

AdaFlock: 予測モデリングのための クラウドソーシングによる適応的特徴生成法

AdaFlock: Adaptive Boosted Generation of Crowd Features for Predictive Modeling

高濱 隆輔 *^{1,2} 馬場 雪乃 *¹ 清水 伸幸 *³ 藤田 澄男 *³ 鹿島 久嗣 *¹
Ryusuke Takahama Yukino Baba Nobuyuki Shimizu Sumio Fujita Hisashi Kashima

*¹京都大学大学院 情報学研究科 知能情報学専攻

Department of Intelligence Science and Technology, Graduate School of Informatics, Kyoto University

*²JST ERATO 河原林巨大グラフプロジェクト *³ヤフー株式会社 Yahoo! JAPAN 研究所
JST, ERATO, Kawarabayashi Large Graph Project Yahoo! JAPAN Research, Yahoo Japan Corporation

Feature engineering is the key to successful application of machine learning algorithms to real-world data. The discovery of informative features often requires domain knowledge or human inspiration, and data scientists expend a certain amount of effort into exploring feature spaces. In this paper, we present a novel algorithm called *AdaFlock* to efficiently obtain informative features through crowdsourcing. *AdaFlock* is inspired by *AdaBoost*, which iteratively trains classifiers by increasing the weights of samples misclassified by previous classifiers. *AdaFlock* iteratively generates informative features; at each iteration, crowdsourced workers are shown samples selected according to the classification errors of the current classifiers and are asked to generate new features that are helpful for correctly classifying the given examples. The results of our experiments conducted using real datasets indicate that *AdaFlock* successfully discovers informative features with fewer iterations and achieves high classification accuracy.

1. はじめに

近年の著しい機械学習技術の発展に伴い、画像・音声・言語認識、異常検知、推薦システム、ファイナンスなどの非常に広い範囲で機械学習技術が応用されるようになってきているが、どのような機械学習の応用であっても、対象としたい実世界の情報を何らかの特徴量に落とし込んで、計算機もしくは学習アルゴリズムへ入力する必要がある。その際に「どのような特徴量を用いるか」という問題は、機械学習技術の利用結果に大きく影響する。そのため、機械学習技術者は予測に有用な特徴量の効率的な生成や選択を行いたいと考えているが、有用な特徴量の設計にはしばしば専門性の高い知識や直感、そして多大な労力が要求されるという問題がある [Domingos 12]。

しかし、近年のクラウドソーシングプラットフォームの台頭に伴い、人間と計算機の能力を組み合わせることによってこの問題を解決する新たな道筋が拓けてきている。クラウドソーシングによって安価で大量の人的リソースを機械的に獲得できるようになったため、いままでは専門家によって行われてきた専門性の高い仕事を、専門知識を持ち合わせていない多くの一般人の意見を統合することで代替したり、人間の計算能力をアルゴリズムに組み込むことで、より高度なシステムを構築したりすることができるようになった。

例えば分類問題に取り組むとき、クラウドソーシングで分類自体を行うことも考えられるが、ドメインによっては、専門知識がなければ分類は難しい [Branson 10]。一方で、非専門家にとって分類そのものが難しくても、特徴量の生成は容易なことがあり、クラウドソーシングを特徴量の生成に利用する取り組みがいくつか行われている。Cheng と Bernstein は、analogical encoding に基づき、正例と負例を区別させる一対比較タスクを用いてクラウドワーカーに Crowd-Feature を作成させ、Crowd-Feature と機械的に生成された特徴量を組み合

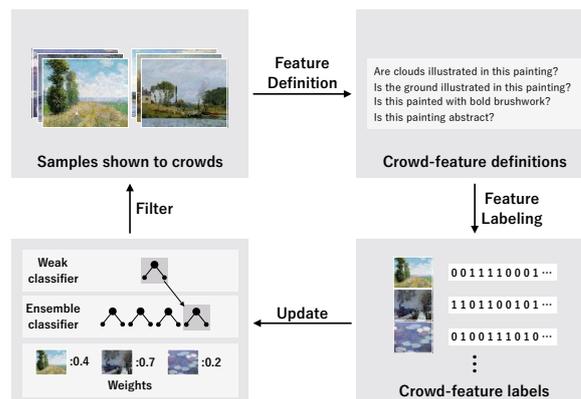


図 1: 提案手法 *AdaFlock* の概略図。クラウドソーシングを用いた特徴生成手法である *AdaFlock* は、誤分類しやすいデータを優先的にクラウドワーカーに提示することで、反復的かつ適応的に分類に有用な特徴の生成を行う。

わせることで高精度な分類器を構築するプラットフォームである *Flock* を提案しており、機械的な特徴量のみでは正常な分類を行うことが極めて困難である動画の分類タスクでも一定の精度を達成することが確認されている [Cheng 15]。Flock では、一対比較タスクにおいてクラウドワーカーに提示されるサンプルの選び方は特に工夫されていないが、効率的な特徴の生成のためには適応的な提示サンプルの選択が有効であると考えられる。Zou らは冗長な特徴が生成されるのを避けるため、過去に作られた特徴とそれに対するラベルを利用して提示サンプルを選び、特徴生成を行う手法を提案している [Zou 15]。しかし、この手法ではサンプル選択時に分類器の性能は考慮されていない。特徴生成の最終的な目的は分類器の精度の向上である場合、精度向上に貢献するという意味で有用な特徴を生成できるように適応的に提示サンプルを選びたい。

本研究では、クラウドソーシングを用いた新しい特徴生成アルゴリズムである AdaFlock を提案する。AdaFlock は二値分類問題を対象とし、ブースティングを参考に、より高精度な分類を行える特徴量を生成するような比較タスクを逐次的にクラウドワーカに提示する。ブースティングの最も有名なアルゴリズムである AdaBoost [Bradley 07] では、誤分類されやすいデータにより大きな重みを与えて弱学習器の逐次的な訓練を行う。AdaFlock では、Filter [Freund 95] を利用して誤分類されやすいデータをワーカに提示し、誤分類されやすいデータをうまく分類するような「良い」特徴を逐次的に生成する。

画像分類および動画分類の実データセットを用いて実験を行った結果、AdaFlock は Flock で生成される特徴のクラスタをカバーしながら Flock で生成されないクラスタの特徴を生成することができ、また、最終的に得られる学習器が既存手法よりも高い精度を達成することを確認した。

2. AdaFlock

2.1 問題設定

はじめに、本論文で扱う予測モデリングのための特徴生成問題の定式化を行う。本問題は、機械学習における二値分類問題を基盤とする。 X をサンプルの集合、 $Y = \{-1, +1\}$ を二値ラベルの集合とする。 N 個のラベル付きサンプルからなる訓練データセット $D = \{(x_i, y_i) \in X \times Y\}_{i=1}^N$ が与えられたとき、本問題の目的は分類器 $H: X \rightarrow Y$ を構築することである。ここで、典型的な二値分類問題と異なり、本問題では各サンプル x_i の特徴ベクトルは与えられない。したがって、学習器 H の獲得のためには、サンプルの特徴ベクトルの生成と D による分類器の訓練を同時に行う必要がある。

2.2 アルゴリズム

目的となる学習器を構築するため、AdaFlock はクラウドワーカに繰り返し特徴の生成を依頼する。より詳細には、AdaFlock は**特徴定義タスク**と**特徴ラベリングタスク**の二つのタスクをクラウドワーカに実行させる。

特徴定義タスクでは、クラウドワーカにいくつかの正例と負例を示したのち、それらの正例と負例の違いを識別する基準を記述させる。このような方法は analogical encoding と呼ばれ、単一のオブジェクトを提示した場合に比べて、オブジェクトを比較させるように提示した場合の方が、オブジェクトのより複雑な特徴に着目できることが知られている [Gentner 04]。また、クラウドソーシングによって有用な特徴を生成するために analogical encoding を応用する既存研究が存在する [Cheng 15]。AdaFlock はクラウドワーカに、「この絵画には空が描かれていますか?」「この動画に映っている人は体を動かしながら笑っていますか?」といった「はい」または「いいえ」で回答できる文を作成するよう指示する。

特徴ラベリングタスクでは、データセットに含まれる各データが、特徴定義タスクで定義された特徴の各文に合致しているか否かをクラウドワーカに回答させることで、特徴ベクトルを作成する。

クラウドソーシングを利用して有用な特徴を効率的に生成するため、AdaFlock は特徴定義タスクと特徴ラベリングタスクをブースティングの枠組みを応用して反復的に実行する。一般に、ブースティングのアルゴリズムは以下の3つのステップを反復することで学習器を構築する: (a) 新しい分布のデータを用いて弱学習器の訓練を行う、(b) 新しい分布のデータに対して分類誤差を計算する、(c) 既存の学習器では分類が困難なデータの重みが大きくなるような新しいデータ分布を作成

Algorithm 1 AdaFlock

```

1: input Crowd oracle  $\mathcal{O}$ ,  $D = \{(x_i, y_i)\}_{i=1}^N$ ,  $\delta_t$ ,  $\epsilon$ ,  $m$ ,  $k$ ,  $T$ 
2: Set weights  $w_i^1 = 1/N$ ,  $i = 1, \dots, N$ 
3:  $D^+ = \{(x, y) \mid (x, y) \in D, y = +1\}$ 
4:  $D^- = \{(x, y) \mid (x, y) \in D, y = -1\}$ 
5: for  $t = 1, \dots, T$  do
6:    $S_F^+ = \phi$ ,  $S_F^- = \phi$ 
7:   while  $|S_F^+| \leq \frac{m}{2}$  and  $|S_F^-| \leq \frac{m}{2}$  do
8:     Sample an object  $(x, y)$  using  $\text{FILTER}(w^t, \delta_t)$ 
9:     if  $y = +1$  and  $|S_F^+| \leq \frac{m}{2}$  then
10:       add  $(x, y)$  to  $S_F^+$ 
11:     else if  $y = -1$  and  $|S_F^-| \leq \frac{m}{2}$  then
12:       add  $(x, y)$  to  $S_F^-$ 
13:   Create a feature making sample set  $S_F = S_F^+ \cup S_F^-$ 
14:   Use  $\mathcal{O}$  to get crowd features  $F_t = f_t^1, \dots, f_t^k$  for  $S_F$ 
15:   Use  $\mathcal{O}$  to label crowd features  $F_t$  for  $D$ 
16:   Train a classifier  $h_t$  for  $D$  using  $F_t$ 
17:   Edge  $\gamma_t = \sum_{i \in D} \mathbf{1}[y_i = h_t(x_i)] w_i^t / \sum_{i \in D} w_i^t - 0.5$ 
18:   Set weight  $\alpha_t \leftarrow 0.5 \log(0.5 + \gamma_t) / (0.5 - \gamma_t)$ 
19:   Update  $H_t(x) \leftarrow H_{t-1}(x) + \alpha_t h_t(x)$ 
20:   Update  $w_i^{t+1} \leftarrow 1 / (1 + \exp(y_i H_t(x_i)))$ 
21: return strong learner  $H_T(x) = \text{sign}(\sum_t \alpha_t h_t(x))$ 

22: function  $\text{FILTER}(w^t, \delta_t)$ 
23:    $r = \#$  calls to  $\text{FILTER}$  so far on round  $t$ ;  $\delta'_t = \frac{\delta_t}{r(r+1)}$ 
24:    $(D', w') \leftarrow$  Permutation of  $(D, w^t)$ 
25:   for  $i = 0$ ;  $i < \frac{2}{\epsilon} \ln(1/\delta'_t)$ ;  $i = i + 1$  do
26:      $(x, y) \leftarrow$  the  $i$  th element of  $D'$ 
27:     return  $(x, y)$  with propability  $w'_i$ 
28:   exit algorithm; return  $H_t(x)$ 

```

する。AdaFlock もこの3つのステップを反復して、クラウドワーカにタスクを発行しながら学習機を構築する。

ステップ (a) では、AdaFlock は与えられたデータ分布を用いて新しい特徴を生成し、生成された特徴を用いて弱学習器を訓練する。データ分布は、直前の反復での分類誤りから計算され、各サンプルの「重み」の値として反映される。分類器の分類誤りが大きければ大きいほど各サンプルの重みが大きくなるよう重みの計算式を設計する。この重みを反映しながらクラウドワーカに特徴生成を行わせたいが、各サンプルとその重みのリストは人間にとって解釈が難しいものである。この問題を解決するための方法が Pareek と Ravikumar によって提案されている [Pareek 13]。彼らが提案したアルゴリズムである HumanBoost では、クラウドワーカを弱学習器に見立て、訓練されたクラウドワーカの意見を統合することでアンサンブル分類器を構築する。クラウドワーカを訓練する際には、データの重みに応じた確率でサンプリングされたデータのみを用いることによって、データ分布を反映した訓練を行っている。サンプリングに基づく弱学習器の訓練は Bradley と Schepire によって提案された FilterBoost [Bradley 07] における Filter 関数を応用したものである。AdaFlock でも同様の方法を用いて、Filter 関数によってサンプリングしたデータをクラウドワーカに提示し、特徴定義を依頼する。

Algorithm 1 に AdaFlock の手続きを示す。はじめに、訓練データを正例データセット D^+ および負例データセット D^-

に分割する。各ラウンド t において、ステップ (a) では、特徴定義に用いる正例集合 S_F^+ と負例集合 S_F^- を、それぞれ D^+ , D^- からサンプリングする。このとき、 S_F^+ と S_F^- の要素数がともに $\frac{m}{2}$ に達するまで、Filter 関数が呼び出される。ここで m は、アルゴリズムに事前に与えられるパラメータである。Filter 関数は、訓練データ全体である D からデータ (x, y) およびそれに対応する重み w をランダムに非復元抽出する。 (x, y) は、確率 w で採用され、その場合 Filter 関数は (x, y) を返す。そうでなければ (x, y) は破棄される。

次に、AdaFlock クラウドワーカに特徴定義タスクを依頼する。クラウドワーカは Filter で得られた特徴定義用のデータセット S_F^+ , S_F^- に含まれるサンプルを提示され、 $F_t = f_t^1, \dots, f_t^k$ の定義を行う。ここで、 t はラウンドの番号を表し、反復の回数を表す。また、 k は各ラウンドで生成する特徴数を表す。続いて AdaFlock は特徴ラベリングタスクを実行する。クラウドワーカは、 D 中の全てのサンプル $x_i \in X$ と生成された各特徴 F_t のペアに対して、2 値のラベル $\mathbf{x}_{it} = x_{it}^1, \dots, x_{it}^k$ を付けるタスクを依頼される。得られた特徴ベクトル $\{\mathbf{x}_{it}, y_i\}_{i=1}^N$ を用いて、AdaFlock は弱学習器 h_t *1 の訓練を行う。

ステップ (b) では、HumanBoost と同様に、Filter 関数が取り出した w^t を用いて h_t の D のもとの優位度である γ_t を計算する:

$$\gamma_t = \frac{\sum_{i \in D} \mathbf{1}[y_i = h_t(x_i)] w_i^t}{\sum_{i \in D} w_i^t} - 0.5.$$

ここで、 $\mathbf{1}[\cdot]$ は条件が真なら 1 を、そうでなければ 0 を返す。 h_t の D のもとの優位度 γ_t を用いて、アンサンブル分類器全体における h_t の重みである α_t を決定する: $\alpha_t = (0.5 \log(0.5 + \gamma_t)) / (0.5 - \gamma_t)$ 。

データの重みを更新するステップ (c) も HumanBoost と同様の手続きを行う。 α_t および h_t を用いてアンサンブル学習器 H_t を更新したのち、各サンプル $(x_i, y_i) \in D$ について、 H_t による学習データの分類結果に応じて重みを w_i^{t+1} に更新する: $w_i^{t+1} = (1 + \exp(y_i H_t(x_i)))^{-1}$ 。重み $w_i \in (0, 1)$ は、 (x_i, y_i) が H_t によって正しく分類されれば $w_i < 0.5$ 、誤って分類されれば $w_i > 0.5$ となる。 $y_i H_t(x_i)$ が小さくなればなるほど、つまり、誤分類の程度が酷くなればなるほど、 w_i は 1 に近づき、次のラウンドでの特徴定義タスクに利用される確率が高くなる。

以上の 3 ステップ (a), (b), (c) を T 回繰り返す、最終的な分類器 H_T を出力する。ただし、ステップ (a) において、 D から取り出されたデータがある回数以上連続で破棄された場合、AdaFlock は実行を停止し、その時点での学習器 H_t を返す。

なお、 H_T を用いて新たなサンプル x^{new} の分類を行う場合、AdaFlock によって定義された特徴 $\{F_t\}_t$ に関する x^{new} の特徴ベクトルを得るため、特徴ラベリングタスクを発行する必要がある。各弱学習器 $h_t(x^{\text{new}})$ の出力の α_t による重み付き和を計算することで、 H_T の出力が決定される。

3. 実験

3.1 実験設定

データセット 本論文では、以下に示す 2 つの実データセットを用いて実験を行った: (1) **Paintings**: 印象派の画家 Claude Monet と Alfred Sisley による絵画のデータセットである。Monet の絵画は “Claude Oscar Monet -

The complete works” *2 より、Sisley の絵画は “Alfred Sisley - The complete works” *3 より、それぞれ取得した。(2) **Smiles**: 人が笑っている動画のデータセットである。UvA-NEMO Smile Database *4 [Dibeklioğlu 12] より、自然に発生した笑いの動画と指示された作り笑いの動画をそれぞれ取得した。Paintings, Smiles はともに訓練データセット 200 件およびテストデータセット 200 件からなる。また、200 件の内訳は正例と負例 100 件ずつである。

実験の手続き 本論文の実験では、AdaFlock の特徴生成を合計 10 ラウンド行う。各ラウンドでは、10 個の特徴の生成とそのラベリング、そして学習器のアップデートを行う。結果として、実験全体では各データセットにつき 100 個の特徴が生成されることとなる。

ベースライン 本論文では、以下に示す 3 つのベースライン手法と提案手法である AdaFlock との比較を行った。(1) **Crowd prediction**: クラウドワーカにサンプルの正解ラベルを直接聞くことによってラベルの推定を行う手法である。(2) **AdaBoost with Off-the-shelf-features**: 簡単に抽出できる機械的な特徴を用いて、AdaBoost によってラベルの推定を行う手法である。Paintings データセットに対しては、各絵画データを横 300 ピクセル、縦 167 ピクセルにリサイズし、RGB の値に randomized PCA を適用したものを特徴ベクトルとする。Smiles データセットに対しては、動画データから抽出できる単純な特徴量がない [Cheng 15] ことを鑑みて、本手法による精度は考えないこととする。(3) **AdaBoost with StandardFlock-crowd-features**: クラウドソーシングによって一括で特徴生成を行い、AdaBoost によってラベルの推定を行う手法である。AdaFlock における逐次的な特徴の生成を行わず、一括で 100 個の特徴を生成する。

3.2 実験結果

図 2(a), 2(b) に AdaFlock の反復における訓練精度とテスト精度の変化を示す。これらの図より、反復に伴って訓練精度が漸増し、それに伴ってテスト精度も向上していることを読み取れる。Painting データセットについては、5 回目の反復で「この絵画には地面が描かれていますか?」というシスレーの絵画の特定に有用な特徴を発見しており、この発見が 10% のテスト精度の向上に寄与している。また、同様の現象が Smiles データセットについても確認されており、「この動画に映っている人は笑っている時間が短いですか?」という作り笑いの特定に有用な特徴が発見された 7 回目の反復では、テスト精度が 8% の性能向上を達成している。これらの結果より、AdaFlock が特徴生成を反復的に繰り返すなかで、分類の難しいサンプルをクラウドワーカに提示することではじめて有用な特徴を発見できていることが示唆されている。

続いて、表 1 に AdaFlock の反復終了時のテスト精度とベースライン手法による予測精度の比較結果を示す。表 1 より、AdaFlock がベースライン手法をテスト精度の観点で性能的に上回ることを確認できる。Paintings データセットでは、Crowd prediction, Off-the-shelf-features による予測精度に対して StandardFlock および AdaFlock による予測精度が高い

*1 この弱学習器は Decision Tree などの機械学習モデルを利用することを想定しており、HumanBoost で行われているように弱学習器を人間で代替することはしない。

*2 <http://www.claudemonetgallery.org/>

*3 <http://www.alfredsisley.org/>

*4 <http://www.uva-nemo.org>

表 1: テスト精度の比較表。提案手法である AdaFlock による精度がベースライン手法の精度を上回っていることを確認できる。

Dataset	Crowd prediction	AdaBoost with Off-the-shelf-features	AdaBoost with StandardFlock-crowd-features	AdaFlock
Paintings	0.560	0.650	0.755	0.820
Smiles	0.720	-	0.715	0.780

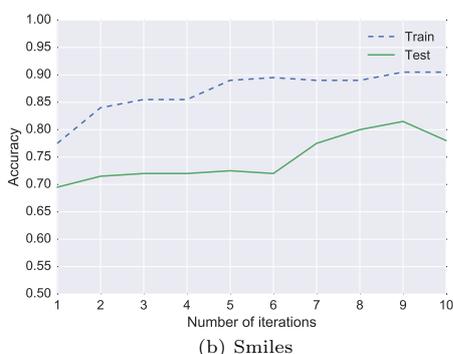
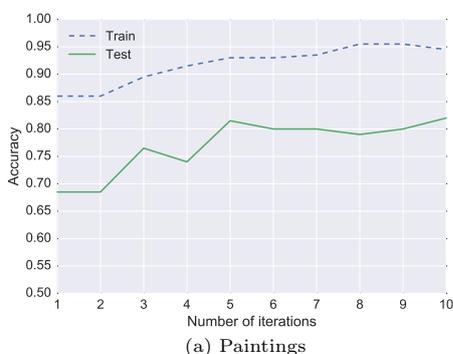


図 2: 各データセットの AdaFlock における訓練精度およびテスト精度を示した図。いずれのデータセットについても、反復に伴って両精度が漸増していることを読み取れる。

ことから、クラウドワーカによる直接的な予測や単純な機械学習手法では分類が難しいタスクでも、特徴生成を経由することで高精度な分類ができることが示唆され、クラウドソーシングによる特徴生成の有用性を確認することができる。一方、Smiles データセットに関しては、Crowd prediction による予測が一定の精度を達成するため、StandardFlock による特徴生成を介した手法はクラウドワーカによる予測に対して精度的な有用性を示さないが、AdaFlock による予測はさらに高いテスト精度を達成することを確認した。

加えて、AdaFlock は、2 回目ないし 3 回目の反復で StandardFlock と同等の推定精度を達成している。これは、適切な反復回数で AdaFlock の実行を停止することによって、クラウドソーシングによる特徴生成に伴うコストを削減できる可能性を示唆している。

4. 結論

本論文では、クラウドソーシングによって分類に有用な特徴を生成する新しいアルゴリズムである AdaFlock を提案した。AdaFlock は分類精度を改善するという観点で有用な特徴を反復的に生成する。AdaFlock の反復の過程では、AdaBoost を参考に、最新のアンサンブル学習器で誤分類されやすいサ

ンプルを優先的にクラウドワーカに提示して特徴生成を依頼する。サンプルをクラウドワーカに提示する際の選択方法は、FilterBoost で用いられている Filter 関数を応用した。実データを用いて行った実験の結果、AdaFlock が分類に有用な特徴を効率的に発見できることを確認した。また、反復を繰り返すことによって、テストデータに関する分類精度の観点で、AdaFlock による分類精度がベースライン手法による分類精度を上回ることも確認した。

参考文献

- [Bradley 07] Bradley, J. K. and Schapire, R. E.: Filter-Boost: Regression and Classification on Large Datasets, in *Advances in Neural Information Processing Systems 20*, pp. 185–192 (2007)
- [Branson 10] Branson, S., Wah, C., Schroff, F., Babenko, B., Welinder, P., Perona, P., and Belongie, S.: Visual Recognition with Humans in the Loop, in *Proceedings of the 11th European Conference on Computer Vision*, pp. 438–451 (2010)
- [Cheng 15] Cheng, J. and Bernstein, M. S.: Flock: Hybrid Crowd-Machine Learning Classifiers, in *Proceedings of the 18th ACM Conference on Computer-Supported Cooperative Work and Social Computing*, pp. 600–611 (2015)
- [Dibeklioglu 12] Dibeklioglu, H., Salah, A. A., and Geyer, T.: Are You Really Smiling at Me? Spontaneous Versus Posed Enjoyment Smiles, in *Proceedings of the 12th European Conference on Computer Vision*, pp. 525–538 (2012)
- [Domingos 12] Domingos, P.: A Few Useful Things to Know About Machine Learning, *Communications of the ACM*, Vol. 55, No. 10, pp. 78–87 (2012)
- [Freund 95] Freund, Y. and Schapire, R. E.: A Decision-Theoretic Generalization of On-line Learning and an Application to Boosting, in *Proceedings of the 2nd European Conference on Computational Learning Theory*, pp. 23–37 (1995)
- [Gentner 04] Gentner, D., Loewenstein, J., and Thompson, L.: Analogical Encoding: Facilitating Knowledge Transfer and Integration, in *Proceedings of the 26th Annual Meeting of the Cognitive Science Society*, Vol. 26 (2004)
- [Pareek 13] Pareek, H. and Ravikumar, P.: Human Boosting, in *Proceedings of the 30th International Conference on Machine Learning*, pp. 338–346 (2013)
- [Zou 15] Zou, J. Y., Chaudhuri, K., and Kalai, A.: Crowdsourcing Feature Discovery via Adaptively Chosen Comparisons, in *Proceedings of the 3rd AAAI Conference on Human Computation and Crowdsourcing*, pp. 198–205 (2015)