

# 曖昧なユーザ要求に対する因果関係知識を用いた 気の利いたシステム行動の選択

田中翔平<sup>1,2,3</sup> 吉野幸一郎<sup>2,1,3</sup> 須藤克仁<sup>1,3</sup> 中村哲<sup>1,3</sup>

<sup>1</sup> 奈良先端科学技術大学院大学 <sup>2</sup> 理化学研究所 ガーディアンロボットプロジェクト

<sup>3</sup> 理化学研究所 革新知能統合研究センター

{tanaka.shohei.tj7, sudoh, s-nakamura}@is.naist.jp

koichiro.yoshino@riken.jp

## 概要

タスク対話システムは、ユーザの要求に応じて適切なタスク行動を行う。この際、ユーザ発話が明確な場合のみならず、ユーザ発話が曖昧な場合でも適切なタスク行動を行うことが求められる。本研究で提案するシステムは、こうした曖昧なユーザ発話に対して、そこから推測できる潜在的な要求に対応したタスク行動を選択することを目標とする。曖昧なユーザ発話に対して気の利いたシステム行動は複数存在しうるため、これらが一対一で対応付けられたデータから、Positive/Unlabeled (PU) 学習による分類器を学習する。また、学習データセットから因果関係検出モデルによって因果関係に相当する知識を抽出・蒸留することで、より精度の高いシステム行動の識別を実現する。実験の結果、提案する因果関係検出モデルを用いる手法は、単純にPN学習やPU学習を用いた場合よりも高精度で気の利いたシステム行動を識別できることが示された。

## 1 はじめに

音声対話研究において、従来の音声言語理解システムの多くは、ユーザがシステムに対する要求を明示的に示すことを仮定している [1]。しかし実際には、ユーザの要求はしばしば曖昧であり、ユーザ自身が明示的な要求を自覚していない場合も多い [2]。こうした場合、人間のコンシェルジュやガイドであれば気を利かせてユーザが必要とする補助を提供できる。例えば、ユーザが「この景色最高だね！」と発話したとき、「写真を撮りましょうか？」などと応答することができる。

こうした行動選択を実現する方法のひとつとして、特定のシステム行動に対してそれが気が利いて

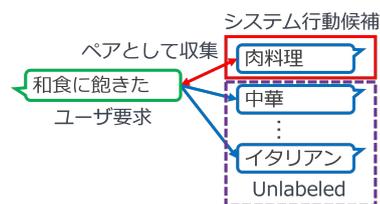


図1 ユーザ要求とシステム行動のペア

いると見なすことができる曖昧なユーザ発話を収集し、これを学習データとして識別に用いる方法が挙げられる [3]。この方法論では、ある曖昧なユーザ要求に対して、気の利いた行動である可能性が高い行動候補の一つを収集することができる。例えば図1のような「和食に飽きた」というユーザ要求に対しては、「肉料理店の提案」や「中華料理店の提案」など複数のシステム行動候補のうち一つだけが収集される。つまり、意図が曖昧なユーザ発話に対して、少数の気の利いた行動候補と、気が利いていないかわからない多数の行動候補が存在する、Positive/Unlabeled (PU) 分類問題 [4, 5, 6] が定義される。

本研究ではこうした曖昧なユーザ発話からシステムが取りうる気の利いた行動を識別する問題において、事例から汎化された因果関係のような知識を用いることを考える。具体的には、学習データから因果関係に相当する知識を蒸留し、システム行動の選択においてこの知識を用いる。このため、テキストデータからの因果関係検出モデルを導入する。この因果関係検出モデルにおいては、曖昧なユーザ発話を原因、気の利いたシステム行動を結果とした学習を行い、因果関係に相当する特徴量抽出を行った。この抽出特徴量を、PU学習における分類器のラベル伝播に利用し、気の利いた行動の識別に利用したところ、識別の精度向上が確認された。

## 2 データセット

既存の多くのタスク対話システムは、ユーザの意図はユーザの中で明確化されており、曖昧性のないユーザ発話が与えられることを仮定している。しかし実際には、システムと対話する時に何を要求したいか定まっていない場合も多く、そのようなケースでのユーザ発話は意図が曖昧なものになりやすい。ARTA コーパス [3] は観光案内のドメインにおいてユーザとスマートフォンアプリケーション上の対話エージェントとの対話を想定し、システムが取りうる行動 (応答) に対してそれが気が利いていると見なすことができる曖昧なユーザ発話を付与したものである。図 2 にこのコーパス中の対話例を示す。ここでユーザの「この景色最高だね!」という発話は、必ずしも特定の機能に対する要求として言語化されているわけではない。これに対して、対話エージェントが「カメラアプリを起動しましょうか?」という気の利いた応答を返し、実際のカメラ機能を起動できるようにする。

このコーパス中では対話エージェントの機能 (システム行動; 応答) はスポット検索, レストラン検索, アプリ起動の 3 つに大きく分けて定義されている。定義された機能のリストを表 1 に示す。各機能はそれぞれ細分化されたカテゴリを持ち、コーパス中の対話エージェントの応答はこれらのカテゴリに紐付いて生成される。定義されたカテゴリは全部で 70 種類である。本コーパスでは、これらのあらかじめ定義されたエージェントの応答カテゴリに対して、広範なユーザの先行発話をワーカーに輸入してもらう方法により発話-応答ペアが収集されている。収集されたコーパス中の発話例を表 2 に示す。コーパス中に含まれるユーザ発話と応答カテゴリのペアは合計 27,230 であり、学習データ:検証データ:テストデータ = 24,430 : 1,400 : 1,400 の割合で分割されている。

ARTA コーパスに含まれるユーザ発話は要求が曖昧であるため、気が利いていると認定される応答があらかじめ定義された 1 種類とは限らない。しかし全てのユーザ発話とシステム行動の組み合わせに対して全応答カテゴリに対応するかどうかラベリングを行うことは多大なコストが掛かり、現実的ではない。そこでこのコーパスは、テストデータに含まれる 1,400 発話に対してのみ、可能なシステム行動が網羅的に付与されている (平均 9.55 種類)。例えば

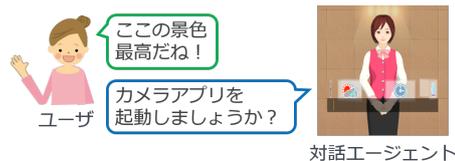


図 2 気の利いた対話例

表 1 対話エージェントの機能とカテゴリ: # はカテゴリ数を表す。

機能	カテゴリ	#
スポット検索	遊園地, 美術館, など	30
レストラン検索	カフェ, かき氷, など	30
アプリ起動	カメラ, 乗り換え, など	10

表 2 の「最近和食が多くて少し飽きてきたんですよ」というユーザ発話例では、肉料理や中華, イタリアンといった複数の応答カテゴリが気が効いている行動と認定されている。つまり、正解となるテストデータには正例・負例となるシステム行動が明示的にアノテーションされているが、学習データには正例の一部のみがアノテーションされており、残りの部分のアノテーションがない Positive/Unlabeled (PU) の問題になっている。

## 3 分類モデル

本論文では、気の利いたシステム行動を識別するタスクにおいて因果関係に相当する知識を導入する。このため、まず問題に対応した PU 学習に基づく分類モデルについて説明する。このモデルに対して、因果関係検出モデルを用いた因果関係スコアを導入し、学習データの正例中に存在する因果関係相当の知識を用いて分類精度の向上を目指す。

### 3.1 PU 学習を用いた分類モデル

図 3 にユーザ発話の分類タスク及びモデルの概要を示す。分類モデルは入力されたユーザ発話を、そのユーザ発話に紐付けられた対話エージェントの正解応答 (正例) カテゴリへと分類する。ここで、曖昧なユーザ発話に対して正解となりうるシステム応答の候補は一つではない場合が多いため、PU 学習を用いる。具体的には Cevikalp ら [6] が提案したラベル伝播 [7, 8] に基づいた PU 学習を用いる。この手法はラベル付けされていない事例に対して、分散表現上で近傍のラベル付けされた事例のラベルを伝播するものである。具体的には、事例同士の類似度スコア  $s_{ij}$  を次式のように計算する。

$$s_{ij} = \exp\left(-\frac{d(\mathbf{x}_i, \bar{\mathbf{x}}_j)}{\bar{d}} \cdot \frac{|C|}{|C|-1}\right). \quad (1)$$

表 2 ARTA コーパス [3] 中に含まれるユーザ要求の例

ユーザ発話 (クラウドソーシングを用いて収集)	システム応答 (あらかじめ定義)
汗をかいて気持ち悪い	周辺の温泉を検索しましょうか？
最近和食が多くて少し飽きてきたんですね。	周辺の肉料理を検索しましょうか？
いい景色だな	カメラを起動しましょうか？

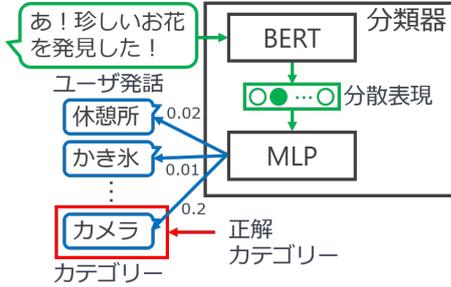


図 3 ユーザ要求分類器

$\bar{\mathbf{x}}_j$  はシステム行動  $j$  が正例とされるユーザ発話の分散表現の平均ベクトルである。  $d(\mathbf{x}_i, \bar{\mathbf{x}}_j)$  は  $\mathbf{x}_i$  と  $\bar{\mathbf{x}}_j$  のユークリッド距離であり、  $\bar{d}$  は全距離の平均である。  $C$  は定義されたシステム行動の集合であり、  $|C| = 70$  である。  $s_{ij}$  の値域は  $0 \leq s_{ij} \leq 1$  であり、2つの分散表現間のユークリッド距離が短いほど大きな値をとる。本研究ではこの手法を **PU** と呼ぶ。類似度スコア  $s_{ij}$  は次式を用いて  $-1 \leq s_{ij} \leq 1$  の範囲にスケールされる。

$$\tilde{s}_{ij} = -1 + \frac{2(s - \min(s_{ij}))}{\max(s_{ij}) - \min(s_{ij})}. \quad (2)$$

スケールされた類似度スコア  $\tilde{s}_{ij}$  が  $0 \leq \tilde{s}_{ij} \leq 1$  であるとき、システム行動  $j$  を対象のユーザ発話の正例として扱い、このサンプルの正例としての重みを  $\tilde{s}_{ij}$  とする。  $\tilde{s}_{ij}$  が  $-1 \leq \tilde{s}_{ij} < 0$  のとき、システム行動  $j$  を対象のユーザ発話の負例として扱い、重みを  $-\tilde{s}_{ij}$  とする。

類似度スコア  $\tilde{s}_{ij}$  を用いたラベル伝播に基づく損失関数は次式の通りである<sup>1)</sup>。

$$\begin{aligned} Loss = & \sum_i^{|U_{train}|} \sum_{j=1}^{|C_+^+|} \sum_{k=1}^{|C_-^-|} \tilde{s}_{ij} \tilde{s}_{ik} L(r_j) R_s(\mathbf{w}_j^T \mathbf{x}_i - \mathbf{w}_k^T \mathbf{x}_i) \\ & + \kappa \sum_i^{|U_{train}|} \sum_{j=1}^{|C|} \tilde{s}_{ij} R_s(y_{ij}(\mathbf{w}_j^T \mathbf{x}_i)). \end{aligned} \quad (3)$$

式 (3) において、  $\tilde{s}_{ij}$  は伝播されたシステム行動の、損失関数への寄与率を表す重みとして働く。正例として元々付与されているシステム行動の類似度スコア  $\tilde{s}_{ij}$  は 1 である。

1) 式の詳細は付録を参照。

## 3.2 因果関係知識を用いた類似度スコアの調整

3.1 節におけるラベル伝播法は、ユーザ発話同士の類似に基づいてラベルを伝播している。しかし、ユーザがある要求を行った場合に一般的などのような行動を取ることが適切か、学習データにおいて観測される因果関係を考慮することで、より適切なラベル伝播を行うことができる可能性がある。そこで本研究では、こうした因果関係を汎化された知識として抽出・蒸留しシステム行動の識別に利用するために、ユーザ発話を原因、システム行動を結果とみなして構築した因果関係検出モデルを用いる。例えば「和食に飽きる」→「肉料理を食べる」というペアがあった場合、前者を原因、後者を結果と見なしてモデルを構築する。この構築されたモデルから出力された因果関係スコアを、ラベル伝播における新しい特徴として利用する。

図 4 に因果関係スコアを算出する因果関係検出モデルの概要を示す。このモデルは [SEP] トークンで連結された原因、結果を BERT によって実数ベクトルに変換し、MLP によって実数値の因果関係スコアを算出する。因果関係スコアの値が大きいほど、入力された原因、結果は因果関係として強く結びつくことを意味する。

本研究ではユーザ発話、システム行動に含まれるイベント間の因果関係認定を行いたい。より単純なイベント同士の関係に着目することで、今回のタスクにおける因果関係を蒸留した因果関係スコアを算出できることが期待される。そこで本研究では因果関係グラフ [9] を用いてユーザ要求を解析し、ユーザ要求に含まれる述語項構造をイベントの基本単位として BERT に入力する。因果関係グラフは「なぜなら」などの言語学的な特徴量に基づいて文中の因果関係を検出するツールであり、前処理として文中のイベントを述語項構造を基本単位として抽出する。また結果として入力するシステム行動に含まれるイベントについては、システム行動が「肉料理の検索」であれば「肉料理を食べる」など、70 種類の各システム行動に対応した述語項構造として定義する。

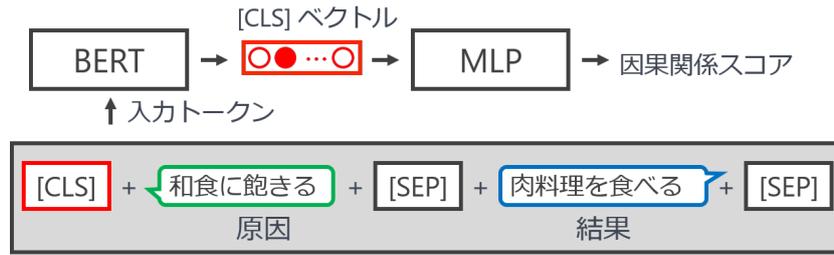


図 4 因果関係検出モデル

表 3 ユーザ要求の分類性能

Model	Acc. (%)	R@5 (%)	MRR
PN	89.04	98.10	0.9304
PU	89.28	97.87	0.9305
PU-causal	90.05	98.27	0.9366

因果関係検出モデルの学習には、ユーザ発話と正例として紐づけられたシステム行動に含まれるイベントのペアを正例  $cx_i^+$ 、正例として紐づけられていないペアを負例  $cx_i^-$  とした次式の margin ranking loss を用いる。

$$Loss(cx_i^+, cx_i^-) = \max(0, -(c_i^+ - c_i^-) + 0.5). \quad (4)$$

$c_i^+, c_i^-$  はそれぞれ正例、負例に対してモデルが出力した因果関係スコアであり、式 (4) は  $c_i^+ - c_i^- \geq 0.5$  のとき 0 となる。

学習済みの因果関係検出モデルを用いて、式 (3) の類似度スコア  $s_{ij}$  を更新する。具体的には、まずユーザ発話  $x_i$  に対して紐づけられていないシステム行動  $j$  のイベントの全てのペアについて因果関係スコア  $c_{ij}$  を計算する。ユーザ要求中に複数のイベントが含まれている場合、最も因果関係スコアが高くなるイベントをユーザ要求  $x_i$  としてスコアを計算する。次に、全体の因果関係スコアについて式 (2) を用いて  $-1 \leq c_{ij} \leq 1$  の範囲にスケールリングした上で、式 (5) の通り PU 学習における類似度スコアの更新に用いる。

$$\hat{s}_{ij} = \max(\min(\tilde{s}_{ij} + c_{ij} - \gamma, 1), -1). \quad (5)$$

$\gamma$  は因果関係スコアの影響を調整するためのハイパーパラメータである。アップデートされた類似度スコア  $\hat{s}_{ij}$  を式 (3) にて用い、分類モデルを学習する。本研究ではこの手法を **PU-causal** と呼ぶ。

## 4 実験

ARTA コーパスで定義されている通り、曖昧なユーザ発話を入力として対応しうる気の利いたシステム行動のカテゴリを出力するモデルを評価した結

表 4 ラベル伝播の性能

Model	Pre. (%)	Rec. (%)	F1
PU	62.19	13.51	0.2200
PU-causal	98.03	0.61	0.0121

果を表 3 に示す。比較手法としては、論文で述べた PU 学習、PU 学習のラベル伝播時に因果関係知識を用いる提案法の他に、単純に Unlabeled なデータを負例として扱う PN 学習による分類を用いた。評価指標として、Accuracy (Acc.), Recall@5 (R@5), Mean Reciprocal Rank (MRR) を用いた<sup>2)</sup>。このとき、PN 学習や単純にユーザ発話とシステム行動の関係を PU 学習で識別する手法 (PU) よりも、因果関係知識を用いた PU 学習手法 (PU-causal) がすべての指標について有意な性能差 (t 検定; 有意水準 0.01) を示した。

実際にラベル伝播法が PU 学習にどの程度効果的に働くかを検証するため、テストデータに対してラベル伝播を実施した場合の性能を表 4 に示す。表 3 と表 4 を比較すると、ラベル伝播時の誤伝播がモデルへの影響が大きく、より伝播の適合率が重視されるタスクであることがわかる。また提案する因果関係知識を用いた手法の方がユーザ要求間の類似度スコアのみを用いた場合よりも精度が高いことがわかる。

## 5 おわりに

本研究ではユーザの曖昧な要求と対話エージェントの気の利いた応答を対応付ける分類モデルの学習に因果関係知識を用いた。具体的には PU 学習におけるラベル伝播に用いる特徴量として、ユーザ要求とカテゴリ応答が因果関係として結びつくかを判定する因果関係スコアを提案した。ユーザ要求の分類実験の結果、提案する特徴が PU 学習におけるラベル伝播の適合率を高め、その結果分類器の精度を向上できることがわかった。

2) 実験設定及び評価指標の詳細は付録を参照。

## 謝辞

本研究に利用した因果関係グラフを提供して頂いた京都大学黒橋研究室の黒橋禎夫教授，植田暢大さん，清丸寛一さんに感謝いたします。

## 参考文献

- [1] Steve Young, Milica Gašić, Simon Keizer, François Mairesse, Jost Schatzmann, Blaise Thomson, and Kai Yu. The hidden information state model: A practical framework for pomdp-based spoken dialogue management. *Computer Speech & Language*, Vol. 24, No. 2, pp. 150–174, 2010.
- [2] Koichiro Yoshino, Yu Suzuki, and Satoshi Nakamura. Information navigation system with discovering user interests. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pp. 356–359, Saarbrücken, Germany, August 2017. Association for Computational Linguistics.
- [3] Shohei Tanaka, Koichiro Yoshino, Katsuhito Sudoh, and Satoshi Nakamura. ARTA: Collection and classification of ambiguous requests and thoughtful actions. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pp. 77–88, Singapore and Online, July 2021. Association for Computational Linguistics.
- [4] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pp. 740–755, Cham, 2014. Springer International Publishing.
- [5] Charles Elkan and Keith Noto. Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '08, p. 213–220, New York, NY, USA, 2008. Association for Computing Machinery.
- [6] Hakan Cevikalp, Burak Benligiray, and Omer Nezih Gerek. Semi-supervised robust deep neural networks for multi-label image classification. *Pattern Recognition*, Vol. 100, p. 107164, 2020.
- [7] Denny Zhou, Jiayuan Huang, and Bernhard Schölkopf. Learning from labeled and unlabeled data on a directed graph. In *ICML '05 Proceedings of the 22nd international conference on Machine learning*, p. 1036. ACM Press, August 2005.
- [8] Hakan Cevikalp, Jakob Verbeek, Frédéric Jurie, and Alexander Klaser. Semi-supervised dimensionality reduction using pairwise equivalence constraints. 3rd International Conference on Computer Vision Theory and Applications (VISAPP '08), pp. 489–496, 2008.
- [9] 清丸寛一，植田暢大，児玉貴志，田中佑，岸本裕大，田中リベカ，河原大輔，黒橋禎夫. 因果関係グラフ: 構造的言語処理に基づくイベントの原因・結果・解決策の集約. 言語処理学会 第 26 回年次大会 発表論文集 (ANLP), pp. 1125–1128, 2020.
- [10] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019.
- [11] Tomohide Shibata, Daisuke Kawahara, and Kurohashi Sadao. Kurohashi Lab. BERT ([http://nlp.ist.i.kyoto-u.ac.jp/?ku\\_bert\\_japanese](http://nlp.ist.i.kyoto-u.ac.jp/?ku_bert_japanese)). 2019.
- [12] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.

## 付録

### 式 (3) の詳細

$U_{train}$  は学習データに含まれるユーザ発話の集合であり,  $C_{x_i}^+, C_{x_i}^-$  はそれぞれユーザ発話  $x_i$  に対応付けられた正例応答カテゴリの集合, 対応付けのない応答カテゴリの集合である.  $r_j$  は正例カテゴリ  $j$  に対してモデルが予測した順位であり,  $L(r_j)$  は次式を満たす重み関数である.

$$L(r_j) = \sum_{k=1}^{r_j} \frac{1}{k}. \quad (6)$$

式 (6) はモデルが予測した順位が一位から遠いほど大きな値をとる.  $\mathbf{w}_j$  はカテゴリ  $j$  に対応する重みベクトル,  $\mathbf{x}_i$  はユーザ発話  $x_i$  に対応する分散表現である.  $R_s(t)$  は次式で表せられる Ramp loss である.

$$R_s(t) = \min(1 - m, \max(0, 1 - t)). \quad (7)$$

$m$  は分類境界を決定するハイパーパラメータである.  $y_{ij}$  はユーザ発話  $x_i$  に対してカテゴリ  $j$  が正例である場合 1, 負例である場合  $-1$  となる.  $\kappa$  は第二項の重みを表すハイパーパラメータである.

### 実験設定

モデルの実装には PyTorch [10] を用いた. BERT モデルとして, Wikipedia 記事を対象に事前学習を行った BERT モデル [11] を用いた.

パラメータの最適化には Adam [12] を使用し, 学習率は  $1e-5$  とした. 式 (7) の  $m$ , 式 (3) の  $\kappa$  について, Cevikalp ら [6] の実装と同様に  $m = -0.8, \kappa = 5$  とした. 本研究では, BERT が出力した分散表現をラベル伝播に必要な分散表現  $\mathbf{x}_i$  として用いる. 因果関係検出モデルの学習データ, 検証データ, テストデータには ARTA コーパス [3] の学習データを分割したものを用いた. また因果関係グラフによって抽出されたユーザ要求中に含まれる因果関係も学習に用いた. 学習した因果関係検出モデルの精度は 91.70 であった. 式 (5) の  $\gamma$  は 1.0 とした. 実験に用いる各モデルのパラメータは, 検証データにおける損失関数の値が最小のものを使用した.

### 評価指標

R@5 は分類モデルが出力した正解カテゴリの順位が上位 5 位以内に含まれている割合である. MRR ( $0 < MRR \leq 1$ ) は次式の通り算出される.

$$MRR = \frac{1}{|U_{test}|} \sum_i^{|U_{test}|} \frac{1}{r_{x_i}}. \quad (8)$$

ここで  $r_{x_i}$  はユーザ発話  $x_i$  に対応する正解カテゴリについて分類モデルが出力した順位を意味し,  $U_{test}$  はテストデータに含まれるユーザ発話の集合である. 全ての指標について, 数値が大きいほど分類モデルの性能が高いことを意味する. 各モデルの性能はそれぞれ 100 回の試行の平均より算出した. なおテストデータに対しては正解となる応答カテゴリが網羅的に付与されており, それぞれマルチラベルのスコアについて計算を行っている.