マッチング数制約下でのアノテーション検証割り当ての自動化

守山慧 [1]　中山功太 [2]　馬場雪乃 [1]
[1] 東京大学大学院総合文化研究科　[2] 理研 AIP
{kei-moriyama,yukino-baba}@g.ecc.u-tokyo.ac.jp　kouta.nakayama@riken.jp

概要

自然言語処理のためのデータセット構築において，アノテーションの質と量が重要である．アノテーションの労力を軽減するために，LLM 等の機械学習モデルにアノテーションを任せる方法が考えられる．機械学習モデルによるアノテーションは正しいとは限らないため，アノテーション結果を検証する必要がある．アノテーション検証を人間のアノテータに依頼すると検証の正確性が高い代わりにコストも高い．コストを下げるために，クラウドワーカーや LLM の活用が考えられるが，検証の正確性は下がる．データに応じて異なるエージェントに検証タスクを割り当てることで，予算を有効活用したい．本研究では，エージェントごとに割り当て可能な検証タスク数に制約がある下で，自動的に検証タスクを割り当てる手法を提案する．エージェントとタスクの相性を捉えた重みを正解既知のデータから推定し，タスクの割り当てを重み付き 2 部グラフの最適化問題として定式化して解く．特に，そのエージェントのみが正解している時に重みが大きくなるようにする．実データを用いた実験で，ランダム割り当てと比較して，提案手法により正解率が改善することを確認した．

1　はじめに

自然言語処理のためのデータセット構築において，アノテーションの質と量が重要である．人手によるアノテーションの労力を軽減するため，大規模言語モデル (LLM) などの機械学習モデルにアノテーションを任せるという方法が考えられる．機械学習モデルは，大量のデータに対して低コストでアノテーションを付与できるが，アノテーションの結果が正しいとは限らないため，その正しさを検証する必要がある．

アノテーション検証を専門のアノテータに依頼すると，検証の正確性が高い代わりに，コストも高い．コストを下げるために，クラウドワーカーや LLM の活用も考えられるが，アノテータと比較すると検証の正確性は下がる．検証の正確性を維持しつつコストを下げるために，データに応じて異なるエージェント（例：アノテータ，クラウドワーカー，LLM）に検証を割り当てることが考えられる．コストの低いエージェントでも正確に検証できると期待できるデータはそのエージェントに任せ，検証が難しいデータだけを正確性の高いエージェントに任せるという割り当てを実現したい．

本研究では，検証タスクの割り当てを自動化する手法を提案する．エージェントへの検証タスクの割り当てを，重み付き 2 部グラフのマッチング問題として扱う．予算に応じて，各エージェントに割り当て可能なタスク数（マッチング数）の上限が与えられているとする．この 2 部グラフを用いて，マッチング数の制約下での線形計画問題としてマッチング問題を解くことで，タスクの割り当てを行う．提案手法の全体図を図 1 に示す．提案手法では，正解既知のデータを利用して，エージェント毎の専門性に合わせた重みを推定する重みモデルを学習し，2 部グラフに対して重みを付与する．エージェント毎の重みを推定するために，エージェント毎の正解数に応じた報酬を返す報酬関数を設計した．特に，他のエージェントが誤答するデータに正解できるエージェントは専門性が高いとみなし，類似データも割り当てるように重みを決めたい．そのため，正答エージェント数が少ない場合報酬の値は大きく，多い場合は報酬が少なくなるようにした．

実データを用いた実験では，ランダムな割当よりも，提案手法による割当の方が正解率が良いことが確認できた．一方で，正答エージェント数に基づく報酬導入の効果は僅かであることが確認された．

2　関連研究

エージェントに対するタスク割り当てでは様々な観点からの研究が行われている．クラウドソーシン

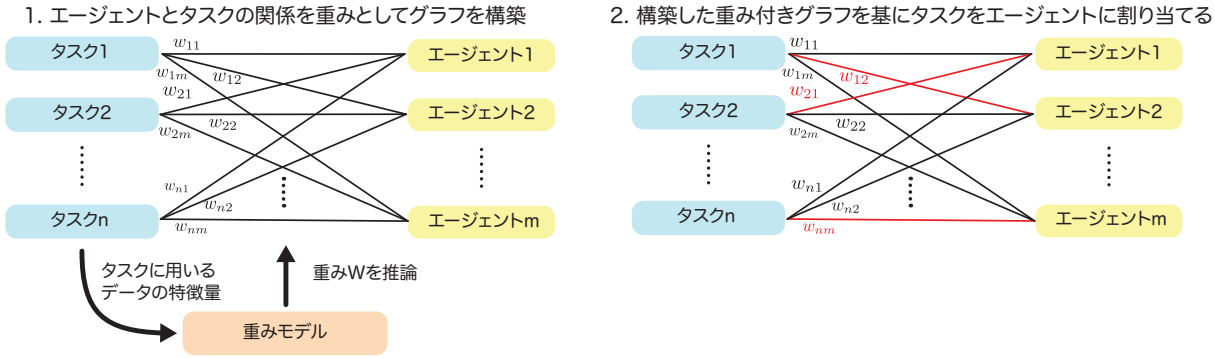

図 1　タスク割り当てで用いる手法の全体図

グにおけるタスク割当では，能動学習においてエージェントに割り当てるタスクを決める手法 [1, 2] や，線形計画法を基に最適化を行う手法 [3, 4, 5] がある．線形計画法を用いた手法では制約や目的関数の中で様々な拡張がされており，公平性に関する制約を組み込んだ手法 [4] などがあり，様々な形に拡張されている．

人間と分類モデルで協調してタスクを解く手法 [6, 7, 8, 9] では，分類モデルの学習と，人間がタスクを割り当てられた時の正解率を推定する関数の学習を同時に行う．タスクの割当は，分類モデルの予測の最大確率と人間の正解率の推定関数の値を比較し，依頼先を決定する．Mozannar らの手法 [7] では，タスクを割り当てることができる 人間が 1 人である問題設定であったが，verma らの手法 [9] において人間が複数の場合においてそれぞれのエージェントの信頼度を推定する関数を学習させることで十分であることを示している．

3　準備

3.1　問題設定

本研究では，あるアノテーションデータの検証タスクをどのエージェントに割り当てるかを決める問題を扱う．この時，各エージェントが担当できるタスクの数（マッチング数）には上限があり，これを超えないように割り当てる．

この問題を図 1 にあるように，重み付き 2 部グラフ $G = (X, A, E)$ のマッチング問題として扱う．ここで X は検証タスクの対象となるデータ集合，A はエージェント集合，E はデータとエージェントの頂点間の辺集合を表す．全てのデータ $x_i \in X$ とエージェント $a_j \in a$ の辺 $e_{ij} \in E$ において，割り当てた際の効用として重み $w_{ij} \in W$ がグラフに存在する．データのエージェントに対する割り当てを，データとエージェントの頂点間で辺を張ることとみなし，重みが最大になるように割り当てを最適化することを目指す．加えて，エージェントの過去の検証結果から，適切なエージェントに対して割り当てが行われるような重みをデータの特徴量から推論する，重みモデルの学習を行う．

4　提案手法

本手法では，Dickerson らの手法 [5] で提案されたエージェントへのタスク割り当ての手法を基にする．加えて，グラフの頂点間に付与するための重み割り当て関数を，正解が既知の検証データにおける正答エージェント数を基に学習する．

4.1　エージェントに対するタスク割り当て

検証タスクに用いるデータ $x_i \in X$ に対して重みモデル $f(\cdot)$ が j 番目のエージェント $a_j \in A$ に割り当てた重みを $w_{ij} = f_j(x_i)$ とする．$f_j(x_i)$ はデータ x_i における重みモデル $f(x_i)$ の j 番目の値，N_j はエージェント a_j に対するマッチング数の上限を表す．エージェントに対するタスク割り当ての目的関数を式 (1)，制約式を式 (2) から式 (4) に示す．

$$\text{maximize} \quad \sum_{i=0}^{n}\sum_{j=0}^{m} w_{ij}e_{ij} \tag{1}$$

$$\text{subject to} \quad \sum_{j=0}^{m} e_{ij} = 1 \qquad \forall i \in \{0,\dots,n\} \tag{2}$$

$$\sum_{i=0}^{n} e_{ij} \le N_j \qquad \forall j \in \{0,\dots,m\} \tag{3}$$

$$e_{ij} \in \{0,1\} \tag{4}$$

式 (2) は，各タスクに割り当てるエージェントの数は 1 つという制約を示している．式 (3) は，エージェント i のマッチング数は N_i 以下であるという制約を表している．

このような制約を導入することで，予算に応じて事前に設定された，各エージェントのマッチング数の上限 N_i を超えない割当を実現する．

4.2 エージェントとデータ間の重みの推定

重みモデル $f(\cdot)$ の学習は式 (5) を用いて行う．この損失関数で学習した重みモデル $f(\cdot)$ を用いて，エージェントとデータ間の重みを推定する．データ x_i において，検証の正解を $y_i \in \{0,1\}$，j 番目のエージェントの回答を $a_{ij} \in \{0,1\}$，正答エージェント数を n_i とする．この損失関数は，式 (6) で定義した報酬関数の値を活用して，正答エージェント数が少ないデータに正解しているエージェントのペアに，大きな重みを割り当てるように学習を行う．

$$\mathcal{L}_{weight} = -\sum_{i=0}^{n}\sum_{j=0}^{m} r(y_i, a_{ij}, n_i) \cdot \log f_j(x_i) \qquad (5)$$

重みの学習における報酬関数の定義を式 (6) に示す．$\alpha (0 \le \alpha \le 1)$ は，正答エージェント数による報酬の割引をどの程度反映させるかをコントロールするためのハイパーパラメータである．

$$r(y_i a_i, n_i) = \begin{cases} (1-\alpha) + \alpha \cdot \frac{1}{n_i} & y_i = a_i \\ 0 & \text{otherwise} \end{cases} \qquad (6)$$

報酬関数は，そのデータにおいてエージェントが正解している時に報酬を与える．各エージェントの専門性を考慮した重みの学習において報酬を決めるために，データ x_i における正答エージェント数 n_i を用いた．正答エージェント数が少ない場合，報酬の割り引きを小さくし，多い場合，報酬の割り引きが大きくするように設計した．そのため，重みモデルは正答エージェント数が少ないデータに正解しているエージェントについては，当該データに対して大きな重みを割り当てる．

5 実験

5.1 実験設定

本手法の有効性を確かめるために，実データを用いた実験を行った．対抗手法として，各エージェントのマッチング数の上限に応じてランダムに割り当てを行う手法と比較した．この手法をシード値を替えながら 100 回割当先を決定し，正解率の平均値をスコアとして用いた．

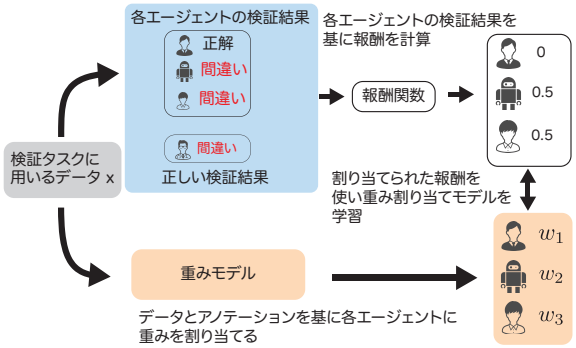

図 2 重みモデルの学習の概要図．3 つのエージェントのうち，2 つのエージェントが正答している場合を表している．

正答エージェント数を用いた報酬の割引の効果を検証するために，式 (6) の α を 0 から 1 まで 0.1 刻みで変化させ，正解率の推移を比較した．全ての実験において，式 (3) における，各エージェントのマッチング数の上限 N_i には，エージェント全体で同数にし，N_i の総和はデータ数と同じになるように設定した．

5.1.1 データセット

実験では，森羅プロジェクト [10] において開催された共有タスクの 1 つである属性値抽出タスクにおいて提出された 6 種類の機械学習モデルの予測を用いた．属性値抽出タスクとは，関根らが定義した拡張固有表現 [11] を文章中から該当する文章とカテゴリを属性値として抽出するタスクである．共有タスクで提出された属性値を，専門家に検証してもらい正解となる検証結果を作成した．エージェントの検証結果としてクラウドソーシングを通して 1 つのデータ当たり 10 人の非専門家に検証してもらい，その結果と 6 種類のシステムの抽出結果を基に作成した．特徴抽出に使用するためのテキストには，抽出された属性値とそのカテゴリ，加えて属性値の周囲の文章を用いた．

5.1.2 使用するモデル

重みモデルには多層パーセプトロン (以下 MLP) を用いて学習を行った．テキストの特徴量抽出

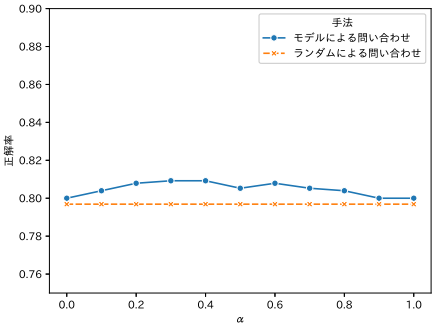

図 3　システムとクラウドワーカーの判定結果を用いた場合の実験結果

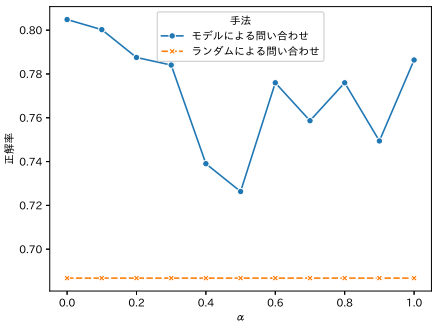

図 4　6 種類のモデルの抽出結果を用いた場合の実験結果

には事前学習済みの RoBERTa [12] を使用した．RoBERTa の事前学習タスクには森羅 2022 の共有タスクの 1 つである属性値抽出タスクを用いた．抽出された特徴量と式 (5) を用いて重みモデルを学習させた．モデルの学習時には，RoBERTa のパラメータは固定し，MLP のみの学習を行った．

推定された重みを用いて，最適化問題を解くための線形計画法のソルバには CBC[1] を用いた．

5.2　実験結果

5.2.1　システムとクラウドワーカーの判定結果を用いた実験

この実験では，エージェントとして，モデルの多数決 (以下システム) とクラウドワーカーの多数決を使用した．システムの判定結果にはその属性値を抽出した機械学習モデルの数が過半数 (3 種類) 以上のとき正しいと判定したとして作成し，クラウドワーカーの判定結果は「正しい」と回答したワーカーの数が 7 個以上の時正しいと判定したとして検証結果を作成した．

正解率の推移を図 3 に示す．図 3 より，α に関わらず提案手法における割当の正解率は，ランダムに割当先を決めるよりも正解率が良いことが分かる．また，人数による優先度の割引が無い $\alpha = 0$ の場合の正解率は 0.8 であったのに対して，割引がある $\alpha = 0.3, 0.4$ の場合の正解率は 0.809 であった．そのため，僅かではあるが正答エージェント数における報酬の割引に効果があると言える．

5.2.2　6 種類のモデルの抽出結果を用いた実験

この実験では，エージェントとして 6 種類のモデルの抽出結果を用いた．検証結果は，それぞれのモデルがその固有表現を抽出した場合正しい，抽出していない場合は間違いとして作成した．

実験結果を図 4 に示す．5.2.1 項の実験結果と同様に，α の値に関わらず提案手法により割当先を決める方が正解率が良くなることが確認できた．一方で，$\alpha = 0$ の時が正解率が最も良いことから，正答エージェント数による報酬の割引を行うことで正解率が低下していることがわかる．このことから，この実験設定における正答エージェント数による報酬の割引の効果が無いことがわかる．

6　結論

本研究では，エージェントに対する検証データの割り当てを重み付き 2 部グラフで扱う問題を扱い，2 部グラフの構築時に付与する重みを正答エージェント数を基に学習する手法を提案した．実験結果より，ランダムに問い合わせ先を決定する手法に比べ精度が良いことや，正答エージェント数による報酬の割り引きが有効になるパラメータ設定があるが，正解率の改善は僅かであることが確認できた．

今後の展望として，より適切な問い合わせが可能な重みを学習するための損失関数を設計することや，正答エージェント数における報酬の割引が有効になる場合の条件の特定が挙げられる．

1)　https://www.coin-or.org/Cbc/

謝辞

本研究は、JST ムーンショット型研究開発事業 (JPMJMS2236-8) の支援を受けたものである.

参考文献

[1] Yan Yan, Romer Rosales, Glenn Fung, and Jennifer G. Dy. Active learning from crowds. In **Proceedings of the 28th International Conference on International Conference on Machine Learning, ICML 2011**, pp. 1161–1168, 2011.

[2] Yan Yan, Romer Rosales, Glenn Fung, Faisal Farooq, Bharat Rao, and Jennifer Dy. Active learning from multiple knowledge sources. In **Proceedings of the 15th International Conference on Artificial Intelligence and Statistics, AISTATS 2012**, pp. 1350–1357, 2012.

[3] Chien-Ju Ho, Shahin Jabbari, and Jennifer Wortman Vaughan. Adaptive task assignment for crowdsourced classification. In Sanjoy Dasgupta and David McAllester, editors, **Proceedings of the 30th International Conference on Machine Learning, ICML 2013**, pp. 534–542, 2013.

[4] Naman Goel and Boi Faltings. Crowdsourcing with fairness, diversity and budget constraints. In **Proceedings of the 2nd AAAI/ACM Conference on AI, Ethics, and Society, AIES 2019**, pp. 297—304, 2019.

[5] John P. Dickerson, Karthik Abinav Sankararaman, Aravind Srinivasan, and Pan Xu. Assigning tasks to workers based on historical data: Online task assignment with two-sided arrivals. In **Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems, AAMAS 2018**, p. 318–326, 2018.

[6] David Madras, Toniann Pitassi, and Richard Zemel. Predict responsibly: Improving fairness and accuracy by learning to defer. In **Proceedings of the 32nd International Conference on Neural Information Processing Systems, NIPS 2018**, pp. 6150–6160, 2018.

[7] Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In Hal Daumé III and Aarti Singh, editors, **Proceedings of the 37th International Conference on Machine Learning, ICML 2020**, pp. 7076–7087, 2020.

[8] Rajeev Verma and Eric Nalisnick. Calibrated learning to defer with one-vs-all classifiers. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, **Proceedings of the 39th International Conference on Machine Learning, ICML 2022**, pp. 22184–22202, 2022.

[9] Rajeev Verma, Daniel Barrejon, and Eric Nalisnick. Learning to defer to multiple experts: Consistent surrogate losses, confidence calibration, and conformal ensembles. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent, editors, **Proceedings of The 26th International Conference on Artificial Intelligence and Statistics, AISTATS 2023**, pp. 11415–11434, 2023.

[10] Satoshi Sekine, Kouta Nakayama, Maya Ando, Yu Usami, Masako Nomoto, and Koji Matsuda. SHINRA2020-ML: Categorizing 30-language wikipedia into fine-grained NE based on ""resource by collaborative contribution" scheme. In **Proceedings of The 3rd Conference on Automated Knowledge Base Construction, AKBC 2021**, 2021.

[11] Satoshi Sekine, Kiyoshi Sudo, and Chikashi Nobata. Extended Named Entity Hierarchy. In **Proceedings of The 3rd International Conference on Language Resources and Evaluation, LREC 2002**, 2002.

[12] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. **arXiv preprint arXiv:1907.11692**, 2019.