_{h=1}^{\min(s,r_{ij}+1)}\beta_{jh}^{(z_{j(h-1)})}\sigma\right)$$

$$\frac{\partial Q}{\partial \beta_{jh}^{(z_{j(h-1)})}} = \sum_{i\in\mathbf{I}_j}\sum_{\boldsymbol{z}_j} p_{\boldsymbol{z}_j}\left(\delta(r_{ij}\geq h) - \sigma\delta(\min(s,r_{ij}+1)\geq h)\right)\alpha_i$$

なお、δ は与えられた不等式が成り立つとき 1 を、そうでないとき 0 を取る関数であり、σ は

$$\sigma = \frac{1}{1+\exp(-\beta_{jh}^{(z_{j(h-1)})}\alpha_i)}$$

である。

5 実験

5.1 結果

本実験の結果を表 1 に示す。どちらのデータセットにおいても提案手法が既存手法より高い精度を達成した。

本章では提案手法の有用性を評価するために、クラウドソーシングで実行した階層分類のデータセットを用いた実験結果を示す。

5.2 設定

5.2.1 提案手法

本研究の提案手法は 3 章で示した下記の 3 つである。(1) STEPS GLAD (task-dep.) はタスクごとに固有の難易度を持つモデルである (式 (3))。GLAD を階層分掛け合わせたものに等しい。(2) STEPS GLAD (step-dep.) は難易度が判定箇所ごとに定まる場合のモデルである (式 (4))。提案手法の中で最も少ないパラメーター数を持つ。(3) STEPS GLAD は難易度がタスクと判定箇所ごとに定まる場合のモデルである (式 (5))。いずれのモデルに関しても、比較手法の GLAD によって推定したワーカーの能力の推定値を初期値とした。ただし、難易度パラメーターは GLAD の初期値と同じ値に設定した。

5.2.2 比較手法

比較対象として、下記の 2 つを用意した。(1) MV はラベル集合の多数決を取る手法 (Majority Voting) である。投票数が同数のクラスが存在した場合、クラス ID が小さいものを選択する。(2) GLAD は Whitehill ら [Whitehill 09] が提案した手法である。実装は Whitehill らが公開するコードを用いた。

5.2.3 評価指標

各手法で出された推定結果を **Hierarchical Precision** [Kiritchenko 05] で評価する。正解クラスを $\boldsymbol{z} = (z_1,...,z_s)$、予測クラスを $\hat{\boldsymbol{z}} = (\hat{z}_1,...,\hat{z}_s)$ とするとき、Hierarchical Precision は $\sum_{h=1}^{s}\delta(z_h,\hat{z}_h)/s$ となる。ここで δ は 2 つの引数の値が等しいとき 1、それ以外のとき 0 を返す関数である。比較評価では、回答ワーカー数を変動させながらスコアの変化を調べる。

5.3 データセット

クラウドソーシングで収集した 2 つのデータセットについて説明する。分類作業はともにクラウドソーシングプラットフォームであるランサーズ [*1] で依頼した。ワーカーは分類対象に関する情報（識別名ほか数項目）を見て、我々が用意した分類用のインターフェースを操作し回答を行った。

5.3.1 図書分類タスク (NDLC)

このタスクの目標は、与えられた図書を特定の体系に沿って分類することである。日本で広く用いられている分類法である国立国会図書館分類表 (National Diet Library Classification; NDLC) の 2 階層目までを使用する。1 階層目のカテゴリー数が 9 個、2 階層目のカテゴリー数が 101 個である。

分類対象のデータは、国立国会図書館のデータベース [*2] より、2014 年に出版された図書の中から各クラス同数ずつ取得し、その中から 895 冊をランダムにサンプリングした。それぞれの図書に対して 10 人に分類を依頼した。

5.3.2 産業分類タスク (TDB)

このタスクの目標は、企業を業務内容に基づいて産業分類の該当カテゴリーに分類することである。帝国データバンクの分類体系（『TDB 産業分類表』）の 3 階層目までを使用する。

∗1 https://www.lancers.jp/
∗2 NDL-OPAC (http://ndlopac.ndl.go.jp)

表 1: ワーカー数別のスコア。ワーカー数 1 人と 5 人に対しては 10 回ランダムサンプリングしたときのスコアの平均と標準偏差を示す。1 人と 5 人の部分では、MV と GLAD に対して t 検定 ($p < 0.05$) で有意に精度改善が見られる部分は太字で表す。

	NDLC			TDB		
手法	1 人	5 人	10 人	1 人	5 人	10 人
MV	0.546 ± 0.010	0.620 ± 0.009	0.659	0.515 ± 0.019	0.583 ± 0.008	0.608
GLAD	0.546 ± 0.010	0.637 ± 0.011	0.669	0.515 ± 0.019	0.590 ± 0.012	0.602
STEPS GLAD (task-dep.)	0.546 ± 0.010	0.645 ± 0.007	0.674	0.477 ± 0.031	0.593 ± 0.011	0.616
STEPS GLAD (step-dep.)	0.546 ± 0.010	0.645 ± 0.007	0.674	0.515 ± 0.019	0.593 ± 0.011	0.622
STEPS GLAD	0.546 ± 0.01	**0.647** ± 0.007	**0.675**	0.515 ± 0.019	0.594 ± 0.012	**0.624**

本実験では 1 階層目のカテゴリーを製造業に固定した。2 階層目の中カテゴリー数は 21 個、3 階層目のカテゴリー数は 273 個である。

分類対象のデータは、関東甲信越地方の企業の中から 388 社をランダムにサンプリングした。それぞれ企業に対して 10 人に分類を依頼した。

5.4 結果

本実験の結果を表 1 に示す。どちらのデータセットにおいても提案手法が既存手法より高い精度を達成した。

GLAD は多数決とほとんど差がないかそれ以下の正解率を示した。1 つの原因はクラスの多さだと考えられる。GLAD はワーカー間の意見の一致に基づいて正解とパラメーターを推定する。本実験のデータのようにクラスが多く意見の一致度合いが低い場合はこのプロセスがうまく働きにくい。一方提案手法は中間層を見ることができるため利用できる情報が多く、GLAD よりも効果的に推定を行うことができる。

多数決と GLAD ではうまく推定できなかったが、提案手法では推定できた図書の例を挙げる。『地域のグローバル化にどのように向き合うか :外国人児童生徒教育問題を中心に』(田巻松雄著, 下野新聞社, 2014) という本のクラスは (教育, 教育理論) である。これに対し 10 人のワーカーは以下のように回答した:

- (教育, 教育 [一般]) : 1 名
- (教育, 教育理論) : 2 名
- (教育, 特別教育) : 1 名
- (教育, 各国の教育・教育史) : 2 名
- (社会・労働, 社会学) : 1 名
- (社会・労働, 社会問題) : 3 名

多数決と GLAD は (社会・労働, 社会問題) を答えと推定した。確かに下位を見ると 3 人が一致しており優勢である。しかし上位を見ると 6 人が教育と答えている。提案手法はこの情報を用いて正解 (教育, 教育理論) を導くことができた。

6 結論

本研究では、階層分類タスクについて、その構造情報を活かして効果的に回答を統合する手法を提案した。既存手法との比較実験によって提案手法が精度を改善することを示した。提案手法によって、少ないワーカー数で従来手法より高い品質の結果が得られることが期待でき、クラウドソーシングのコスト削減に有用である。

参考文献

[Bragg 13] Bragg, M., Jonathan and Weld, D. S.: Crowdsourcing multi-Label classification for taxonomy creation, in *Proceedings of the 1st AAAI Conference on Human Computation and Crowdsourcing* (2013)

[Callison-Burch 10] Callison-Burch, C. and Dredze, M.: Creating speech and language data with Amazon's Mechanical Turk, in *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pp. 1–12 (2010)

[Cooper 10] Cooper, S., Khatib, F., Treuille, A., Barbero, J., Lee, J., Beenen, M., Leaver-Fay, A., Baker, D., Popović, Z., et al.: Predicting protein structures with a multiplayer online game, *Nature*, Vol. 466, No. 7307, pp. 756–760 (2010)

[Dawid 79] Dawid, A. P. and Skene, A. M.: Maximum likelihood estimation of observer error-rates using the EM algorithm, *Applied Statistics*, Vol. 28, pp. 20–28 (1979)

[Kamar 15] Kamar, E. and Horvitz, E.: Planning for crowdsourcing hierarchical tasks, in *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems* (2015)

[Kiritchenko 05] Kiritchenko, S., Matwin, S., and Famili, A. F.: Functional annotation of genes using hierarchical text categorization, in *Proceedings of the BioLINK SIG: Linking Literature, Information and Knowledge for Biology* (2005)

[Verhelst 97] Verhelst, N. D., Glas, C., and De Vries, H.: A steps model to analyze partial credit, in *Handbook of Modern Item Response Theory*, pp. 123–138, Springer (1997)

[Welinder 10] Welinder, P., Branson, S., Perona, P., and Belongie, S. J.: The multidimensional wisdom of crowds, in *Advances in Neural Information Processing Systems 23*, pp. 2424–2432 (2010)

[Whitehill 09] Whitehill, J., Wu, T.-f., Bergsma, J., Movellan, J. R., and Ruvolo, P. L.: Whose vote should count more: Optimal integration of labels from labelers of unknown expertise, in *Advances in Neural Information Processing Systems 22*, pp. 2035–2043 (2009)