2025 年度

# 修　士　論　文

# 単語種別依存の作業者能力逐次推定を用いた対訳評価タスクのオンライン割当て


指導教員: 村上 陽平

立命館大学大学院 情報理工学研究科
情報理工学専攻 博士課程前期課程
計算機科学コース

学生証番号: 6611240069-0

氏名: 山本 涼太郎

# 単語種別依存の作業者能力逐次推定を用いた対訳評価タスクのオンライン割当て

山本　涼太郎

**内容梗概**

低資源言語の言語資源を効率的に構築する手段としてクラウドソーシングが利用されている．このようなクラウドソーシングでは，不特定多数の作業者の能力にばらつきがあるため，作業者の回答をそのまま採用すると誤った言語資源が混入する可能性があり，品質管理が重要な課題となる．そこで，複数の作業者に同一のタスクセットを割当て，多数決により最終ラベルを決定する手法が用いられている．この手法は，正答を知っている作業者間ではラベルが一致する傾向を用いて，作業者能力を事前に推定することなく正答か誤答かを分類できる点に強みがある．しかしながら，作業者とタスクのミスマッチにより誤答が集中した場合，誤答を分類できるものの，その結果作業のやり直しが必要となり，作業効率の低下を招く．

この課題を解決するために，本研究では作業者ごとのタスク難易度を考慮し，成否予測に基づくタスク割当てを行う手法を提案する．作業者が新たなタスクに取り組むたびに作業履歴から成否確率を逐次更新し，次タスクの割当てに反映させることで効率的な作業進行を実現することを目的とする．本研究の課題は以下の 2 点である．

**タスクの分類**

対訳評価タスクでは，評価対象となる原語の単語の種類によってタスクの難易度や正答率が異なるという単語種別依存性が存在する．この依存性を考慮せずに各タスクを独立に扱うと，作業者の履歴のみでは未経験タスクに対する成否予測が困難となるため，類似タスクをクラスタリングする必要がある．

**オンラインタスク割当て**

クラウドソーシングにおいて作業者の能力は作業途中で変動するものではないが，作業開始時点では履歴が十分でないため，作業者能力の推定精度が低いというコールドスタート課題がある．静的割当てでは，初期の推定結果のみに基づいて割当てるため，作業履歴の蓄積による能力推定の改善をタスク割当てに反映できない．そのため，作業者能力を逐次的に推定し，その結果を即座に割当てに反映するオンラインタスク割当てが必要である．

これらの課題に対して，原語に対し意味に基づく分散表現ベース分類と，レーベンシュタイン距離に基づく形態類似度ベース分類の 2 種類を用いた．分散表現ベース分類では，Word2Vec により単語の分散表現を取得し，単語の意味的類似性に基づいてクラスタリングを実施した.分散表現は k-means 法によりクラスタを形成した．この際，クラスタリング結果の言語的妥当性をインドネシア語母語話者により確認した．一方，形態類似度ベース分類では，インドネシア語の語根と接辞の性質を活かしたレーベンシュタイン距離を単語間距離として計算し，文字列の形態的類似性に基づいて同様に k-means 法によるクラスタリングを行った．これら 2 種類のクラスタ分類の結果を問題種別として付与し，学習履歴から将来の成否を推定する深層学習モデルである Deep Knowledge Tracing(DKT)の入力として利用した．

二つ目の課題に対しては，タスクの作業結果を逐次 DKT に入力して次タスクの成功確率に基づいて割当てを行う．タスク割当てには，1 人の作業者に複数のタスクを割当てるハンガリアン法を用いる．

提案手法であるオンラインタスク割当て手法では，作業結果を逐次 DKT に入力して成功確率を更新し，更新された確率に基づいてタスクを動的に割当てる．比較対象として，タスク開始時点に全タスクの成否確率を DKT で予測し，一括で割当てを決定する静的 DKT 割当て手法とランダムな割当てを用いた．提案手法の有効性を検証した．本研究の貢献は以下のとおりである．

**タスクの分類**

分散表現ベース分類と形態類似度ベース分類の 2 種類を用いて DKT による成否予測を行った結果，両手法とも作業者の過去履歴に基づき将来のタスク成否を推定可能であることを確認した．さらに，分散表現ベース分類は，形態類似度ベース分類より約 3.4%に高い予測精度を示し，意味的特徴を用いた分類がより有効である傾向が見られた．

**オンラインタスク割当て**

提案手法であるオンラインタスク割当て手法と比較手法である静的 DKT 割当て，履歴正答率ベース割当て，ランダム割当てを 5-fold Cross Validation により比較した結果，オンラインタスク割当てが 4 条件中 3 条件で約 2~3%高い平均正答率を示した．

Online Task Assignment for Translation Evaluation

# Using Word-Type Dependent Sequential Worker Ability Estimation

Ryotaro Yamamoto

**Abstract**

Crowdsourcing has been widely used as an efficient approach for constructing linguistic resources for low-resource languages, where translation evaluation tasks are assigned to a large number of anonymous workers. However, variations in worker ability make quality control a critical challenge. When worker ability is unknown in advance, directly accepting worker responses may introduce incorrect entries into linguistic resources. To address this issue, prior studies often assign the same task set to multiple workers and determine the final label by majority voting. Although this approach does not require prior estimation of worker ability, mismatches between workers and task difficulty can lead to concentrated errors, resulting in task rework and reduced efficiency.

To overcome this limitation, this study proposes a task assignment framework based on success probability prediction that explicitly considers task difficulty for each worker. The proposed framework sequentially updates success probabilities from worker task histories whenever a worker completes a new task and immediately reflects the updated predictions in subsequent task assignments, thereby enabling efficient online task execution.

This study addresses the following two challenges.

**Task Classification.**

In translation evaluation tasks, task difficulty and accuracy depend on the word type of the source word, indicating a word-type dependency. Treating tasks independently without considering this dependency makes it difficult to predict worker performance on unseen tasks solely from worker histories. Therefore, grouping similar tasks is necessary to enable reliable generalization.

**Online Task Assignment.**

Although worker ability remains relatively stable during task execution, limited initial histories reduce estimation accuracy at early stages. Static task assignment relies only on initial predictions and cannot incorporate improvements in worker ability estimation obtained from accumulated histories. Consequently, online task assignment that sequentially updates worker ability and immediately reflects

the updates in task allocation becomes essential.

To address these challenges, this study introduces two task classification methods: semantic embedding–based classification and morphological similarity–based classification. In the semantic embedding–based approach, distributed word representations are generated using Word2Vec, and k-means clustering is applied to groups words based on semantic similarity. The number of clusters is determined to ensure that each cluster contains a sufficient number of samples for learning, and the linguistic validity of the clustering results is verified by native Indonesian speakers. In contrast, the morphological similarity–based approach computes Levenshtein distances between words by exploiting the characteristics of Indonesian roots and affixes and applies k-means clustering based on morphological similarity. The resulting clusters are treated as problem types and are used as inputs to Deep Knowledge Tracing (DKT), a deep learning model that predicts future task success from historical performance.

For online task assignment, the framework sequentially inputs task outcomes into DKT and assigns subsequent tasks based on predicted success probabilities using the Hungarian algorithm to allocate multiple tasks to each worker. The proposed method dynamically updates success probabilities, while comparative evaluations include static DKT-based assignment, which predicts all probabilities only at the initial stage, and random assignment.

Experimental results demonstrate the effectiveness of the proposed approach. Both semantic embedding–based and morphological similarity–based task classifications enable DKT to predict future task success from worker histories, with semantic embedding–based classification achieving slightly higher accuracy. A 5-fold cross-validation comparison among online task assignment, static DKT-based assignment, history-based accuracy assignment, and random assignment shows that the proposed online task assignment achieves the highest average accuracy in three out of four experimental conditions.

# 単語種別依存の作業者能力逐次推定を用いた
# 対訳評価タスクのオンライン割当て

# 目次

# 第1章 はじめに

インドネシアとその周辺地域では 700 以上の言語が話されているが，その多くは話者減少により消滅の危機に瀕している．実際に，今後 50 年から 100 年の間に話者数の減少によって使用言語数は約 50 にまで減少すると予測されており，147 言語がすでに深刻な危機状態にある[1]．これらの言語資源を保護し，地方言語間におけるコミュニケーション支援を行うためには，対訳辞書やコーパスなどの言語資源の整備が不可欠である．

このような言語資源の構築手段として，近年ではクラウドソーシングが広く活用されている．クラウドソーシングは，不特定多数の作業者にインターネットを通じてタスクを依頼することで，情報や労力を集める手法である．特に，コンピュータでは困難だが，人間にとって比較的簡単な翻訳，画像アノテーション，感情評価タスクなどに用いられることが多い．このようなクラウドソーシングでは，不特定多数の作業者が参加するため，作業者の能力にばらつきがある．特に，言語資源作成タスクは，専門性を必要とし，作業者全員が同じレベルでタスクを処理できるとは限らないため，成果物の品質を保証することは困難であり，クラウドソーシングにおける品質管理は重要な課題となる．作成タスクだけでなく，作成した言語資源の評価タスクも組み合わせる事で品質の保証を目指している．しかしながら，タスクを実行するまで作業者の能力は不明であるため，能力の高く，信頼できる作業者にのみタスクを割当てることはできない．そのため，作業者の作成した言語資源をそのまま利用するのではなく，複数の言語資源を一つにまとめた評価タスクセットを複数の作業者に割当て，評価タスクセットの多数決を取る方法が取られている[2]．これにより能力の高い作業者が少ない場合でも評価精度を向上させることができる．しかしながら，作業者と作成タスクのミスマッチにより誤った言語資源が作成されると，作成および評価作業のやり直しが多発する．既存研究では，作業者の信頼度推定や，重み付き多数決法，作業者の専門性モデリングに基づくタスク割当て手法が提案されてきた．例えば，Dawid-Skene モデルや GLAD モデルを用いて作業者の信頼度を推定し，重み付き多数決で正答推定を行う手法が用いられている[3]．また，クラウドソーシングにおけるタスク割当て最適化の研究では，作業者の専門性やスキルを事前に評価して最適な割当てを行う方式が提案されている[4]．

しかし，これらの手法の多くは作業途中の能力変動を考慮できず，作業者のスキルを静的に扱ってしまうという課題がある．このため，作業者の能力変化を逐次反映しながらタスク割当てを行う動的な仕組みが求められている．

そこで， 本研究では， 作業者に作成可能なタスクを割当てるために，作業者ごとのタスク難易度を考慮したタスク割当てを行うことで，成果物の正確性を向上， 作業時間を最小化するというアプローチをとった．Deep Knowledge Tracing を用いて， 作業者の作業履歴から次タスクの成否予測を行う． 作業者が新しいタスクに取り組むごとに成否予測を行い，この成否予測を用いて， 逐次的にタスク割当てをする． この手法の実現にあたり， 以下の課題に取り組む必要がある．

**タスクの分類**

対訳評価タスクでは， 評価対象となる言語の単語の種類によってタスクの難易度や作業者の正答率が異なるという単語種別依存性が存在する． 例えば，意味的に類似した単語や形態類似的に近い単語に対する評価タスクは，作業者にとって同様の難易度を持つ場合が多い． したがって， これらの依存性を考慮せずに各タスクを独立したものとして扱うと， 作業者が過去に経験していない単語に対する成否予測が困難となる． 特に， 作業者の履歴が十分に存在しない場合には， 同一タスクの経験がない作業者に対して成否予測を行えず， 作業者能力推定やタスク割当ての精度低下を招く． このため， 原語の単語を単語種別に基づいて分類し， 類似したタスクをグルーピングすることで， 未経験タスクに対しても過去の類似タスクの履歴を活用可能とする必要がある．

**オンラインタスク割当て**

クラウドソーシングにおいて作業者の能力は作業途中で変動するものではないが， 作業開始時点では作業履歴が十分に存在しないため， 作業者能力の推定精度が低下するという課題がある． 静的割当てでは， タスク開始時点で得られた初期の成否予測結果のみに基づいて割当てを決定するため作業の進行に伴って作業履歴が蓄積され， 作業者能力の推定精度が向上した場合であってもその改善が割当てに反映されない． そのため， 作業履歴の更新に応じて作業者能力を逐次的に推定し， 推定結果を即座にタスク割当てに反映するオンラインタスク割当てが必要となる．

# 第2章 クラウドソーシング

## 2.1 クラウドソーシングの概要

クラウドソーシングとは，インターネットを介して不特定多数の人に仕事を依頼すること，もしくはその仕組みのことを指す．クラウドソーシングは，画像のラベリングや文章の翻訳など計算機では作成が比較的困難であり，人間の持つ能力を用いればそれほど難しくはないタスクにおいて一般的に使用されている．Amazon Mechanical Turk(AMT)のような大規模なクラウドソーシングプラットフォームの存在により，インターネットを介して大量の作業者を容易に確保することができる．現在，多言語データの収集にクラウドソーシングが広く用いられており，これにより多言語話者が用例対訳を作成するプロジェクトや，複数の言語におけるテキストの精度を評価する研究が進められている．これらのタスクは，計算機による自動処理が困難であり，人間の判断が不可欠である点に特徴がある．

## 2.2 品質管理手法

クラウドソーシング分野で注目されている研究として，品質管理の方法がある．人間が作業を行うため，能力が低い作業者や意図的に品質を落とす作業者(スパムワーカー)が存在するリスクがあり，常に正確な結果が得られるとは限らない．この問題に対処するため，品質管理手法の研究では主に以下の 2 つのアプローチがある[5][6][7]．

**作業結果を集約して全体の品質を向上させる方法**

複数の作業者に同じタスクを割当て，冗長性を利用し最終的には多数決で結果を決める方法が主に取られている．この方法では，作業者の能力が高い場合には正しい答えを導けるが，作業者の能力が低い場合では正しい答えを導き出すことは困難である．

**個々の作業結果の品質を向上させる方法**

報酬の設定，タスクのデザイン，作業者の選定などを行うことで，作業者によるタスクの実行結果そのものの品質を向上させる方法．特に，高い能力を持つ作業者を事前に抽出し，タスクを割当てる方法は，低い能力の作業者やスパムワーカーの排除することができるため，作業結果の品質向

上が期待される．

## 2.3 タスク割当て

近年，クラウドソーシングは言語資源構築，社会科学調査，モバイルアプリ価など様々な領域で利用される一般的なデータ収集基盤となっている．その中で，タスク割当ての設計は成果物の品質と作業効率を左右する重要な課題である．

### 2.3.1 クラウドソーシングのデータ収集と品質管理

Sakti らはインドネシア諸語の並列音声コーパス構築にクラウドソーシングを用い[1]，地田らも超問題を用いたインドネシアの低資源言語の対訳辞書構築においてクラウドソーシングの品質管理手法を検討している[2]．また，クラウドソーシングにおける品質管理やコスト最適化の基礎研究として，Karger らによる予算制約下での最適タスク割当て[3][4]，および人間との AI の相互作用による品質改善の取り組み[5]が挙げられる．さらに，鹿島らがクラウドソーシングと機械学習の関係性を整理し，ワーカーの能力ばらつきやラベル品質を統計モデルで推定する重要性を指摘している[6]．特に，人間と機械学習を統合したハイブリッド型の学習枠組みや，信頼度推定モデルによる品質向上手法が紹介されている．金地らはワーカー特性が品質に与える影響を分析している[7]．
これらの研究は，クラウドソーシングが多様なタスク収集の基盤として有用である一方で，適切なタスク割当てと品質管理が不可欠であることを示している．

### 2.3.2 最適化割当てに基づくタスク割当て

タスク割当て手法そのものに関する研究も数多く存在する．Punitha らの研究では，クラウドソーシングにおける割当て問題が，どのワーカーにどのタスクを割当てるかという組み合わせの数が急激に増大するため，制約の設定によっては計算量が爆発的に増加する難しい最適化問題であることを明らかにした[8]．特に，ワーカーのスキル，タスクの難易度，時間的制約，品質要求などの要因を同時に考慮すると問題規模が指数的に大きくなるため，一般的に最適解の計算は容易ではない．また，Machado らはタスクや作業者が一定ではなく時間と共に新たに到着，離脱する動的な環境が一般的であると指摘し，そのような状況でも即時に対応できるよう，各時点で利用可能な情報に基づいてタスクを逐次的に割当てるオンライン型手法を提案している[9]．Yu らは，作業者の稼働時間や能力情報に基づきグループを形成し，ハンガリアン法を活用した協力的タスク

割当てモデルを提案し，効率を向上させた[10]．さらに Patel らは，クラウドコンピューティング環境におけるジョブスケジューリングにハンガリアン法を適用し，資源割当ての最適化を示している[11]．

### 2.3.3 信頼性・コンテキストを考慮したタスク割当て

一方，クラウドワーカーの信頼性や利用環境を考慮市チアタスク割当ても重要な研究領域である．Miao らは，クラウドワーカーには熟練度や信頼性にばらつきがある点に着目し，ワーカーごとの信頼度とタスク側の品質要求を数理モデルとして定式化した上で，両者を満たすワーカーに優先的にタスクを割当てる Quality-aware Online Task Assignment を提案した[12]．この手法は，単に空いているワーカーに割当てるのではなく，品質を確保できるワーカーを選ぶことで，全体の成果物品質を向上させることを目的とする．Wang らは，マルコフモデルと協調フィルタリングを組み合わせた MCTR モデルを用いてシステム効率を向上させる割当て手法を提案している[13]．さらに，Hettiachchi らの研究では，スマートフォン，タブレット，PC といった複数デバイス上でのクラウドソーシング作業を比較し，ワーカーの利用デバイスがタスク遂行能力に影響を与えることを示した[14]．

総じて，既存研究は

(1) クラウドソーシングによるデータ収集と品質管理

(2)最適化割当てに基づくタスク割当て

(3)信頼性・コンテキストを考慮したタスク割当て

といった多角的な観点からタスク割当て問題を扱い，クラウドソーシングの品質向上と効率的なタスク分配に貢献している．

しかし，これらの研究の多くはワーカーの能力を固定的な値として扱う前提に立っておりスキルが時間とともに変化することを十分に考慮していないという問題がある．実際のクラウドソーシングでは，新規ワーカーには初期能力が存在しない場合があり，また作業の進行に伴って特定のタスクに対する理解度が向上するなど，ワーカーのスキルは動的に変動するにも関わらず，このようなスキル変動をリアルタイムに推定し，タスク割当ての判断に直接反映する枠組みはほとんど存在ない．

この点は，スキルがタスク成功率に大きく影響する言語資源構築のようなタスクにおいて特に重要であり，動的に変化するワーカー能力を推定して活躍できるタスク割当て手法の必要性を示している．

# 第3章 対訳辞書作成のためのワークフロー

## 3.1 ワークフロー

本研究では，クラウドソーシングを用いた低資源言語対訳辞書構築を対象として，同一研究室における先行研究「Quality Control for Crowdsourced Bilingual Dictionary in Low-Resource Languages[2]」で用いられたデータ収集ワークフローを基盤として実験を行う．本節では，本研究で扱うデータがどのような手順で生成されているかを明確にするため，対訳作成および評価の流れについて説明する．

本研究で使用するデータセットの取得に用いられたワークフローは，対訳作成タスクと対訳評価タスクの二段階から構成される(図 1)．本ワークフローは，クラウドソーシング環境における作業者のスキルばらつきや判断誤りを考慮し，対訳品質を安定的に確保することを目的として設計されている．

まず，対訳作成タスクにおいて，作業者(対訳作成タスク作業者)は与えられた原語に対して対象言語の対訳を一つ生成する．この段階では，各作業者が自身の言語知識や経験に基づいて自由に対訳を作成するため，作成される対訳の品質には個人差が生じる可能性がある．

次に，作成された各対訳に対して，複数名の別作業者(対訳評価タスク作業者)による対訳評価タスクが実施される．評価者は，提示された対訳が意味的に正しいか否かを判定し，正解または不正解の二値で評価を行う．このように，一つの対訳作成結果に対して複数回の評価タスクを割当てることで，単一の評価者による誤判定や主観的判定の影響を緩和し，評価に冗長性を持たせている．

対訳の最終的な採否は，これら複数の評価結果に基づく多数決によって決定される．具体的には，作成された対訳が評価タスクにおいて「正解」と判定された結果が多数決により多数であった場合に，当該対訳を最終的な対訳結果として採用する．評価結果において不一致が生じた場合や，多数決の結果として正答性が確認できない場合には，当該対訳は採用されない．

採用されなかった言語については，再度対訳作成タスクが実行され，新たな対訳候補が生成される．この後，同様に複数回の評価タスクを経て採否判定が行われる．以上の処理を，全ての原語について最終的な対訳が確定するまで繰り返すことで，クラウドソーシング環境においても一定水準以上の品質を持つ対訳

辞書を構築することが可能となる.

本研究では, このワークフローによって得られた作業履歴および評価結果を入力データとして用い, 後述する Deep Knowledge Tracing モデルによる成否予測およびタスク割当て手法の評価を行う.

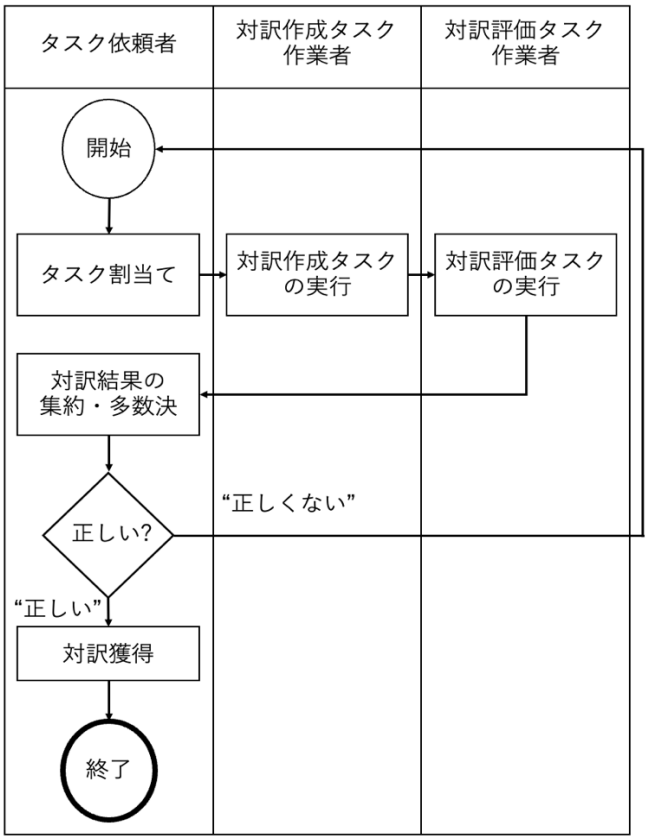


図 1: 対訳辞書作成タスクのワークフロー

## 3.2 タスク内容

本研究では扱うクラウドソーシングタスクは, 対訳作成タスクと対訳評価タスクの 2 種類から構成される. 本説では, それぞれのタスク内容について説明する.

- 対訳作成タスク

  対訳作成タスクでは, 作業者に対して言語となる単語が提示され, その単語に対応する対訳語を入力する作業を行う. 対象言語は, インドネシア語を原

語とし，ミナンカバウ語への対訳作成を対象とする．

- 対訳評価タスク

  対訳評価タスクでは，対訳作成タスクによって作成された対訳に対して，その正誤を判定する作業を行う．評価者には，原語と作成された対訳が提示され，「正しい」または「正しくない」の 2 値で評価を行う．

本研究の割当てで使用するのは，以上のクラウドソーシングタスクから得られた，対訳評価タスクの作業履歴を用いてタスク割当てを行う．実験で扱う，対訳評価タスクのデータは，原語，対訳，評価，作業者，作業者の信頼度，時系列情報というふうに構成されている(表 1).

表 1: 対訳評価タスク結果の一部

| sourceword | targetword | evaluation | worker | level | evaluated |
|---|---|---|---|---|---|
| ada | ado | CORRECT | 作業者 1 | 5 | 2020-12-10 08:37:38 |
| ada | lai | CORRECT | 作業者 1 | 5 | 2020-12-08 10:53:25 |
| ada | lai, ado | CORRECT | 作業者 1 | 5 | 2020-12-08 10:56:09 |
| adalah | adalah | WRONG | 作業者 1 | 5 | 2020-12-08 10:56:12 |
| adalah | adolah | CORRECT | 作業者 1 | 5 | 2020-12-08 10:56:17 |
| adalah | artinyo | WRONG | 作業者 1 | 5 | 2020-12-08 10:56:20 |
| adalah | iko | WRONG | 作業者 1 | 5 | 2020-12-08 10:56:23 |
| adalah | inyotu | WRONG | 作業者 1 | 5 | 2020-12-07 23:40:29 |
| adalah | iyolah | CORRECT | 作業者 1 | 5 | 2020-12-07 23:40:30 |
| adalah | yaitu | WRONG | 作業者 1 | 5 | 2020-12-07 23:40:30 |
| adil | adia | CORRECT | 作業者 1 | 5 | 2020-12-07 23:40:32 |

# 第4章 単語分類手法

## 4.1 タスク分類

本研究では，作業者のタスク成否を適切に予測し，効率的なタスク割当てを行うために，各タスクをあらかじめ分類し，タスクの難易度構造を明示的に導入する．

本研究で用いる DKT モデルは，作業履歴を知識タグ単位で扱い，各知識タグに対する習熟度を時系列的に推定するモデルである．そのため，各タスクがどの知識に対応するかを示す知識タグの定義は，DKT を適用する上で必要不可欠な前処理となる．しかし，本実験で使用する実際のクラウドソーシングにおける低資源言語対訳辞書構築タスクでは，知識タグなどによるタスクの体系的な分類がなされておらず，タスクは個々独立した作業単位として存在している．このため，このクラウドソーシングの作業履歴を DKT モデルに適用するためには，実験に用いる実タスクに対して新たに知識タグを設計し，付与する必要がある．

そこで本研究では，タスクを知識タグとして定義し，以下の 2 種類の分類方法を用いてタスクを分類する．

- 分散表現ベース分類
- 形態類似度ベース分類

これらの分類結果を，知識タグとして用いて DKT の成否予測に使用する．

### 4.1.1 分散表現ベース分類

分散表現ベース分類では，タスクの原語(インドネシア語)の意味的類似度に基づいてタスクを分類する．まず，各言語を Word2Vec によりベクトル化し，得られた分散表現を用いて単語間の意味的距離を表現する．本研究では，インドネシア語の文章 1,000,000 文からなるコーパスを用いて Word2Vec を学習させて分散表現を獲得した．分散表現で取得した原語数は 1,001 語である．これらの分散表現に対して k-means 法を適用し，意味的に近い単語を同一クラスタに分類する．意味的に類似した語彙は文脈的使用傾向や語義的役割が共通するため，それらを対象とする対訳評価タスクにおいて作業者の正答傾向が類似すると仮定できる．この際，本研究ではクラスタサイズの極端に小さいものが発生するのを防ぐため，最小クラスタサイズに下限値を設け，全てのクラスタが 5 以上の

原語を含むように制約を加えた．これにより，特定のタスクのみが属する小規模クラスタの発生を抑制し，DKT モデルにおける学習の安定性を確保する．

クラスタ数については，全てのクラスタサイズが 5 以上となる最大のクラスタ数を探索した結果，10 クラスが条件を満たす最大のクラスタ数であることが確認された．そのため本研究では分散表現ベース分類におけるクラスタ数( 知識タグ数)を 10 と設定した．

表 2 に示すように，分散表現ベース分類ではクラスタサイズに一定のばらつきが見られる．特に，意味的に汎用性の高い語彙が多く含まれるクラスタでは

表 2: 分散表現ベース分類の各クラスタ要素数

| クラス | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 要素数 | 28 | 39 | 157 | 43 | 106 |
| クラス | 5 | 6 | 7 | 8 | 9 |
| 要素数 | 123 | 75 | 146 | 274 | 10 |

表 3: 分散表現ベース分類 クラス別の代表語と分類傾向

| クラス | 代表する原語 | 大まかな分類傾向 |
|---|---|---|
| 0 | bangun, berangkat, besok<br>datang, dekat | 日常動作・移動・<br>時間 |
| 1 | akan, akhirnya, belum<br>berani, berhasil | 助動詞・状態変化・心的状態 |
| 2 | ada, ambil, bekerja<br>berada, berakhir | 基本動詞・存在 |
| 3 | awal, beberapa<br>berikutnya, depan | 順序・数量・位置・時間 |
| 4 | adil, alam, alasan<br>apakah, apapun | 疑問・論理語・抽象概念 |
| 5 | anaknya, ayahnya<br>belakang, berdiri | 人物・身体動作・関係 |
| 6 | adalah, antara, asing<br>atas, bagi | 機能語・関係 |
| 7 | aman, aneh, anggur<br>bagus, bahagia | 評価・感情・性質 |
| 8 | alih, anggap, angkat, artinya,<br>astaga | 認知動詞・談話的語 |
| 9 | panggilan, pukul<br>selamatkan, sulit | 出来事・否定的表現 |

らつきが見られる．特に，意味的に汎用性の高い語彙が多く含まれるクラスタでは要素数が大きくなる一方で，意味的に限定的な語彙から構成されるクラスタでは要素数が小さくなる傾向が確認できる．このようなクラスタサイズの不均衡は，語彙の意味分布そのものを反映した結果であると考えられる．

表 3 に示す代表語から，各クラスタは動作，状態，評価，論理語，機能語といった語義的役割の共通性に基づいて形成されていることが確認できる．このことから，意味的に整理されており，分散表現が語彙の近さを適切に捉えられていると考えられる．

### 4.1.2 形態類似度ベース分類

形態類似度ベース分類では，言語の文字列構造に着目し，語形の類似性に基づいてタスクを分類する．インドネシア語は，語幹(root word)に対して接頭辞・接尾辞・接中辞などの派生要素が付与されることで多様な語形変化が生じる言語である．そのため，意味的には異なる場合でも，語幹や接辞構造が類似する語彙間には翻訳やその難易度に共通性が存在する可能性がある．このような語形的特徴を捉えるため，本研究では原語間の形態的距離としてレーベンシュタイン距離を用いる．挿入・削除・置換の最小操作回数により語形の近さを定量化し，得られた距離行列に対して k-means 法によって 1001 語の原語のクラスタリングを行う．

クラスタ数については，k の値を複数設定してクラスタリングを行い，その結果をインドネシア語母語話者に評価してもらうことで決定した．その結果，10 クラスでのクラスタリングが語形的なまとまりとして妥当であるとの判断が得られた．以上により，形態類似度ベース分類においてもクラスタ数を 10 と設定した．

表 4 に示すように，各クラスタの要素数は，43 語から 158 語の範囲に分布しており，極端に小さいクラスタは存在しない．これは，形態類似度に基づく分類が，原語集合を比較的均一な粒度で分割できていることを示している．

表 5 に示す代表語から，各クラスタは接頭辞(例: a-, be-, ber-, di-, me-)や接尾辞(-nya)といった語形構造の共通性に基づいて形成されていることが確認される．

特に，be-, be-接頭辞を持つ動詞群が原語群に多く含まれておりそれらがさらに複合語や長語形といった特徴で分類されており，インドネシア語の語形成規則

を反映したクラスタリングが実現されている.

表 4: 形態類似度ベース分類の各クラスタ要素数

| クラス | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 要素数 | 158 | 115 | 95 | 128 | 64 |
| クラス | 5 | 6 | 7 | 8 | 9 |
| 要素数 | 95 | 154 | 82 | 43 | 67 |

表 5: 形態類似度ベース分類 クラス別の代表語と分類傾向

| クラス | 代表する原語 | 大まかな分類傾向 |
|---|---|---|
| 0 | ada, adalah, akan<br>alam, aman | a-で始まる短語 |
| 1 | belajar, benar, berada<br>berani, berapa | be-接頭辞動詞群 |
| 2 | begitulah, bekerja, berakhir<br>berarti, berbagi | ber-接頭辞動詞群 |
| 3 | anggur, bebas, begini<br>begitu, beli | 類似の語形 |
| 4 | ditemukan, diterjemahkan<br>kebetulan, melepaskan | di-/me-受動・派生形 |
| 5 | akhirnya, alasan, anaknya<br>artinya, ayahnya | -nya など派生語尾 |
| 6 | adill, alih, ambil<br>aneh, ayo | 母音始まり短語 |
| 7 | beberapa, berbicara<br>bercinta, bergabung | ber-+長語形 |
| 8 | bagaimanapun, berbohong<br>berhubungan, bersama | 複合語・長語形 |
| 9 | belakang, benarkah<br>berangkat, berbahaya | be-+派生形 |

# 第5章 作業者能力推定

## 5.1 DKT の概要

Knowledge Tracing(KT)は，学習者の過去の回答履歴から知識状態を推定し，将来提示される問題に対する正答確率を予測することを目的とした手法である[15].

KT では，学習者がどの問題をどの順序で解き，正解・不正解をどのように経験してきたかという作業履歴を用いて，学習の進行や能力変化をモデル化する．これらの推定結果は，教育分野において学習支援や問題推薦などに広く利用されてきた．従来の代表的な KT 手法として，Hidden Markov Model に基づく Bayesian Knowledge Tracing(BKT)が知られている[16]. BKT では，各スキルを独立した確率変数として扱い，学習・未学習間の遷移を確率的にモデル化する．しかし，スキル間の相互依存や非線形な学習過程を表現することが難しいという課題がある．

これらの課題に対して提案されたのが Deep Knowledge Tracing(DKT)である．DKT は，Piech らによって提案され，深層学習モデルを用いることで，学習履歴の時系列依存やスキル間の関係性を柔軟に学習できる点を特徴とする[17].

DKT では，学習者の作業履歴を時系列データ

$$\{(q_1, a_1), (q_2, a_2), \cdots, (q_t, a_t)\} \tag{1}$$

として表す．ここで$q_t$は時刻$t$に提示された問題，$a_t \in \{0,1\}$はその正誤結果を表す．各時刻の入力は，「どの問題に対して正解または不正解であったか」を符号化したベクトルとして表現される．問題種別数を$K$とすると，入力ベクトル$x_t \in \mathbb{R}^{2K}$は正解および不正解用の one-hot 表現を用いて構成される．本研究では，この「問題種別」を知識タグ(ProblemID)として定義する．知識タグとは，タスクが要求する知識やスキルの種類を表す識別子であり，各タスクはあらかじめどれかの一つの知識タグに対応付けられている． したがって，本研究における問題$q_t$は，教育分野における設問そのものではなく，特定の知識タグを有するタスクを意味する．

DKT では，入力系列$x_t$を Long Short-Term Memory(LSTM)に入力し，隠れ状態$h_t$を次式により更新する．

$$h_t = LSTM(x_t, h_{t-1}) \tag{2}$$

次時刻における各問題種別に対する正答確率は,

$$\hat{p}_{t+1} = \sigma(Wh_t + b) \tag{3}$$

によって計算される[17].

学習時の損失関数は, 時刻$t+1$に実際に出題された問題種別$q_{t+1}$のみを対象とした 2 値交差エントロピーで定義される. このように出題された問題種別に対応する出力のみで誤差を計算する点が DKT の特徴である.

この定式化により, DKT は履歴全体を通じた長期的な依存関係を学習できる一方で, 正解が連続した場合であっても予測確率が必ずしも単調に増加するとは限らない. この挙動は DKT の既知の性質として報告されており, 時間的一貫性の欠如や不安定なスキル推定として議論されている[18][19].

本研究では, DKT を教育分野の学習者モデルとしてではなく, クラウドソーシング環境における作業者のタスク成否予測モデルとして用いる.

## 5.2 タスク分類間での比較

本節では, タスク分類手法の違いが DKT による成否予測性能に与える影響を詳細に分析する. 具体的には, 分散表現ベース分類と形態類似度ベース分類を用いて構築した DKT モデルについて, 予測精度指標を比較することで, 各分類手法が作業者の知識状態推定にどのような差をもたらすかを検証する.

本実験では, 各作業者について 1450 件の作業履歴を用いて DKT モデルを学習した. 評価時には, 7 章の実験で使用している fold1 に含まれるタスク 400 件に対する正答確率を予測した なお, 本節ではタスク割当ては行わず, DKT モデル単体の予測性能の比較に限定して評価を行う.

表 6 に示すように, Accuracy の平均値は, 分散表現ベース分類が 0.697, 形態類似度ベースが 0.663 となった. これは分散表現ベース分類を用いた場合, 形態類似度ベースを用いた場合に対して約 3.4%の正誤を正しく予測できていることを意味する. 両者の差は, 小さな差ではあるものの, 20 名の作業者に渡って一貫して観測されている点が重要である. また, Accuracy の標準偏差は, 分散表現ベース分類が 0.102, 形態類似度ベース分類が 0.143 であり, 形態類似度ベース分類の方が作業者間でのばらつきが大きい. これは, 形態類似度ベース分類では, 作業者によって予測精度が安定しにくい可能性を示唆している.

AUC は, 予測確率の大小関係が正しく付けられているかを評価する指標であり, 0.5 はランダム予測に相当する. 分散表現ベース分類の AUC の平均値は

0.519, 形態類似度ベース分類は 0.487 であった. いずれの分類手法においても AUC は 0.5 付近の値を示しているが, 分散表現ベース分類の方が高く, 正答確率の相対的な大小関係をより適切に捉えていることが分かる.

次に, 予測確率そのものの妥当性を評価するため, Logloss および Brier score に着目する. Logloss は, 誤った予測を高い確率で行った場合に大きなペナルティを与える指標であり, 値が小さいほどいい. 一方, Brier score は, 予測確率と実際の正誤との差の二乗平均であり, これも小さいほど良い. 分散表現ベース分類における Logloss の平均値は 0.602, 形態類似度ベース分類では 0.623 であった. また, Brier score は, 分散表現ベース分類が 0.206, 形態類似度ベース分類が 0.217 となった. これらの結果から, 分散表現ベース分類を用いた場合, 正答確率の推移が少しだが適切に校正されていることがわかる.

以上の結果から, 分散表現ベース分類は形態類似度ベース分類と比較して, Accuracy, AUC, および確率予測精度の全ての指標において優位な性能を示すことが明らかとなった. また, 分散表現ベース分類では作業者間のばらつきが比較的小さく, 予測性能が安定している点も特徴的である. このことは, 語の意味的近さを反映したタスク分類が, 作業者の潜在的な知識状態をより適切にモデル化できる可能性を示している.

表 6: タスク分類手法間における DKT の予測性能比較

| タスク分類手法 | Accuracy (mean ± std) | AUC (mean ± std) | Logloss (mean ± std) | Brier score (mean ± std) |
|---|---|---|---|---|
| 分散表現ベース分類 | 0.697 ± 0.102 | 0.519 ± 0.126 | 0.602 ± 0.098 | 0.206 ± 0.044 |
| 形態類似度ベース分類 | 0.663 ± 0.143 | 0.487 ± 0.129 | 0.623 ± 0.112 | 0.217 ± 0.050 |

# 第6章 作業者能力逐次推定を用いたオンラインタスク割当て

本章では，第 5 章で構築したタスク分類(知識タグ)および DKT よって求められた作業者の成否予測結果を用いたタスクの難易度を考慮したタスク割当て手法について説明する．

クラウドソーシングにおけるタスク割当ては，成果物の品質と作業効率を左右する重要な要素であり，作業者ごとの能力差やタスクごとに求められる能力を考慮した割当てが求められる．本研究では，DKT によって推定される作業者の予測正答確率を用い，最適化問題としてタスク割当てを定式化する．また，本研究では，割当てアルゴリズムとしてハンガリアン法と拡張ハンガリアン法を用い，割当て方式に応じてその適用方法を切り替えるため，それについても述べる．

## 6.1 タスク割当ての定式化

クラウドソーシングにおけるタスク割当ては，利用可能なタスクを作業者へ適切に配分することにより，作業効率および成果物品質を最大化することを目的とする．本研究では，第 5 章で定義したタスク分類と，作業者のスキルに関する情報を入力として用いる．

作業者集合を$W = \{w_1, w_2, \cdots, w_M\}$，タスク集合を$T = \{t_1, t_2, \cdots, t_N\}$とする．各タスク$t_j$は，第 5 章で定義した知識タグを持つ．DKT により，作業者$w_i$がタスク$t_j$を正しく遂行する確率$p_{ij}$が推定される．

作業者$w_i$にタスク$t_j$を割当てた場合の効用を$U_{ij}$とし，本研究では

$$U_{ij} = p_{ij} \tag{4}$$

と定義することで，成功確率が高い割当てほど高い効用を持つものとする．

次に，タスク割当てを表す二値変数$x_{ij}$を次のように定義する．

$$x_{ij} = \begin{cases} 1 & \textit{作業者}w_i\textit{にタスク}t_j\textit{を割当てる場合} \\ 0 & \textit{それ以外} \end{cases} \tag{5}$$

この時，タスク割当て問題は，次の目的関数を最大化する最適化問題として定式化される．

$$\max \sum_{i \in W} \sum_{j \in T} U_{ij} x_{ij} \tag{6}$$

制約条件として，各タスクは 1 人の作業者のみに割当てられるよう，

$$\sum_{i \in W} x_{ij} = 1 \ \forall j \in T \tag{7}$$

このような制約を課す．また作業者が同時に担当できるタスク数には上限があるため，

$$\sum_{j \in T} x_{ij} \leq MaxTasks_i \ \forall i \in W \tag{8}$$

とする．さらに割当て変数は二値であることから，

$$x_{ij} \in \{0,1\} \tag{9}$$

を満たしている．以上の定式化は，タスクを割当てるか否かを 0 または 1 で表す最適化問題である．特に，各作業者が一つのタスクのみを担当し，各タスクが 1 人の作業者に割当てられる場合には，この問題はハンガリアン法として知られる割当てアルゴリズムによって効率的に解くことが可能になる．

## 6.2 ハンガリアン法

ハンガリアン法は，割当て問題を解決するための代表的な最適化手法の一つである．割当て問題とは，複数のリソースを複数のタスクに対応付け，全体のコストが最小となるように割当てを決定する最適化問題である．本研究では，作業者をリソース，タスクを割当て対象として扱う．

ハンガリアン法では，作業者とタスクの組み合わせごとコストを定義し，それらを要素とするコスト行列を作成する．そのコスト行列に対して以下の手順を適用することで全てのタスクを作業者に割当てる際の最小コスト割当てを求めることができる．図 2 に，ハンガリアン法の処理の流れを示す．

手順は以下の通りである．

1. 各作業者と各タスク間のコスト(成否予測結果)を要素とするコスト行列を作成する．
2. コスト行列の各行について，その行における最小値を求め，行内の全要素から引く．
3. 各行についても同様に，その列における最小値を求め，行内の全要素から引く．

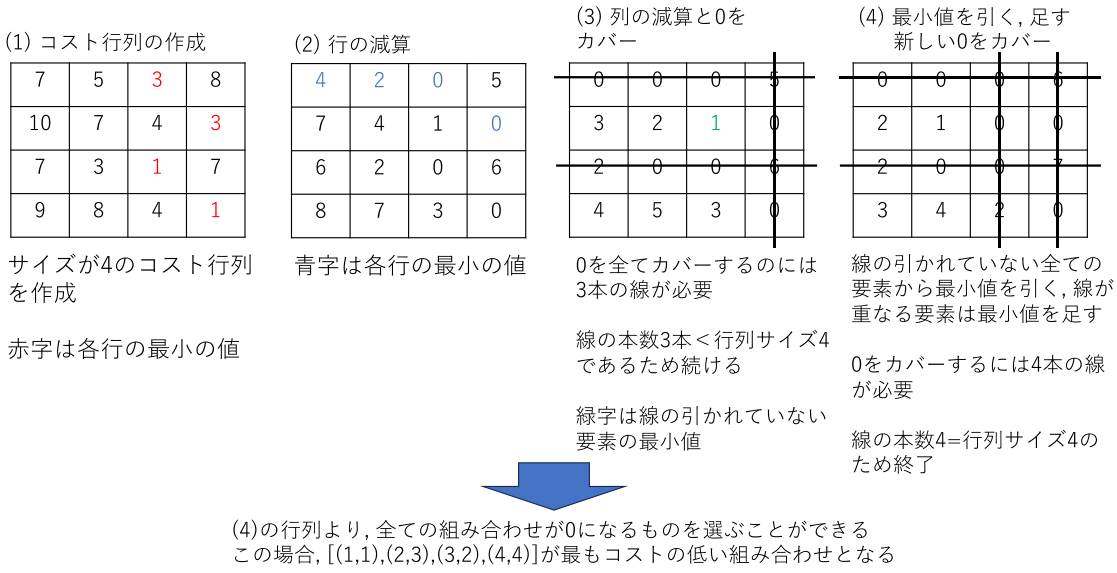


図 2: ハンガリアン法の処理の手順

4. 行または列に引いた水平線・垂直線によって，行列内の全ての 0 を覆うために必要な最小本数の線を引く． この時，引いた線の本数がコスト行列の行数(または列数)以上であれば，割当てが可能であるため処理を終了する．
5. 線が引かれていない要素の中から最小値を求め，その値を線が引かれていない全ての要素から引く．また，線が交差している要素には同じ値を加える．
6. 再び手順(4)に戻り，全ての 0 を覆うために必要な線の本数が行列サイズ以上になるまで，手順(4), (5)を繰り返す．

以上の手順により，最終的に各行および各列から重複なく選択された，コストが 0 となるセルの組み合わせが，ハンガリアン法における最適な割当て解となる．

## 6.3 拡張ハンガリアン法

クラウドソーシング環境では，1 人の作業者が複数のタスクを担当することが一般的であり，作業者数とタスク数が一致しない場合が多い．このような状況では，作業者とタスクが一対一で対応することを前提とした標準的なハンガリアン法をそのまま適用することはできない．そこで本研究で比較手法として使用する静的な割当て手法には，コスト行列の構成を工夫することで，一対多の割当てを可能とする拡張ハンガリアン法を用いる．

まず，各作業者と各タスクの組み合わせに対して，成功確率に基づくコストを計算し，作業者×タスクのコスト行列を作成する．この時，タスク数が作業者数より多い場合，行列は長方形となる．ハンガリアン法を適用するためには正方行列が必要であるため，コスト行列のパディングを行う．具体的には，各作業者が担当可能な最大タスク数に応じて，作業者の複数の仮想的な作業者に複製する．これにより，タスク数と作業者数が一致するように調整する．複製された仮想作業者は，元の作業者と同一の予測正答確率をタスクに対して持っているものとして扱う．一方で，行列サイズが複製をした後にまだ正方行列でない場合は，ダミータスクを追加することで正方行列を構成する．ダミータスクに対応するコストは 0 とすることで，効用の増減に影響を与えないようにする．その結果，ハンガリアン法の解としてダミータスクが選択された場合は，実タスクが割当てられなかったことを意味する．このように，作業者の複製およびダミータスクによるパディングによって構成された正方行列に対してハンガリアン法を適用することで，1 人の作業者に複数のタスクを割当てる一対多の割当てを実現する(図 3).

本研究では，この拡張ハンガリアン法を，後述する静的 DKT 割当てと履歴正答率ベース割当てに適用する．これらの手法では，作業者の成功確率を割当て前に一度だけ推定し，複数のタスクをまとめて割当てるため，一括割当てが可能な拡張ハンガリアン法が適しているためである．次節で述べる，本研究の提案手法であるオンラインタスク割当てでは，タスク実行ごとに成功確率を更新し，逐次的に割当てを行うため，ハンガリアン法を用いる．

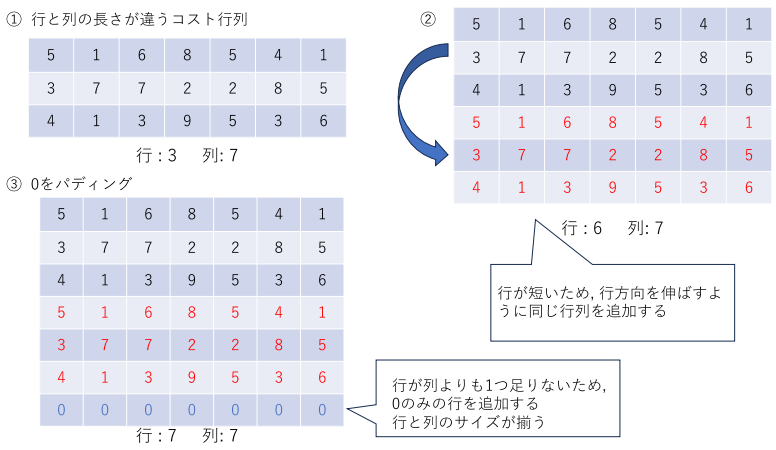


図 3: 拡張ハンガリアン法のコスト行列拡張方法

## 6.4 オンラインタスク割当てプロセス

本研究におけるオンラインタスク割当てプロセスでは，作業の進行に伴って作業者の能力を逐次的に推定し，その推定結果を即座に次のタスク割当てに反映することを目的とする．クラウドソーシング環境では，作業開始時点で作業者の能力を十分に把握できない場合が多く，作業履歴の蓄積に応じて能力推定を更新する仕組みが重要となる．本手法では，Deep Knowledge Tracing(DKT)を用いて作業者の知識状態を推定し，その結果に基づくオンラインタスク割当てを実現する．

まず，各作業者について，過去の作業履歴を用いて DKT モデルを事前に学習する．これらの DKT モデルはオンライン割当て中には再学習せず，割当てフェーズでは固定されたモデルとして用いる．したがって，本手法における「オンライン」とは，作業結果を逐次的に割当てに反映することを示しており，モデルパラメータを逐次更新するオンライン学習は行なっていない．オンライン割当ての処理は，割当て対象となるタスク集合を複数のバッチに分割し，バッチごとに以下の手順を繰り返すことで実現される．まず，各作業者の現時点における作業履歴(知識タグと正誤の系列データ)を入力として，事前学習済みの DKT モデルにより，各知識タグに対する成功確率を推定する この成功確率は，作業者の能力とタスクの難易度を反映した指標として用いられる．

次に，作業者とタスクの組み合わせごとに推定された成功確率を要素とする行列を構築し，その補数(1-成功確率)をコストとしたコスト行列を作成する．このコスト行列に対してハンガリアン法を適用することで，割当て全体のコストが最小となるタスク割当てを決定する．これにより，成功確率の高い作業者に適切なタスクが割当てられるよう設計されている．

タスク実行後には，各作業者が割当てられたタスクの成否結果を取得し，これを新たな作業履歴として当該作業者の履歴に追加する．更新された履歴は次バッチの入力として用いられ，DKT による成功確率推定が再度行われることで，作業の進行に応じた能力推定の逐次的な更新が実現される．一方で，DKT モデル自体のパラメータ更新は行わないため，能力推定の変化は作業履歴の更新によって生じる．

以上のように，本研究のオンラインタスク割当てプロセスは，「作業者履歴の更新に基づく能力逐次推定」と「推定結果を用いた即時的なタスク割当て」を組み合わせた方式である．

このような動的な割当ての流れを図 4 に示す. N 問目までの作業結果に基づいて作業者のスキル推定が更新され，その結果が N+1 問目以降のタスク割当てに反映される様子を模式的に表している.

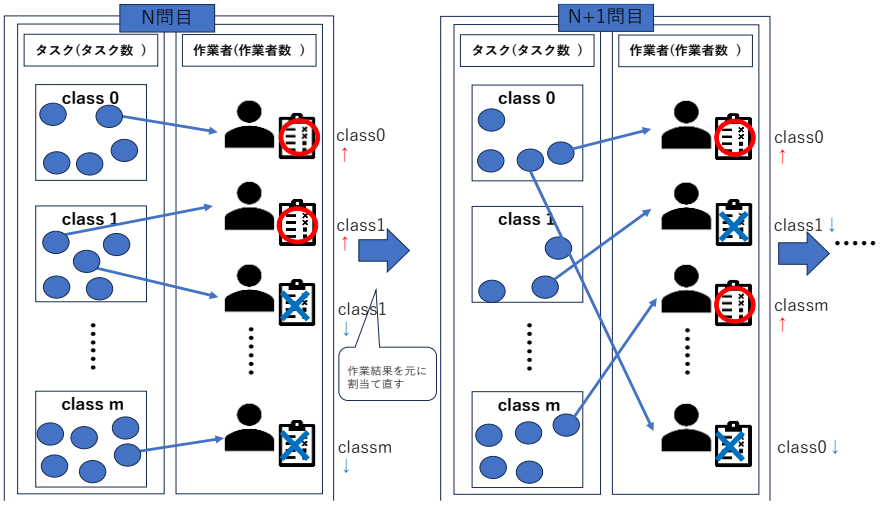


図 4: オンラインタスク割当て

# 第7章 評価

## 7.1 評価データ

本研究では，提案手法の有効性を評価するため，クラウドソーシングにより作成されたインドネシア語-ミナンカバウ語対訳データの評価履歴を用いる．本データセットは，インドネシア語の原語とミナンカバウ語の対訳候補の組に対して，対訳として正しいか否かを評価者が判定するタスクから構成されている．各評価者には，一人当たり約 5,600 件の対訳評価タスクが割当てられており，各タスクについて CORRECT または WRONG の二値で回答する形式となっている．

元のデータセットは，原語であるインドネシア語を表す sourceword，対訳候補であるミナンカバウ語を表す targetword，作業者による評価結果である evaluation，作業者名を表す worker，作業者の信頼度を表す level，および評価時刻を示す evaluated の 6 つの属性から構成されている(表 1)．

本研究では，これらの作業履歴データに加え，level の値が最も高い 2 名の作業者の評価結果のみを用いて構成されたデータセットを対訳として正しい可能性が高いデータ(以下，正解データ)として使用する．評価実験では，元の作業履歴データと正解データの 2 種類を用いる．

DKT を用いた成否予測を行うためには，「作業者 ID」，「知識タグ」，「作業結果の成否」からなる時系列データが必要となる．しかし，元のデータセットにはこれらの情報が直接含まれていないため，本研究では以下の手順で新たに生成した．

まず，作業者 ID については，作業者が 20 名存在するため，各作業者に対して 1~20 の整数を割当てることで作業者 ID を作成した．

次に，知識タグについては，第 5 章で述べた 2 種類のタスク分類手法を用いて作成する．一つ目は分散表現ベース分類であり sourceword(インドネシア語)1,001 語を Word2Vec によりベクトル化した後，最小クラスタサイズが 5 以上となるよう制約を加えた上で k-means 法によるクラスタリングを行った．その結果，sourceword は 10 クラスに分類され，各クラスに対して 0~9 の整数を知識タグとして割当てた．二つ目は形態類似度ベース分類であり，原語間の文字列構造に着目し，レーベンシュタイン距離に基づいて語系の類似度を算出した後，k-means 法によるクラスタリングを行った．こちらについても同様に 10 ク

| 正解データ / 評価結果 | 正解データに存在する場合 | 正解データに存在しない場合 |
|---|---|---|
| Correct | 1 | 0 |
| Wrong | 0 | 1 |

正解データに，りんご→appleがあり，りんご→grapeがない場合

| | sourceword | targetword | evaluation |
|---|---|---|---|
| ① | りんご | apple | 1 |
| ② | りんご | grape | 0 |

①，②のタスクは共に，正解となり「作業結果の成否」の値は1

図 5: 作業結果の成否の作成方法の例

ラスに分類し，各クラスに対応する整数を知識タグとして付与した．

最後に，作業結果の成否は二値で定義する．先述した信頼度の高い作業者のデータで作成した正解データに基づき，作業履歴データにおいて「Correct」と評価された対訳ペアが正解データに含まれている場合，あるいは「Wrong」と評価された対訳ペアが正解データに含まれていない場合，その作業結果は正解であるとみなし，成否を 1 とする．一方で，「Wrong」と評価された対訳ペアが正解データに含まれている場合，または「Correct」と評価された対訳ペアが正解データに含まれていない場合は，不正解とみなし，成否を 0 とする(図 5).

**作業履歴例**

| Unnamed: 0 | sourceword | targetword | evaluation | worker | level | evaluated |
|---|---|---|---|---|---|---|
| 0 | ada | ado | CORRECT | 作業者 1 | 5 | 2020-12-10 08:37:38 |
| 5832 | bangga | gadang hati | CORRECT | 作業者 2 | 5 | 2020-12-19 19:44:38 |

作業履歴を必要な値に置き換え，時系列順に並べる．

| 作業者ID | 知識タグ | 作業結果の成否 |
|---|---|---|
| 5 | 6 | 0 |
| 8 | 7 | 0 |

図 6: DKT で使用するためのデータ修正

以上の手順により，作業履歴データに含まれる worker を作業者 ID に，sourceword を知識タグに置き換え，作業結果の成否を追加したデータを構築し，DKT による成否予測に用いた(図 6). なお，本研究では，分散表現ベース分類および形態類似度ベース分類のそれぞれについて同一の評価手順を適用し，運類手順の違いが割当て性能に与える影響を比較する.

## 7.2 評価方法

本節では，本研究において提案するタスク割当て手法の有効性を検証するために用いた評価指標および評価手順について説明する．クラウドソーシングにおけるタスク割当ては，最終的な成果物の品質だけでなく，作業者間の能力差やタスクの難易度構造，さらに割当て時の判断の妥当性といった複数の要素が複雑に関係している問題である．そのため，本研究では，単一の評価指標に依存するのではなく，複数の観点から割当て結果を評価することを目的とする．具体的には，割当て結果そのものの品質を評価する指標に加え，作業者単位および知識タグ単位での分析，さらに割当て時に用いた成否予測の妥当性を評価する指標を導入することで，多角的な評価を行う.

### 7.2.1 評価前提

提案するタスク割当て手法の有効性を適切に評価するため，以下の前提条件のもとで評価を行う.

まずは，評価はクラウドソーシング環境を想定し，作業者はタスク割当て時点において，各タスクの正解・不正解の結果を事前に知ることはできないものする．すなわち，実際の割当ては，過去の作業履歴や成否予測に基づいて行われる．しかし，本研究では評価基準を明確にするため，成否結果を事前に全て把握している場合に達成可能な理想的なタスク割当てを定義する．この理想割当ては，当該評価データにおいて理論的に達成可能な上限性能を表すものとみなし，実運用において実現可能な手法と比較するための基準としてのみ用いる.

また，評価は 5-fold Cross Validation により行う．評価対象となるタスク集合を 5 分割し，各 fold に含まれるタスクのみを評価用タスクとして用いて，特定のタスク分割に依存しない評価を行う．これにより，割当て手法の性能を複数のタスク構成に対して測定し，その平均を取ることで，安定した評価結果を得る.

以上の前提のもと，本研究では理想割当てを基準として，作業者ごとの正答

率や知識タグごとの正答率などの評価指標を算出し，各タスク割当て手法を相対的に評価する．

### 7.2.2 評価指標

本研究では，提案手法および比較手法の性能を評価するために，以下の 4 種類の評価指標を用いる．

1. 正答率

   正答率は，割合てられたタスクのうち，作業者が正しく回答押したタスクの割合を示す指標である．

   タスク集合を$T$，各タスクに対する正誤結果を$y_t \in \{0,1\}$とすると，正答率は次式で定義される．

$$Accuracy = \frac{1}{|T|}\sum_{t \in T} y_t \tag{10}$$

   この指標は，タスク割当てによって得られる成果物の品質を直接的に評価するための基本的な指標であり，割当て手法間の性能を比較する際の基準として用いる．

2. 作業者ごとの正答率

   作業者ごとの正答率は，各作業者に割当てられたタスクに対する正答率を個別に算出した指標である．

   作業者集合を$W$，作業者$w \in W$に割当てられたタスク集合を$T_w$とすると，作業者$w$の正答率は次式で定義される．

$$Accuracy_w = \frac{1}{|T_w|}\sum_{t \in T_w} y_t \tag{11}$$

   この指標を用いることで，割当て手法が特定の作業者に偏った性能を示していないか，また作業者間の能力差がどの程度存在するかを分析する．

   本研究では，第 7.2.1 節で定義した理想割当ての結果と比較することで，各手法が作業者単位でどの程度効率的な割当てを実現できているかを評価する．

3. 知識タグごとの正答率

   知識タグごとに正答率は，各タスクが属する知識タグ単位で正答率を算出した指標である．知識タグ集合を$K$，知識タグ$k \in K$に属するタスク集合を$T_k$とすると，知識タグ$k$の正答率は次式で定義される．

$$Accuracy_k = \frac{1}{|T_k|}\sum_{t \in T_k} y_t \tag{12}$$

この指標により，タスク集合が一様ではなく，知識的な難易度構造を持つ可能性を考慮した評価を行う．また，本研究では，2 種類のタスク分類手法に基づいて知識タグを付与しているため，知識タグごとの正答率を比較することで，タスク分類方法の違いが割当て性能に与える影響について検討する．

4. 予測と結果の乖離

本研究では，タスク割当て時に用いた成否予測と，実際の割当て結果との整合性を評価するため，割当てに使用した予測と結果の乖離を評価指標として用いる．

作業者$w$に割当てられたタスク集合を$T_w$とし，成否予測モデルにより各タスク$t \in T_w$に対して与えられた予測正答確率を$p_t$とする．この時割当てられたタスクに対する平均予測確率$\bar{p}_w$は，次式で定義される．

$$\bar{p}_w = \frac{1}{|T_w|}\sum_{t \in T_w} p_t \tag{13}$$

また，作業者$w$に対する実際の正答率を$Accuracy_w$とすると，割当てに用いた予測と結果の乖離は，次式で定義される．

$$Gap_w = |\bar{p}_w - Accuracy_w| \tag{14}$$

本指標は，成否予測モデル単体のキャリブレーション性能を評価するものではない．本研究におけるタスク割当てでは，成否予測に基づいてハンガリアン法による全体最適化を行っているため，予測確率が高いタスクであっても，割当ての制約や全体最適性の観点から割当てられない場合がある．そのため，本指標は，成否予測と割当てアルゴリズムを組み合わせた割当て判断全体として，実際に選択されたタスク集合に対する予測が，結果とどの程度整合していたかを評価するための指標として位置付ける．

### 7.2.3 評価手順

本研究では，提案するタスク割当て手法の性能を評価するため，5-fold Cross Validation を用いて評価を行う．放火対象となるタスク集合を 5 分割し，各分割を一つの fold とする．各 fold において，当該 fold に含まれる 400 タスクを評価用タスクとし，残りのタスクは当該評価には使用しない．この手順を 5 回繰り返すことで，全てのタスクが一度ずつ評価用タスクとして用いられる．

評価に際しては，作業者の学種履歴が割当て性能に与える影響を検証するため，1459 件の作業履歴，3000 件の作業履歴の 2 条件を設定する．これらの学習履歴は，DKT による成否予測および履歴正答率ベース割当てにおいて共通に使用し，学習履歴長以外の条件を揃えることで，履歴長の違いが割当て性能に与える影響を比較する．

各 fold に対して，まず第 7.2.1 節で定義した評価前提に基づき，成否結果を事前に全て把握していると仮定した理想割当てを計算する．この理想割当てにより得られた割当て結果は，当該 fold において理論的に達成可能な上限性能を表す基準として用いる．次に，提案手法および比較手法それぞれについて，評価用タスクを作業者に割当てる．成否予測を用いる手法では，過去の作業履歴から得られた成否予測結果を用いてタスクと作業者の対応関係を構築し，ハンガリアン法による最適割当てを行う．履歴正答率ベース割当ておよびランダム割当てについても，同一の制約条件のもとでタスク割当てを行う．割当て後，各作業者が実際に回答したタスクに対する成否結果を取得し，正答率，作業者ごとの正答率，知識タグごとの正答率，および予測と結果の乖離を算出する．これらの評価指標は，理想割当ての結果と比較することで，各割当て手法の性能を相対的に評価するために用いる．以上の処理を全ての fold について実施し，各 fold において得られた評価結果の平均を最終的な評価値とする．この手順により，特定のタスク分割に依存しない，安定したタスク割当て性能の評価を行う．

## 7.3 比較手法

本節では，7.2 節で示した評価手順に基づき比較を行うタスク割当て手法について説明する．提案手法であるオンラインタスク割当てに加え，成否予測の利用方法や割当て更新のタイミングが異なる 3 種類の手法を比較対象として用いる．

### 7.3.1 オンラインタスク割当て

オンラインタスク割当ては，本研究で提案するタスク割当て手法である．本手法では作業者の作業履歴を時系列データとして逐次的に DKT へ入力し，作業の進行に応じて成否予測を更新しながらタスク割当てを行う．

具体的には，各タスクの実行後にその作業結果を作業履歴へ追加し，更新された履歴に基づいて次に実行されるタスクに対する予測正答確率を DKT により推定する．得られた予測正答確率を用いて作業者とタスクの対応関係を表すコ

スト行列を構築し，ハンガリアン法を適用することで，次タスクを決定する．この処理をタスク毎に繰り返すことで，作業途中に生じる作業者の学習や疲労といった能力変動をタスク割当てに動的に反映する．

### 7.3.2 静的 DKT 割当て

静的 DKT 割当ては，成否予測に DKT を用いる点ではオンラインタスク割当てと共通しているが，割当て前に一度だけ成否予測を行い，その結果に基づいてタスク割当てを一括で決定する．

本手法では，評価開始時点で得られている作業履歴を用いて DKT モデルを構築し，全ての作業者・タスクの組み合わせに対する成功確率を推定する．この成功確率に基づいてコスト行列を作成し，拡張ハンガリアン法を適用することで，作業者に複数のタスクをまとめて割当てる．

静的 DKT 割当ては，作業途中の能力変化を考慮しない代わりに，計算コストが比較的低く，一括割当てが可能である．

### 7.3.3 履歴正答率ベース割当て

履歴正答率ベース割当ては，作業者の過去の作業履歴から算出される各知識タグの正答率のみを用いてタスク割当てを行う手法である．

本手法では，各作業者について，評価開始時点までの作業履歴に基づく知識タグごとの正答率を算出し，その正答率に基づいてコスト行列を構築し，拡張ハンガリアン法を適用することでタスク割当てを行う．この手法は，成否予測モデルを用いないため実装が容易であり従来手法として用いられてきた方法である．

### 7.3.4 ランダム割当て

ランダム割当ては，作業者の能力やタスクの難易度に関する情報を一切用いず，タスクを作業者に無作為に割当てる手法である．本手法では，各タスクに対して，作業者の中からランダムに作業者を選択し，各作業者に割当てられるタスク数が 20 件となる制約を満たすように割当てを行う．

ランダム割当ては確率的な手法であり，単一の試行結果に依存すると評価が不安定になる可能性がある．そのため，同一条件下でランダム割当てを 5 回独立に実行し，得られた結果の平均値をランダム割当ての評価値として用いる．本手法は，作業者の履歴情報や成否予測を利用しないため，提案手法および他の比較手法と比較した際のベースラインとして位置付けられる．

## 7.4 評価結果

本節では，第 7.2 節で述べた評価方法に基づき，各割当て手法の性能を比較する．まず，全ての割当て制約および成否結果が既知であると仮定した場合に得られる理想割当てを算出し，これを評価の上限として位置付ける．その上で，オンラインタスク割当て，静的 DKT 割当て，履歴正答率ベース割当て，ランダム割当ての各手法について，正答率の観点から比較を行う．

以下では，まず理想割当ての結果を示し，続いて各割当て手法の評価結果を報告する．

### 7.4.1 理想割当て

割当て手法の性能を相対的に評価するための基準として，理想割当てを導入する．理想割当てとは，割当て後に得られる総正答数が最大となるようにタスクを割当てた場合の割当てである．

理想割当ては，以下の条件で定義する．

- 各タスクは 1 人の作業者のみに割当てられる．
- 各作業者に割当てられるタスク数は 20 件とする．
- 目的は，割当てられたタスクにおける総正答数の最大化である．

この設定は，第 6 章で述べたタスク割当ての定式化において，予測正答確率の代わりに実際の成否結果を用いた場合に相当し，本研究で使用している実際の作業結果における性能の上限を与える．理想割当ての算出にあたっては，各タスクと作業者のくみに対して，正解の場合をコスト 0，不正解の場合をコスト 1 とするコスト行列を構成した．さらに，各作業者に割当てられるタスク数が 20 件となる制約を満たすため，拡張ハンガリアン法の行列複製を用いて一対多の割当て制約を正方行列として表現した．このコスト行列に対してハンガリア

表 7: 理想割当てにおける fold ごとの正答率

| fold | 割当てタスク数 | 正答数 | 正答率 |
|---|---|---|---|
| Fold1 | 400 | 400 | 1.00 |
| Fold2 | 400 | 400 | 1.00 |
| Fold3 | 400 | 400 | 1.00 |
| Fold4 | 400 | 400 | 1.00 |
| Fold5 | 400 | 400 | 1.00 |
| 平均 | - | - | 1.00 |

ン法を適用することで，総誤答数が最小となる割当てを求めた．本処理は，5-fold Cross Validation の各 fold に対して独立に実施した．その結果，すべての fold において，理想割当てによる正答率は 100%となった(表 7).

### 7.4.2 割当て手法ごとの正答率

本節では，理想割当てを性能の上限とした上で，各種割当て手法により得られた正答率を比較する．前節で示した通り，理想割当てはいずれの fold においても正答率 100%を達成しており，本節で扱う各手法の結果は，この上限に対してどの程度の性能が得られたものかを示すものである．評価は，学種履歴長(1450 件/3000 件)および知識タグ付与方法(分散表現ベース分類，形態類似度ベース分類)の組み合わせごとに実施し，5-fold Cross Validation における平均正答率を算出した．以下で，それぞれの条件に対する結果を示す．

まず，学習履歴長 1450・　分散表現ベース分類の場合の結果を表 8 に示す．この条件では，オンラインタスク割当てが平均正答率 0.7020 と最も高い値を示した．一方，静的 DKT 割当ておよび履歴正答率ベース割当てはいずれも 0.6820，ランダム割当ては，0.6754 であった．

次に，学習履歴長 1450 件・形態類似度ベース分類の結果を表 8 に示す．オンラインタスク割当ての平均正答率は，0.6965 であり，静的 DKT 割当ては 0.6835，履歴正答率ベース割当ては 0.6810，ランダム割当ては 0.6754 となった．この条件においても，オンラインタスク割当てが最も高い正答率を示している．

続いて，学習履歴長 3000 件・分散表現ベース分類の結果を表 9 に示す．この場合，オンラインタスク割当ては平均正答率 0.70005 を示し，静的 DKT 割当ては 0.6830，履歴正答率ベース割当ては 0.6825，ランダム割当てじゃ 0.6754 であった．学習履歴長を増加させた場合においても，オンラインタスク割当てが他の手法を上回る結果となっている．

最後に，学習履歴長 3000 件・形態類似度ベース分類の結果を表 9 に示す．この条件では，履歴正答率ベース割当てが 0.6890 と最も高い正答率を示し，静的 DKT 割当ては 00.6885，オンラインタスク割当てが 0.6860，ランダム割当ては 0.6754 であった．他の条件とは異な本条件ではオンラインタスク割当てが必ずしも最良の結果とはならなかった．

以上の結果から，オンラインタスク割当ては多くの条件において最も高い正答率を示した一方で，いずれの手法においても，理想割当てで達成可能な正答率には到達していない．知識タグ付与方法や学習履歴長の違いにより，各手法

表 8: 1450 件・2 分類手法 割当て手法ごとの平均正答率

| 1450 件・分散表現 | | 1450 件・形態類似度 | |
|---|---|---|---|
| 割当て手法 | 平均正答率 | 割当て手法 | 平均正答率 |
| オンラインタスク | 0.702 | オンラインタスク | 0.6965 |
| 静的 DKT | 0.682 | 静的 DKT | 0.6835 |
| 履歴正答率ベース | 0.682 | 履歴正答率ベース | 0.681 |
| ランダム | 0.6754 | ランダム | 0.6754 |

表 9: 1450 件・2 分類手法 割当て手法ごとの平均正答率

| 3000 件・分散表現 | | 3000 件・形態類似度 | |
|---|---|---|---|
| 割当て手法 | 平均正答率 | 割当て手法 | 平均正答率 |
| 動的 DKT | 0.7005 | 動的 DKT | 0.686 |
| 静的 DKT | 0.683 | 静的 DKT | 0.6885 |
| 履歴正答率ベース | 0.6825 | 履歴正答率ベース | 0.689 |
| ランダム | 0.6754 | ランダム | 0.6754 |

の相対的な性能には差が生じることが確認された.

### 7.4.3 作業者ごとの正答率

本節では, 各割当て手法において, 作業者ごとの正答率がどのように分布しているかを分析する. 前節では割当て手法ごとの平均正答率を比較したが, 平均値のみでは, 一部の作業者に正解不正解が偏って割当てられている可能性を十分に評価できない. そこで本節では, 作業者単位での正答率に着目し, 割当て結果の偏りや均一性の観点から, 各手法の特性を比較する.

まず, 学習履歴 1450 件・分散表現ベース分離における作業者ごとの正答率を図 7 および表 10 に示す. 動的割当てでは, 作業者ごとの正答率は 0.48 から 0.94 の範囲に分布しており, 標準偏差は 0.1204 であった. 静的 DKT 割当てでは, 正答率の範囲は 0.34 から 0.95, 標準偏差は 0.1312 であった. 履歴正答率ベース割当てでは, 正答率は 0.51 から 0.92 の範囲に分布しており, 標準偏差は 0.1011 であった.

次に, 学習履歴長 1450 件・形態類似度ベース分類における結果を図 8 および表 10 に示す. オンラインタスク割当てでは, 正答率の範囲は 0.50 から 0.96, 標

準偏差は 0.1222 であった．静的 DKT 割当てでは，正答率の範囲は 0.44 から 0.94，標準偏差は 0.1029 であった．履歴正答率ベース割当てでは，正答率は 0.52 から 0.92 の範囲に分布しており，標準偏差は 0.095 であった．

続いて 3000 件・分散表現ベース分類の結果を図 9 および表 10 に示す．オンラインタスク割当てでは，正答率の範囲は 0.51 から 0.94，標準偏差は 0.1177 であった．静的 DKT 割当てでは，正答率は 0.46 から 0.93，標準偏差は 0.1279 であった．履歴正答率ベース割当てでは，正答率の範囲は 0.36 から 0.93，標準偏差は 0.1282 であった．

最後に，学習履歴長 3000 件・形態類似度ベース分類の結果を図 10 および表 10 に示す．オンラインタスク割当てでは，正答率の範囲は 0.52 から 0.92，標準偏差は 0.1049 であった，静的 DKT 割当てでは，正答率が 0.48 から 0.98 の範囲に分布しており，標準偏差は 0.1289 であった．履歴正答率ベース割当てでは，正答率の範囲は 0.46 から 0.93，標準偏差は 0.1196 であった．

また，オンラインタスク割当てに着目すると，分散表現ベース分類および形態類似度ベース分類のいずれにおいても，作業者ごとの正答率の最小値は 0.48 以上であった．正答率の最大位は条件によって異なり，最大で 0.96(1450 件・形態類似度ベース分類)となった．標準偏差は 0.1049 から 0.1222 の範囲に分布し，オンラインタスク割当てにおける作業者ごとの正答率分布は，条件に応じて異なる広がりを示していることが確認できる．

以上のように，作業者ごとの正答率分布は条件および割当て手法によって異なる数値を示した．

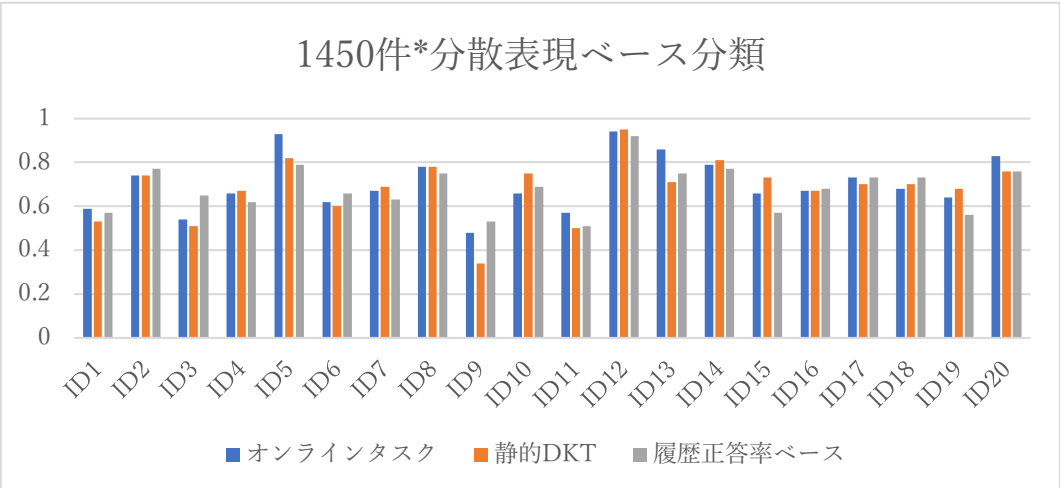


図 7: 1450 件・分散表現ベース分類 作業者ごとの平均正答率

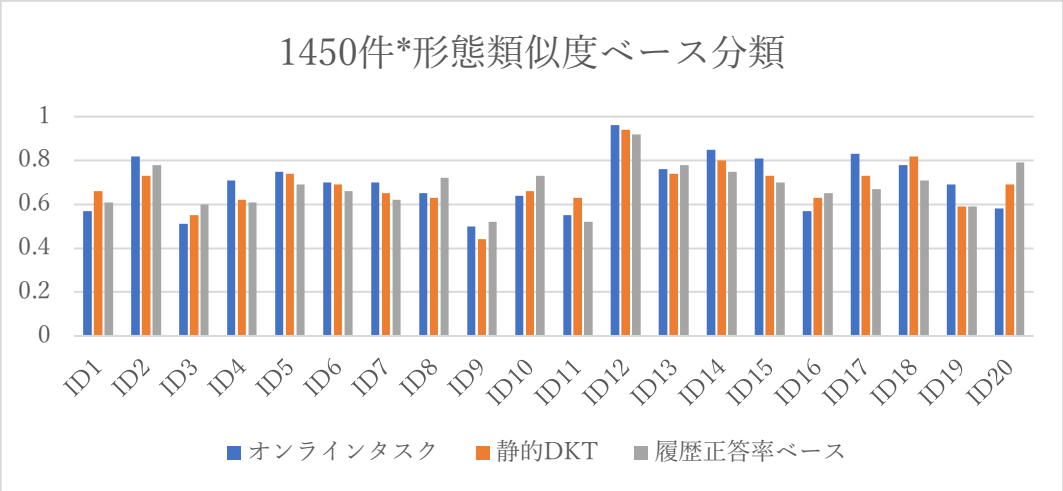


図 8: 1450 件・形態類似度ベース分類 作業者ごとの平均正答率

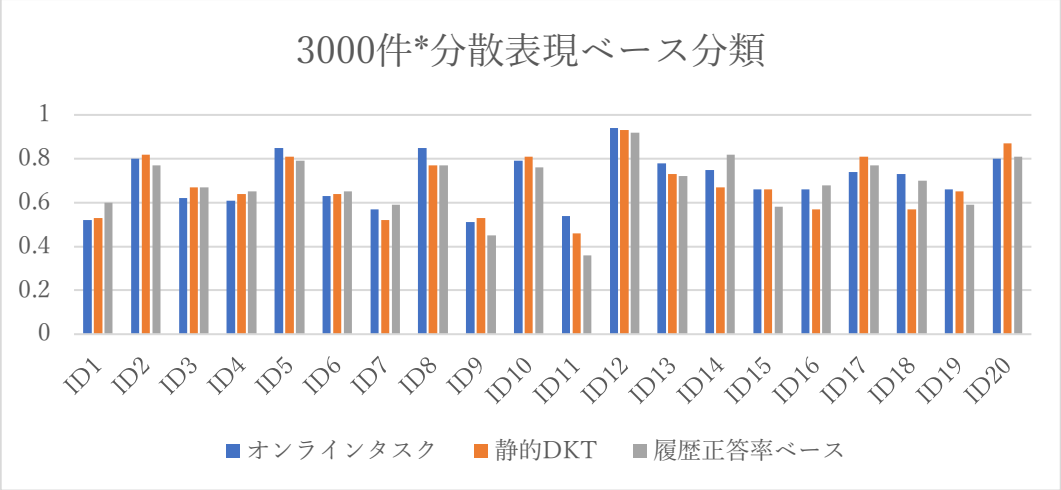


図 9: 3000 件・分散表現ベース分類 作業者ごとの平均正答率

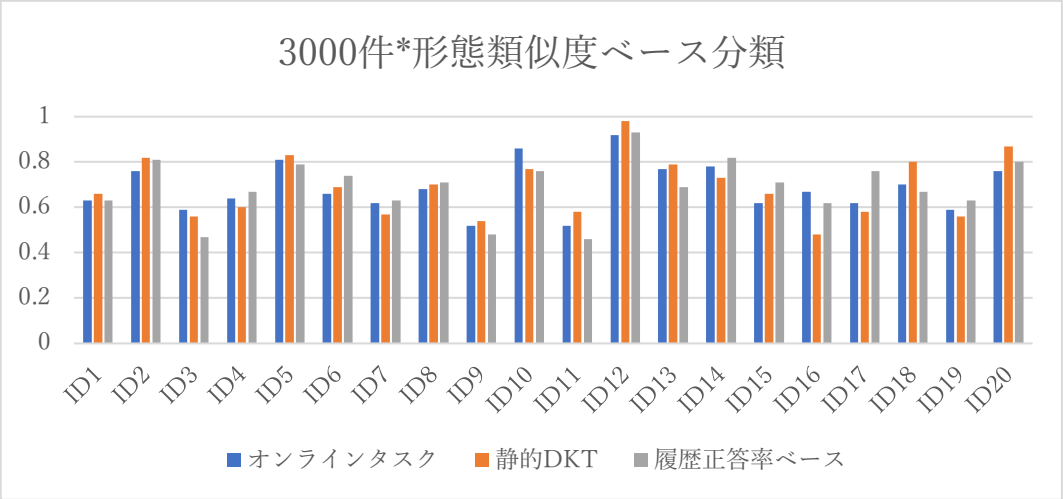


図 10: 3000 件・形態類似度ベース分類 作業者ごとの平均正答率

表 10: 作業者ごとの正答率分布に関する統計量

| 条件 | 割当て手法 | 平均正答率 | 標準偏差 | 最小正答率 | 最大正答率 |
|---|---|---|---|---|---|
| 1450*分散表現 | オンラインタスク | 0.702 | 0.120399 | 0.48 | 0.94 |
| 1450*分散表現 | 静的 DKT | 0.682 | 0.131248 | 0.34 | 0.95 |
| 1450*分散表現 | 履歴正答率 | 0.682 | 0.101124 | 0.51 | 0.92 |
| 1450*形態類似度 | オンラインタスク | 0.6965 | 0.122241 | 0.5 | 0.96 |
| 1450*形態類似度 | 静的 DKT | 0.6835 | 0.102872 | 0.44 | 0.94 |
| 1450*形態類似度 | 履歴正答率 | 0.681 | 0.095021 | 0.52 | 0.92 |
| 3000*分散表現 | オンラインタスク | 0.7005 | 0.117749 | 0.51 | 0.94 |
| 3000*分散表現 | 静的 DKT | 0.683 | 0.12791 | 0.46 | 0.93 |
| 3000*分散表現 | 履歴正答率 | 0.6825 | 0.128175 | 0.36 | 0.92 |
| 3000*形態類似度 | オンラインタスク | 0.686 | 0.104948 | 0.52 | 0.92 |
| 3000*形態類似度 | 静的 DKT | 0.6885 | 0.128852 | 0.48 | 0.98 |
| 3000*形態類似度 | 履歴正答率 | 0.689 | 0.119578 | 0.46 | 0.93 |

### 7.4.4 知識タグごとの正答率

本節では，各割当て手法において，知識タグ(ID0~ID9)ごとの正答率がどのように分布しているかを分析する．前節では作業者単位での正答率分布を示したが，タスク割当ての妥当性を検討する上では，作業者の違いに加えて，タスクが属する知識タグごとの結果の違いを確認することが重要である．そこで本節では，知識タグ単位で正答率を集計し，知識タグごとの分布特性に着目する．

まず，学習履歴長 1450 件・分散表現ベース分類における知識タグごとの正答率を図 11 および表 11 に示す．この条件では，知識タグごとに正答率のばらつきが見られ全ての知識タグで同程度の正答率となるわけではないことが確認できる．具体的には，一部の知識タグでは 0.75 前後の比較的高い正答率が得られて

いる一方で，別の知識タグでは 0.60 前後に止まる場合も見られる．このように，タスクが属する知識タグによって，割当て結果に差が生じていることが分かる．

次に学習履歴長 1450 件・形態類似度ベース分類の結果を図 12 および表 12 に示す．この条件では，分散表現ベース分類の場合とは異なる知識タグにおいて比較的高い正答率が得られる傾向が見られる．同一の知識タグ付与方法の違いによって正答率が変化しており，知識タグの定義方法が結果に影響を与えていることが確認できる．

続いて，学習履歴長 3000 件・分散表現ベース分類における知識タグごとの正答率を図 13 および表 13 に示す．この条件においても，知識タグごとの正答率は一様ではなく，タグによって異なる値を示している．1450 件の場合と比較すると，一部の知識タグでは正答率が上昇している一方で，他の知識タグでは大きな変化が見られない，このことから，学習履歴長の増加が，知識タグ単位の正答率に異なる影響を与えていることが確認できる．

最後に，学習履歴長 3000 件・形態類似度ベース分類の結果を図 14 および表 14 に示す．この条件においても，知識タグごとの正答率は一定ではなく，タグによって正答率の高低が見られる．一部の知識タグでは 0.70 を超える正答率が得られている一方で，別の知識タグでは相対的に低い正答率となる場合が確認できる．以上のように，知識タグごとの正答率は，学習履歴長，知識タグ付与方法，および割当て手法の組み合わせによって，知識タグ単位で異なる分布を示した．本節では，各条件における知識タグごとの結果を事実として示し，特定の知識タグにおいて正答率に差が生じる要因において，次節で詳しく考察する．

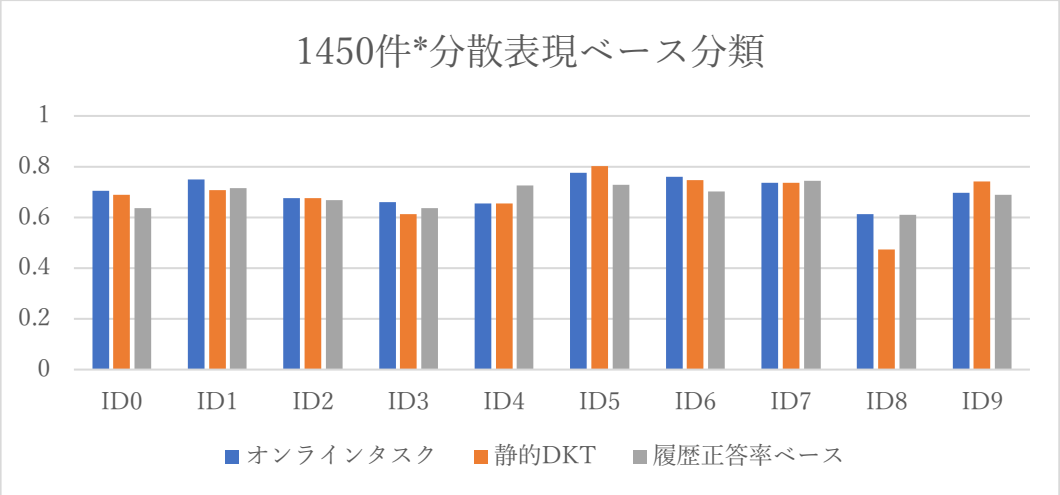


図 11: 1450 件・分散表現ベース分類 知識タグごとの平均正答率

表 11: 1450 件・分散表現ベース分類 知識タグごとの平均正答率

| 割当て手法 | ID0 | ID1 | ID2 | ID3 | ID4 | ID5 | ID6 | ID7 | ID8 | ID9 |
|---|---|---|---|---|---|---|---|---|---|---|
| オンラインタスク | 0.704 | 0.75 | 0.677 | 0.659 | 0.654 | 0.776 | 0.76 | 0.737 | 0.614 | 0.696 |
| 静的 DKT | 0.69 | 0.708 | 0.677 | 0.613 | 0.654 | 0.801 | 0.747 | 0.737 | 0.473 | 0.741 |
| 履歴正答率ベース | 0.637 | 0.714 | 0.668 | 0.636 | 0.727 | 0.729 | 0.701 | 0.744 | 0.609 | 0.688 |

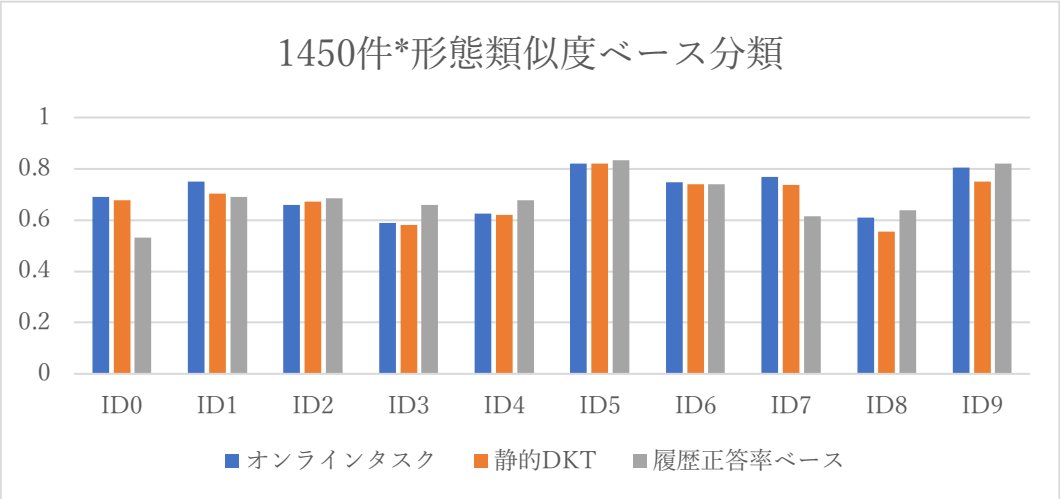


図 12: 1450 件・形態類似度ベース分類 知識タグごとの平均正答率

表 12: 1450 件・形態類似度ベース分類 知識タグごとの平均正答率

| 割当て手法 | ID0 | ID1 | ID2 | ID3 | ID4 | ID5 | ID6 | ID7 | ID8 | ID9 |
|---|---|---|---|---|---|---|---|---|---|---|
| オンラインタスク | 0.69 | 0.75 | 0.659 | 0.59 | 0.624 | 0.819 | 0.747 | 0.769 | 0.609 | 0.804 |
| 静的 DKT | 0.676 | 0.702 | 0.673 | 0.581 | 0.62 | 0.819 | 0.74 | 0.737 | 0.556 | 0.75 |
| 履歴正答率ベース | 0.532 | 0.69 | 0.686 | 0.659 | 0.678 | 0.834 | 0.74 | 0.615 | 0.638 | 0.821 |

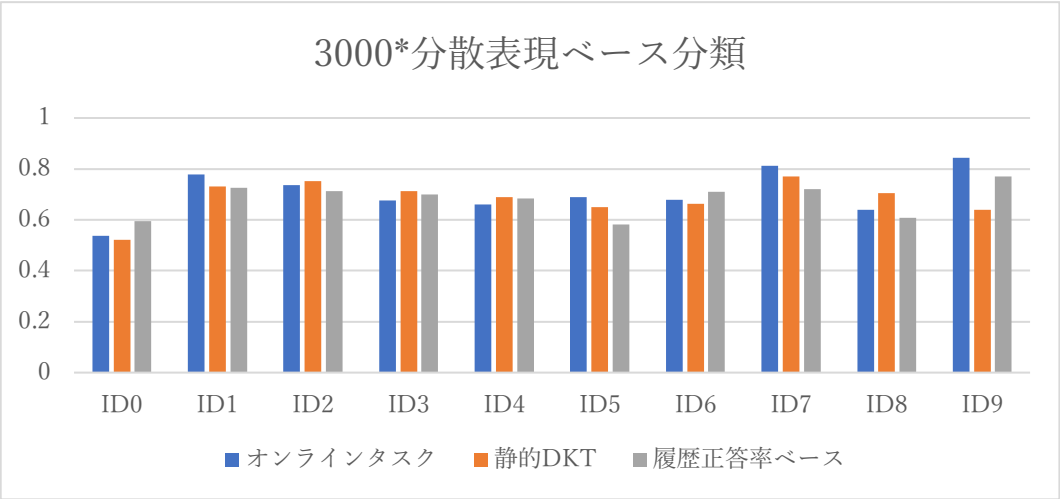


図 13: 3000 件・分散表現ベース分類 知識タグごとの平均正答率

表 13: 3000 件・分散表現ベース分類 知識タグごとの平均正答率

| 割当て手法 | ID0 | ID1 | ID2 | ID3 | ID4 | ID5 | ID6 | ID7 | ID8 | ID9 |
|---|---|---|---|---|---|---|---|---|---|---|
| オンラインタスク | 0.537 | 0.779 | 0.737 | 0.676 | 0.66 | 0.689 | 0.678 | 0.812 | 0.639 | 0.843 |
| 静的 DKT | 0.522 | 0.731 | 0.752 | 0.712 | 0.69 | 0.65 | 0.663 | 0.77 | 0.705 | 0.639 |
| 履歴正答率ベース | 0.596 | 0.727 | 0.714 | 0.699 | 0.685 | 0.583 | 0.709 | 0.721 | 0.607 | 0.771 |

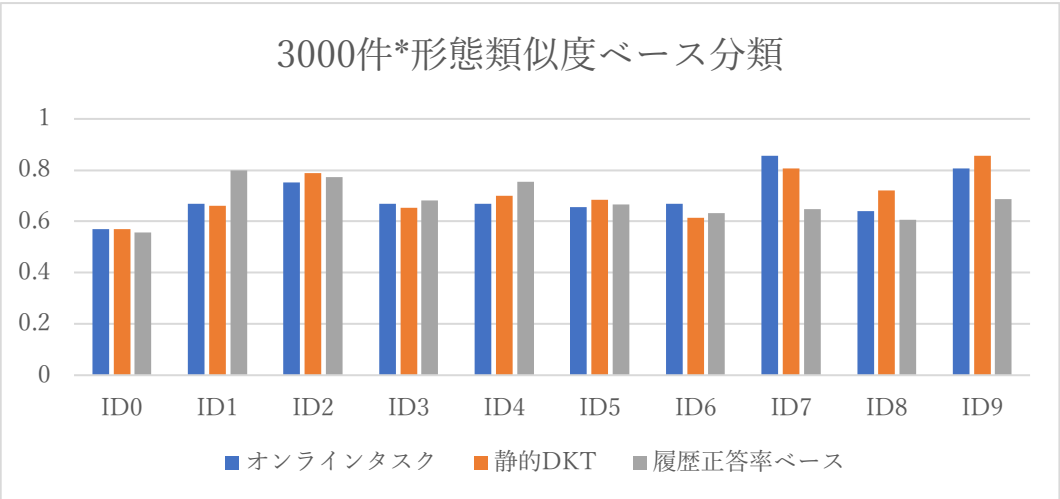


図 14: 3000 件・形態類似度ベース分類 知識タグごとの平均正答率

表 14: 3000 件・形態類似度ベース分類 知識タグごとの平均正答率

| 割当て手法 | ID0 | ID1 | ID2 | ID3 | ID4 | ID5 | ID6 | ID7 | ID8 | ID9 |
|---|---|---|---|---|---|---|---|---|---|---|
| オンラインタスク | 0.569 | 0.668 | 0.752 | 0.67 | 0.67 | 0.655 | 0.668 | 0.855 | 0.639 | 0.807 |
| 静的 DKT | 0.569 | 0.66 | 0.789 | 0.654 | 0.7 | 0.684 | 0.613 | 0.806 | 0.721 | 0.855 |
| 履歴正答率ベース | 0.557 | 0.798 | 0.774 | 0.683 | 0.754 | 0.665 | 0.633 | 0.648 | 0.607 | 0.687 |

### 7.4.5 予測と結果の乖離

本節では，各割当て手法において，タスク割当て時に用いられた正答確率の予測値と，実際の成否結果との乖離について分析する．これまでの節では，正答率を指標として割当て結果を評価してきたが，予測モデルを用いた割当て手法においては，予測値が実際の結果とどの程度一致しているかを確認することも重要である．そこで本節では，予測値と実測結果との差の絶対値を「予測と結果の乖離」として定義し，その分布特性を確認する．

まず，学習履歴長 1450 件・分散表現ベース分類における予測と結果の乖離の分布を図 15 に示す．オンラインタスク割当てでは，乖離の中央値は 0.110 程度であり，四分位範囲は約 0.066 から 0.152 の範囲に分布している．一方で，静的 DKT 割当てでは中央値は 0.093 程度であるものの，最大値は 0.50 と大きな乖離が生じる場合が確認できる．履歴正答率ベース割当てでは，中央値は 0.087 程度であり，分布の上限は 0.166 にとどまっている．

次に，学習履歴長 1450 件・形態類似度ベース分類の結果を図 15 に示す．この条件では，オンラインタスク割当ての乖離の中央値は，0.093 程度であり，四分位範囲は約 0.078 から 0.140 の範囲に分布している．静的 DKT 割当てでは中央値は 0.087 程度であるが，最大値は 0.396 に達しており，一部の作業者において大きな乖離が生じていることが分かる．履歴正答率ベース割当てでは，中央値は 0.085 程度であり，分布のばらつきは比較的抑えられている．

続いて，学習履歴長 3000 件・分散表現ベース分類の結果を図 15 に示す．オンラインタスク割当てでは，乖離の中央値は 0.106 程度であり，四分位範囲は，約 0.073 から 0.131 の範囲に分布している．静的 DKT 割当てでは，中央値は 0.129 程度と比較的大きく，最大値は 0.259 に達している．履歴正答率ベース割当てでは，中央値は 0.111 程度であり，最大値は 0.406 と大きい乖離が確認できた．

最後に，学習履歴長 3000 件・形態類似度ベース分類の結果を図 15 に示す．こ

の条件では，オンラインタスク割当ての乖離の中央値は 0.099 程度であり，四分位範囲は約 0.080 から 0.128 の範囲に分布している．静的 DKT 割当てでは中央値は 0.096 程度であるものの，最大値は 0.294 に達している．履歴正答率ベース割当てでは，中央値は 0.075 程度であり，他の手法と比較して低い値を示していが，最大値は 0.223 となっている．

全条件を通じて，割当て手法ごとに予測と結果の乖離の平均値を算出した結果を表 12 に示す．この表から，割当て手法によって乖離の水準および分布の広がりが異なることが確認できる．特に，静的 DKT 割当ておよび履歴正答率ベース割当てでは，条件によって最大値が大きくなる場合があり，一部の作業者において予測と結果の乖離が大きくなる傾向が見られる．

以上のように予測と結果の乖離は，学習履歴長，知識タグ付与方法，および割当て手法の違いによって，中央値，分布の広がり，および外れ値の出現状況が異なることが確認できた．本節ではこれらの結果を数値に基づいて示し，予測モデルの逐次更新の有無や，割当て手法の設計が乖離に与える影響については，次節の考察において詳しく述べる．

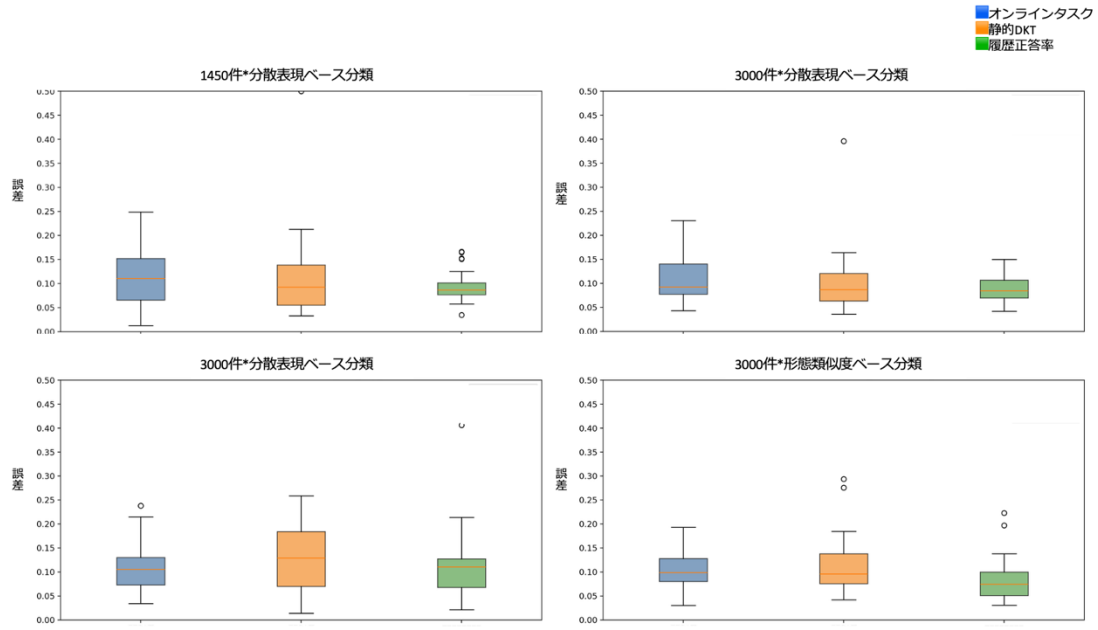


図 15: 4 条件下における 3 種の割当て手法の予測値と結果の乖離

## 7.5 考察

### 7.5.1 オンラインタスク割当てが有効であった理由

第 7.4 節で示した評価結果から，提案手法であるオンラインタスク割当ては，多くの条件において他の割当て手法より高い正答率を示した．特に，学習履歴長 1450 件および 3000 件のいずれにおいても，分散表現ベース分類を用いた条件では，オンラインタスク割当てが一貫して最も高い平均正答率を達成している．この結果は，作業途中の作業結果を逐次的に反映しながら割当てを更新するという本手法の特徴がクラウドソーシング環境において有効に機能したことを示している．

オンラインタスク割当てが有効であった主な要因として，作業者の能力状態を時系列的に捉え，その変化を即時に割当て判断へ反映できた点が挙げられる．静的 DKT 割当てや履歴正答率ベース割当てでは，評価開始時点までの履歴から得られた情報を用いて割当てを一括で決定するため，作業途中で生じる学習効果や疲労による能力変動を考慮できない．オンラインタスク割当てでは，各タスクの実行結果を逐次 DKT に入力し，更新された成否予測に基づいて次の割当てを決定するため，作業者の状態変化に追従した柔軟な割当てが可能となる．また，作業者ごとの正答率分布に着目すると，オンラインタスク割当てでは，多くの条件において作業者間の極端な性能低下が抑えられており，正答率の最小値が比較的高い水準に保たれていた．これは，成否予測の更新により，特定の作業者にとって成功確率の低いタスクが継続的に割当てられる状況が緩和されたためであると考えられる．このことから,オンラインタスク割当ては，全体平均の向上だけでなく，作業者とタスクのミスマッチを軽減する方向に作用したと解釈できる．

さらに，知識タグを介した成否予測の一般化も，オンラインタスク割当ての有効性に寄与している．各タスクは知識タグ単位で DKT に入力されるため，同一知識タグに属する過去の作業結果が，未経験タスクに対する成否予測に反映される．この性質により，特定の知識タグに対して習熟が進んだ作業者に，同系統のタスクが優先的に割当てられやすくなり，結果として正答率の向上につながったと考えられる．

### 7.5.2 比較手法との違い

静的 DKT 割当ては，評価開始前の作業履歴を用いた成否予測に基づき，割当てを固定する手法である．このため，初期予測が安定している条件では一定の

性能を示すものの，作業途中で生じる能力変動を割当てに反映できないという制約を持つ．実験結果においても，多くの条件でオンラインタスク割当てを下回る傾向が確認された．

履歴正答率ベース割当ては，作業者の過去正答率という単純な統計量に基づく手法であり，実装が容易である点が特徴である．しかし，知識タグやタスク内容の違いを考慮できないため，タスクの多様性が高い条件では割当て精度に限界が生じた．

これらと比較して，オンラインタスク割当ては，知識タグ単位で成否予測を逐次更新しながら割当てを行うことで，作業者とタスクの対応関係を柔軟に調整できる点に特徴がある．この違いが，本手法が複数条件で優位な性能を示した主要因である考えられる．

### 7.5.3 学習履歴長と知識タグ付与方法の影響

学習履歴長および知識タグ付与方法は，DKT に入力される履歴構造を変化させ，成否予測および割当て結果に影響を与える重要な要因である．本実験では，学習履歴長として 1450 件および 3000 件の 2 条件を用い，さらに分散表現ベース分類と形態類似度ベース分類の 2 種類の知識タグ付与方法を比較した．分散表現ベース分類を用いた条件では，学習履歴長に関わらずオンラインタスク割当てが安定して高い正答率を示した．これは，意味的に近いタスクが同一タグとしてまとめられることで，作業者の過去の成功・失敗が未経験タスクに対して比較的一般化されているためであると考えられる．この結果から，知識タグがタスクの本質的な類似性を反映している場合，DKT による逐次予測と割当て更新が有効に機能することが示された．形態類似度ベース分類を用いた条件では，特に学習履歴長 3000 件において，オンラインタスク割当てが必ずしも最良の性能を示さない結果が確認された．形態的に類似していても意味的には異なるタスクが同一知識タグに含まれる場合，知識タグ内のばらつきが増大し，成否予測の精度が低下する可能性がある．学習履歴が増加することでこの影響が累積し，逐次更新利点が十分に活かされなかったと考えられる．

以上より，オンラインタスク割当ての性能は，学習履歴長そのものだけでなく，知識タグがどの程度タスクの性質を適切に表現できているかに大きく依存すると考えられ，割当て性能の向上には，予測モデルの高度化だけでなく，タスク分類設計の重要性が不可欠であることを示している．

### 7.5.4 予測と結果の乖離から見た割当ての妥当性

DKT による成否予測と実際の作業結果との差を，予測と結果の乖離として評価した．評価結果から，オンラインタスク割当てでは，多くの条件において乖離の中央値および分布のばらつきが比較的小さく抑えられていた．これは，作業結果を逐次反映することで，成否予測と割当て判断が大きく乖離し続ける状況にならなかったためであると考えられる．しかし，一部の条件では乖離が大きくなるケースも確認されており，予測の不確実性や知識タグ内の多様性が影響した可能性がある．

以上より，オンラインタスク割当ては，予測と結果の乖離の観点においても比較的安定した挙動を示しており，割当て判断が極端に不適切となる状況を抑制できていたと解釈できる．ただし，乖離が完全に解消されるわけではなく，予測誤差やタスク分類の限界が残る点については，次節で理想割当てとの比較を用いて考える．

### 7.5.5 理想割当てとの関係と本研究の限界

本研究では，各タスクを最も正答可能性の高い作業者に割当てた場合の結果を理想割当てとして設定し，提案手法との比較を行った．理想割当ては，実際の成否結果を事前に知っているという後知恵に基づく上限性能であり，現実的な理想割当て手法が到達し得る最大値を示す指標である．評価結果より,オンラインタスク割当てを含む全ての割当て手法は，いずれの条件においても理想割当てよりも約 30%低い正答率にとどまった．この差は小さくない値であり，成否予測に基づく割当てが理論上の最適解と比較して依然として大きな改善余地を有することを示している．

この差は単一の要因によるものではないと考えられる．予測モデル性能この差の主な要因として，成否予測の誤差に加え，知識タグ内に含まれるタスクの多様性や，作業者の集中度や疲労といった履歴からは観測できない要因が挙げられる．特に，成否予測は知識タグ単位で一般化されるため，同一タグ内であっても難易度や意味の異なるタスクが混在する場合，理想的な割当て判断を完全に再現することは困難である．

以上より，本研究で得られた理想割当てとの差は，提案手法の不備というよりも，利用可能な情報と予測精度に基づく構造的な制約を反映した結果であると解釈できる．すなわち，現実のクラウドソーシング環境において完全な最適割当てを実現するためには，予測精度の向上だけでなく，タスク設計およびモ

デル設計の改善が不可欠である．本研究の限界としては，タスク分類設計への依存，2 値正誤による評価の単純化，および実環境との差異などが挙げられる．今後は，知識タグ設計の改良を中心として改良することで，理想割当てとの差の縮小を図ることが課題である．

# 第8章 おわりに

本研究では，クラウドソーシングにおけるタスク割当ての品質向上を目的として，Deep Knowledge Tracing を用いた動的タスク割当て手法を提案した．特に，作業者の過去の作業履歴を時系列的にモデル化し，タスク実行結果を逐次的に成否予測へ反映することで，作業途中における能力変動を考慮したタスク割当てを実現した点に本研究の特徴がある．

提案手法の有効性を検証するため，低資源言語間の対訳辞書作成タスクを対象として実データを用いた評価を行った．その結果，オンラインタスク割当ては，静的 DKT 割当て，履歴正答率ベース割当て，ランダム割当てと比較して，多くの条件において高い正答率を示した．特に，意味的類似性に基づく知識タグ付与を行った条件では，学習履歴長に依らず安定した性能向上が確認され，逐次的な予測更新とタスク割当ての組み合わせがクラウドソーシング環境において有効であることが示された．

しかし，理想割当てとの比較から，提案手法を含む全ての現実的な割当て手法は，いずれの条件においても理想割手より約 30%低い正答率にとどまることが確認された．この差は，成否予測誤差や知識タグ設計の限界，並びに作業者の集中度などの状態といった履歴からは観測できない要因に起因する可能性があり，予測の基づくタスク割当てが本質的に持つ制約を反映した結果であると解釈できる．

以上より，本研究は，DKT に基づく逐次的な成否予測をタスク割当てに統合する自治で，作業者とタスクのミスマッチを緩和し，クラウドソーシングにおける成果物の品質向上に寄与できることを示した．今後の課題としては，知識タグ設計の改良や階層化，不確実性を考慮した割当て手法の導入，および実運用環境におけるオンライン評価を通じて，理想割当てとの差をさらに縮小することが挙げられる．

# 謝辞

本研究を行うにあたり，熱心なご指導，ご助言を賜りました村上陽平教授に深く感謝申し上げます．また，普段からお世話になっている社会知能研究室の皆様にも心より感謝申し上げます．

# 参考文献


[1] Sakti, S. and Nakamura, S.: Towards Language Preservation: Design and Collection of Graphemically Balanced and Parallel Speech Corpora of Indonesian Ethnic Languages, Proc. of the International Conference on Oriental COCOSDA and Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE), pp. 1–5 (2013).

[2] Chida, H., Murakami, Y. and Pituxcoosuvarn, M.: Quality Control for Crowdsourced Bilingual Dictionary in Low-Resource Languages, Proc. of the 13th International Conference on Language Resources and Evaluation (LREC 2022), pp. 6590–6596 (2022).

[3] Karger, D. R., Oh, S. and Shah, D.: Budget-Optimal Crowdsourcing Using Low-Rank Matrix Approximations, Proc. of an International Conference, pp. 284–291 (2011).

[4] Karger, D. R., Shah, D. and Oh, S.: Budget-Optimal Task Allocation for Reliable Crowdsourcing Systems, Operations Research, Vol. 62, No. 1, pp. 1–24 (2014).

[5] 小林正樹： 人間+AIの相互作用によるクラウドソーシングの品質管理に関する研究, 筑波大学修士論文 (2022).

[6] 鹿島久嗣, 梶野洸: クラウドソーシングと機械学習 (〈特集〉 知識の転移), 人工知能, Vol. 27, No. 4, pp. 381–388 (2012).

[7] 金地紗里奈, 小板隆浩: クラウドソーシングにおける品質に対するワーカーの影響, 研究報告情報システムと社会環境 (IS), 2019-IS-148, pp. 1–6 (2019).

[8] Punitha, R. et al.: The Complexity of Task Allocation in Crowdsourcing, Journal of Crowd Science (2020).

[9] Machado, T. et al.: Dynamic Task Allocation in Crowdsourcing Environments, International Journal of Crowd Science (2016).

[10]Yu, Q. et al.: A Collaborative Development Approach for Optimal Task Assignment Using Hungarian Method, IEEE Transactions on Knowledge and Data Engineering (2019).

[11] Patel, P. et al.: Scheduling of Jobs Based on Hungarian Method in Cloud Computing, Proc. of the International Conference on Computing, Communication and Technologies (ICCCT) (2017).

[12] Miao, C. et al.: Quality-Aware Online Task Assignment in Mobile Crowdsourcing, IEEE Transactions on Mobile Computing (2020).

[13] Wang, Y. and Xie, L.: Enhancing System Efficiency Using MCTR Model, Proc. of the International Conference on Mobile Systems, Applications, and Services (MobiSys) (2020).

[14] Hettiachchi, D. et al.: Feasibility of Crowdsourcing Across Multiple Devices, Proc. of the SIGCHI Conference on Human Factors in Computing Systems (CHI) (2022).

[15] Corbett, A. T. and Anderson, J. R.: Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge, User Modeling and User-Adapted Interaction, Vol. 4, No. 4, pp. 253–278 (1995).

[16] Baker, R. S. J. d., Corbett, A. T. and Aleven, V.: More Accurate Student Modeling Through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing, Intelligent Tutoring Systems, pp. 406–415 (2008).

[17] Piech, C., Bassen, J., Huang, J. et al.: Deep Knowledge Tracing, Advances in Neural Information Processing Systems (NIPS), pp. 505–513 (2015).

[18] Wilson, K. H. and Xiong, X.: On Deep Knowledge Tracing, Proc. of the International Conference on Educational Data Mining, pp. 393–398 (2016).

[19] Yeung, C.-K. and Yeung, D.-Y.: Addressing Two Problems in Deep Knowledge Tracing via Prediction-Consistent Regularization, Proc. of the International Conference on Learning Representations (ICLR) (2018).