Commonsense-aware Attention と Discrepancy Resolution Loss を用いたユーモア検出手法の提案

佐々木裕多† 張 建偉† 白石 優旗††

† 岩手大学理工学部 〒020–8551 岩手県盛岡市上田 4-3-5
†† 筑波技術大学産業技術学部 〒305–8520 茨城県つくば市天久保 4-3-15
E-mail: †{s0619027,zhang}@iwate-u.ac.jp, ††yuhkis@a.tsukuba-tech.ac.jp

あらまし 笑いは健康に良い影響をもたらしている．笑いの要因となるユーモアを対話システムへ実装することで，ユーザの心身の満足度が向上すると期待できる．心理学研究における不一致理論において，期待と実際に感受したことの「ずれ」を認知することで笑いが起こるとされている．正しく期待するためには適切な知識が必要である．本研究では，Commonsense-aware Attention を用いてコモンセンスを考慮するモジュールを PLM（**P**re-trained **L**anguage **M**odel）に外挿することで，知識を増強する抽象的なアーキテクチャを提案する．また，不一致理論に着目し，非ユーモアデータに対してコモンセンスから得られる期待と入力文のコンテクストのずれを解消する Discrepancy Resolution Loss を提案する．これにより PLMs に対して，ユーモアの検出性能の向上を試みる．HaHackathon データセットにおいて本提案手法を用いることで，ユーモア検出性能が安定して向上する傾向にあり，特に Recall が改善する．さらに，提案手法の適用による予測確率の摂動の観察や損失関数への異なるモデリングを行うことで，「ずれ」の解消を行う損失関数の導入が効果的であることを示す．入力文のコンテクストとコモンセンスの埋め込みの類似度に関して，その分布を分析することにより，PLM やコモンセンスに対する損失関数の効果と傾向を詳細に評価する．

キーワード ユーモア検出，Commonsense-aware Attention，コモンセンス，事前学習済み言語モデル，キーワード抽出，不一致理論

1 はじめに

笑いの要因になりうるユーモアは，日々のコミュニケーションの重要な要素である．他者と笑うことによる高齢期での機能不全のリスク軽減 [1] や笑うことによるストレス応答抑制と認知機能改善 [2] など，多くの医学研究が笑いの効用を示している．このことから，自動的なユーモアの理解や生成を対話システムへ実装することで，ユーザの心身の満足度が向上すると考えられる．

笑いやユーモアをシステムに組み込むためには，ユーモアを機械的に理解することが重要である．一般的にユーモアの一種であるジョークはいくつかの文から構成されており，ストーリーを構成するフリとそれを結論づけるオチがある．ジョークには，言葉あそびやステレオタイプの指摘，誤解，皮肉などの要素がある．これらの要素が面白さを生み出すが，同時にユーモアの認識の難しさも生み出している．ユーモアを理解することは，AI だけでなく人間にとっても容易なことではない．心理学研究における笑いの不一致理論 [1]によると，期待と実際に感受したことの「ずれ」を認知したときに笑いが起こるとされる [3–6]．適切な知識を持つことではじめて，正しい期待を持つことができ，「ずれ」を認知できる．文の修辞的な構成がユーモアを理解するトリガーになることはあるが，確実にユーモアを理解するためには的確で深いコモンセンスや百科事典的な知識が必要である．Sterwart [7] は，スペイン語圏の人と英語圏の人との会話を通じて，文化の違いが笑いの誤解を引き起こす可能性があることを示している．このことから，ユーモアや笑いを理解するには正しい知識が必要であることがわかる．

これまで，自動的なユーモア検出手法として様々なモデルが提案されてきた．しかし，コモンセンスや百科事典的な知識に着目した手法は少ない．Badri ら [8] は，マルチモダリティを用いたラフトラックの予測を行った．予測のエラー分析によって，適切な知識ベースが必要なユーモアは正しく予測できない典型的な事例であることを示した．このことから，自動的なユーモア検出手法においても，人間と同様に，適切な知識ベースにアクセスする必要が考えられる．

本研究では，上記のような問題に対処するため，次の二つを提案する．一つ目として，コモンセンスに着目し，PLM に対して，知識ベースを CA-MHA（**C**ommonsense-**A**ware **M**ulti-**H**ead **A**ttention）により外挿するユーモア検出手法を提案する．自然言語からコモンセンスを再構築する GPT ベースデコーダ [9] である COMET [10] を知識ベース，Transformer ベース [11] の PLM をコンテクストのエンコーダとして用いる．二つ目として，不一致理論に着目し，非ユーモアデータに対してコモンセンスから得られる期待と入力文のコンテクストの「ずれ」を解消する損失関数 DRLoss（**D**iscrepancy **R**esolution **Loss**）を提案する．PLM のみでは考慮できない知識を与え，不一致

1：不一致理論（incongruity theory）は不調和仮説とも呼ばれる．

理論をモデリングすることで，ユーモア理解の性能向上を試みる．これら 2 つの提案手法を組み合わせることで，特にユーモアデータに対して，検出性能の改善傾向が確認できた．さらに，DRLoss において，ユーモアデータにおける期待とコンテクストのずれの認知を目的としたモデリングを追加し，異なるモデリングの効果を分析する．

本研究の貢献は次のようになっている:

- CA-MHA を用いて外部知識にソフトにアクセスするユーモア検出の抽象的なアーキテクチャを提案する．
- 笑いの不一致理論に基づく DRLoss を提案し，期待とコンテクストの「ずれ」の解消をモデリングする．
- 上記 2 つを組み合わせることにより，特にユーモアデータに対する性能において，ユーモア検出性能の向上を達成する．
- 異なるモデリングの DRLoss の提案と検証を行い，提案手法の効果を詳細に分析する．コンテクストとコモンセンスの埋め込みの類似度の分布の分析により，提案する DRLoss の有効性とユーモアに対するモデリングの難しさを示す．
- 複数の PLM を用いた実験により，PLM と提案手法の関係性を分析し，PLM に対する本提案手法の効果を示す．

2 関 連 研 究

Mihalcea と Strapparava [12] はテキストのスタイル特徴量とコンテンツベース特徴量を用いて，古典的な機械学習手法でユーモア検出を行った．Chen ら [13] は CNN と Highway Networks を導入することで，ディープラーニングを用いた検出手法を提案した．Weller と Seppi [14, 15] は Reddit[2]からユーモア検出のための大規模なデータセットを構築した．彼らはそのデータセットに対し，Transformer ベースの事前学習済みモデルをファインチューニングすることで検出タスクを行った．Annamoradnejad と Zoghi [16] は，多くのジョークはフリとオチから構成されることに着目し，入力テキストを文ごとに区切って Transformer に入力することで文章間の関係性を考慮する ColBERT を提案した．また，SemEval ではユーモア検出タスクのコンペティションが開催されており [17, 18]，様々なシステムが提案され，事前学習済み Transformer モデルが多く採用されている．

多くのユーモアに関する手法が提案されているが，近年では事前学習済み Transformer モデルの利用が多く，コモンセンスや百科事典的な知識を明示的に扱っている手法は多くない．その中で，Zhang ら [19] は Wikipedia から抽出される単語の知識のトリプレットをグラフ構造としてエンコードすることで，より一貫性のあるオチの自動生成を試みた．

他のタスクにおいては，コモンセンスにアクセスすることで性能向上を試みた研究がいくつか存在する．Li ら [20] は入力テキストから関連するコモンセンスを抽出し，2 種類の知識選択戦略を比較することで，コモンセンスがどのように皮肉検出性能に影響を与えるかを調査した．Chowdhury と Chaturvedi [21] はコモンセンスをグラフ構造として捉え，GCN を適用することで皮肉検出を行った．また，Yang ら [22] はコモンセンスを精神状態の知識として扱い，スピーカーの精神状態を明示的にモデル化した．共感対話においても同様にコモンセンスを用いることで，共感した発話応答を試みた研究がある [23, 24].

これらの先行研究を踏まえ，本研究では，外部知識としてのコモンセンスを用い，これをソフトにモデルに統合することで，モデルの知識の増強を試みる．また，心理学研究の知見に基づいたモデリングを行い，モデルの学習とその影響の詳細な分析を行う．

3 提 案 手 法

提案モデルを図 1 に示す．提案モデルは主に次の 3 つのパートから構成されており，(1) Context Encoder, (2) Commonsense Acquisition Module, (3) Commonsense-aware Humor Classification Module である．また，損失関数 Discrepancy Resolution Loss を提案する．ユーモア検出性能の向上と不一致理論のモデリング方法の検証のため，3 タイプのモデリングを行う．

3.1 Context Encoder

Context Encoder は入力文のコンテクストを表す埋め込みを出力する．このエンコーダとして, BERT [25] や RoBERTa [26] などの Transformer ベースの PLM を用いる．入力文から得られるトークン列に対して，最初にコンテクストと対応する特殊トークン([CLS] または<s>)を付加した $X = \{x_0, x_1, ..., x_{L-1}\}$ を PLM に与えることで，特殊トークン（すなわち x_0）に対応する埋め込みを $\mathbf{h}_{CTX}$ として得る:

$$\mathbf{H}_{CTX} = PLM(X) \tag{1}$$

$$\mathbf{h}_{CTX} = \mathbf{H}_{CTX}[0] \tag{2}$$

ここで，$\mathbf{H}_{CTX} \in \mathbb{R}^{L \times d}$，$\mathbf{h}_{CTX} \in \mathbb{R}^{d}$ であり，L はトークン長，d は PLM の出力の次元数である．特殊トークンに対応した $\mathbf{h}_{CTX}$ は入力文全体の意味を表現する．

3.2 Commonsense Acquisition Module

Commonsense Acquisition Module は, さらに Keyword Extractor と Commonsense Encoder から構成される．モジュールの詳細を図 2 に示す．

3.2.1 Keyword Extractor

入力文章から連想されるコモンセンスを捉えるためには，文を構成する単語やフレーズが持つ概念を捉えることが必要である．これを実現するため，予備実験として，後続する Commonsense Encoder に対して，入力文，入力文の要約文，入力文のキーワードの 3 種類の入力を試みた．Commonsense Encoder に用いるモデルの制約のため，入力文と入力文の要約文は適しておらず，入力文のキーワードを入力とする手法が最も有効であった．そのため，このモジュールでは入力文から Keyword Extractor を用いて，文章の中で重要とされる単

2：https://www.reddit.com/

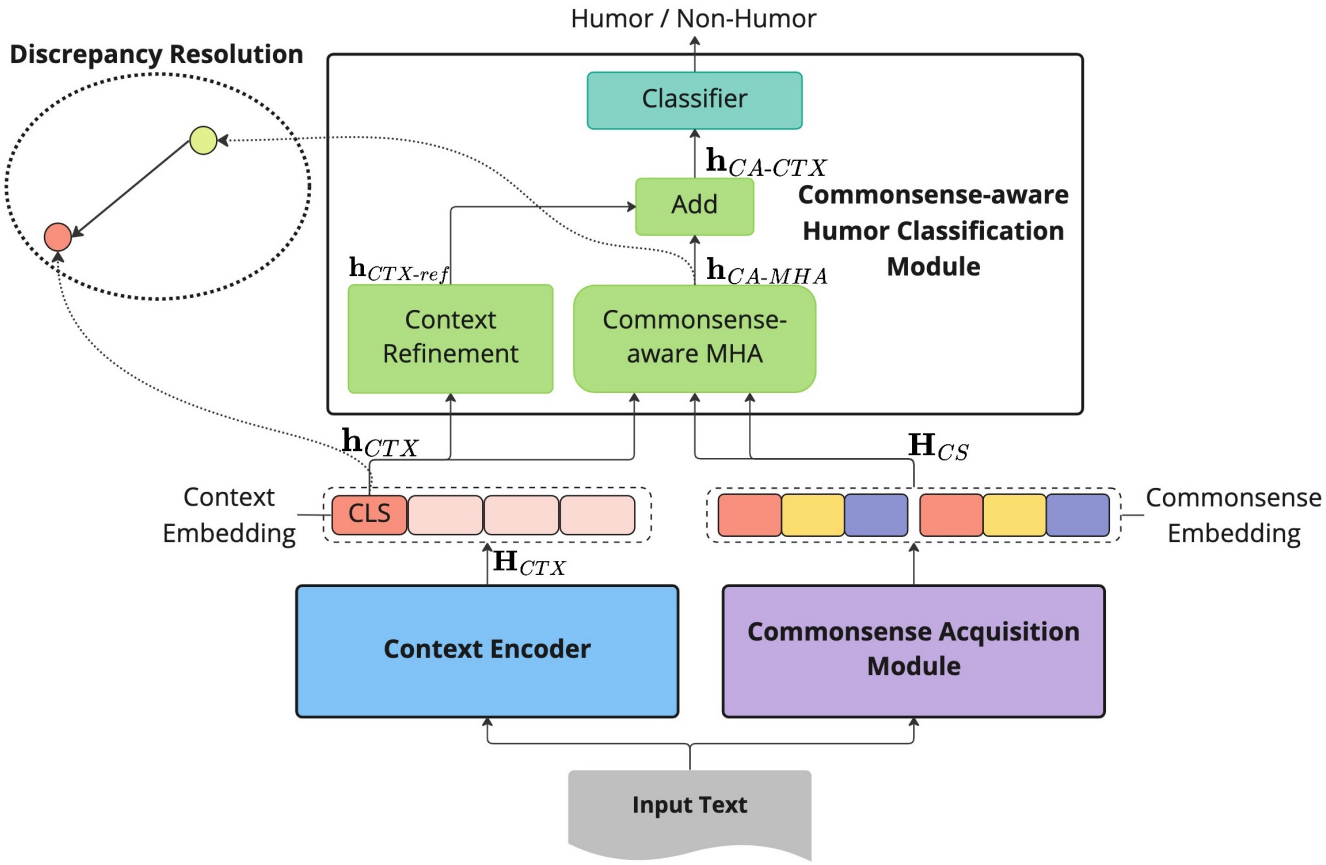


図 1: 提案モデルのアーキテクチャ

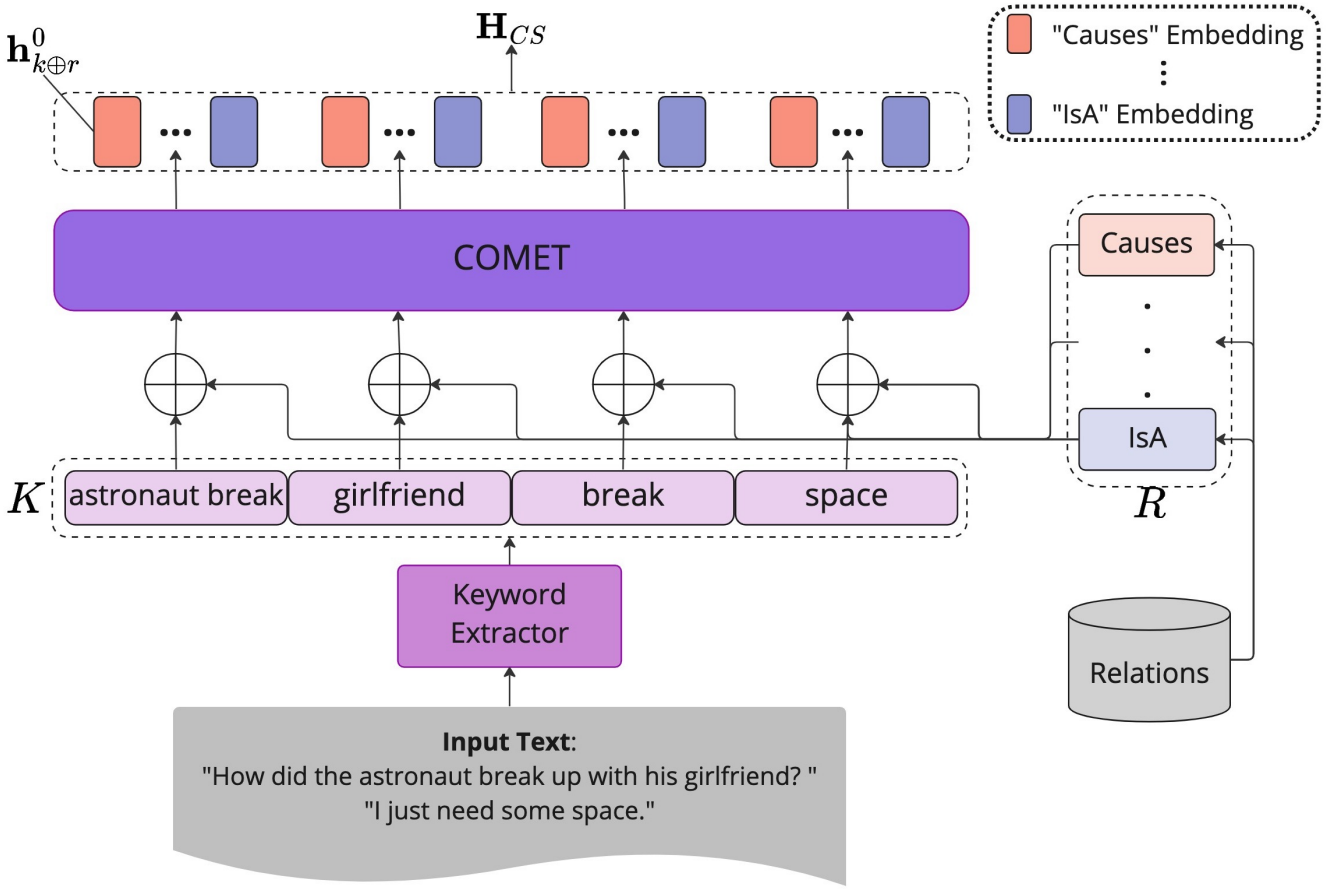


図 2: Commonsense Acquisition Module の詳細

語やフレーズを抽出する．その結果として，キーワードの集合 $K = \{k_1, k_2, ..., k_m \mid m \leqq M_{kwd}\}$ が得られる．M_{kwd} は抽出する最大キーワード数を表す．

Keyword Extractor として YAKE! [27] を用いる．この抽出器は単語の被りを許容する閾値やキーワードの最大 n-gram をハイパーパラメータとして設定できるため，容易にキーワードの出力を調整できる．本研究では，キーワード候補スコアに基づく top-m のキーワードを抽出する．

3.2.2 Commonsense Encoder

Commonsense Encoder は，Keyword Extractor により抽出されたキーワードのコモンセンスを表現する埋め込みを出力する．このエンコーダとして COMET を用いた．COMET は GPT ベースのデコーダであり，入力単語やフレーズからコモンセンスを自然言語として生成するように事前学習されたモデルである．このモデルには，ConceptNet [28] と ATOMIC [29] の二つのデータセットのそれぞれで学習されたモデルが存在する．ConceptNet は単語レベルのコモンセンスの知識ベースであり，単語の概念に関する記述のグラフ構造で構成される．グラフのエッジはリレーションで表現され，“IsA” や “Causes” などの合計 34 個のリレーションが存在する．ATOMIC は日々の人々のインタラクションに関する大規模な if-then 知識ベースである．本実験では，抽出されたキーワードに関する概念的なコモンセンスを対象とするため，ConceptNet で事前学習された COMET を使用する．入力はキーワードとリレーションをテキストとして連結した文である．例えば，キーワードが “astronaut break”，リレーションが “IsA” ならば，“astronaut break IsA” を入力する．よって，キーワード集合 K とリレーション集合 $R = \{r_1, r_2, ..., r_n\}$ から，COMET への入力 $K' = \{k_1 \oplus r_1, k_1 \oplus r_2, ..., k_1 \oplus r_n, k_2 \oplus r_1, ..., k_m \oplus r_n\}$ を構築する．$\oplus$ はテキストの結合操作を示す．また，コモンセンスを文として明示的に生成する代わりに，最後のトークンに対応する最終層の表現ベクトルをコモンセンスの埋め込みとすることで，COMET をエンコーダとして作用させる．これによって，Context Encoder から得られるベクトルと COMET が生成する知識の親和性を向上させることができ，知識ベースをモデルにソフトに統合することを期待する．Commonsense Encoder は以下のように定式化できる:

$$\mathbf{H}^i_{k\oplus r} = COMET(K'[i]) \tag{3}$$

$$\mathbf{h}^i_{k\oplus r} = \mathbf{H}^i_{k\oplus r}[l_i - 1] \tag{4}$$

$$\mathbf{H}_{CS} = [\mathbf{h}^0_{k\oplus r}, \mathbf{h}^1_{k\oplus r}, ..., \mathbf{h}^{m\cdot n-1}_{k\oplus r}] \tag{5}$$

ここで，$\mathbf{H}^i_{k\oplus r} \in \mathbb{R}^{l\times d}$ はキーワードとリレーションの 1 ペアに対する COMET の出力，$\mathbf{h}^i_{k\oplus r} \in \mathbb{R}^d$ は入力の最後のトークンに対応した埋め込みであり，$\mathbf{H}_{CS} \in \mathbb{R}^{(m\cdot n)\times d}$ はキーワードベースのコモンセンス埋め込みを表現している．また，$i \in \{0, 1, ..., m \cdot n - 1\}$ であり，l_i は $K'[i]$ におけるトークン長，d は COMET の出力次元である．

3.3 Commonsense-aware Humor Classification Module

このモジュールはさらに Context Refinement, Commonsense-aware MHA & Add，Humor Classifier の 3 つのコンポーネントから構成される．

3.3.1 Context Refinement

このモジュールでは，式 2 で得られるコンテクスト埋め込み $\mathbf{h}_{CTX}$ をユーモア理解のために洗練し，それを $\mathbf{h}_{CTX\text{-}ref}$ として出力する:

$$\mathbf{h}_{CTX\text{-}ref} = \sigma(\mathbf{W}_{ref}\mathbf{h}_{CTX} + \mathbf{b}_{ref}) \tag{6}$$

ここで，$\mathbf{h}_{CTX\text{-}ref} \in \mathbb{R}^d$ であり，$\mathbf{W}_{ref} \in \mathbb{R}^{d\times d}$，$\mathbf{b}_{ref} \in \mathbb{R}^d$，$\sigma$ はハイパボリックタンジェントである．

3.3.2 Commonsense-aware MHA & Add

このモジュールでは，コンテクスト埋め込み $\mathbf{h}_{CTX}$ とコモンセンス埋め込み $\mathbf{H}_{CS}$ を統合する．その際，コンテクストをユーモアの判断のベースとして，キーワードが持つ概念的なコモンセンスを背景知識のように考慮することを期待する．CA-MHA を用いて，コンテクストから連想または必要とされるコモンセンスの表現ベクトルを獲得し，それをユーモア検出のために洗練したコンテクスト埋め込みに加算することでコモンセンスの統合を実現する．Multi-Head Attention を $MHA(query, key, value)$ と定式化すると，このモジュールは次のようになる:

$$\mathbf{h}_{CA\text{-}MHA} = MHA(\mathbf{h}_{CTX}, \mathbf{H}_{CS}, \mathbf{H}_{CS}) \tag{7}$$

$$\mathbf{h}_{CA\text{-}CTX} = \mathbf{h}_{CTX\text{-}ref} + \mathbf{h}_{CA\text{-}MHA} \tag{8}$$

ここで，$\mathbf{h}_{CA\text{-}MHA} \in \mathbb{R}^d$ はコンテクストによって選択されたコモンセンスの情報を表現し，$\mathbf{h}_{CA\text{-}CTX} \in \mathbb{R}^d$ はコモンセンスにより増強されたコンテクストを表している．

3.3.3 Humor Classifier

Commonsense-aware MHA & Add によって増強されたコンテクストの埋め込みである $\mathbf{h}_{CA\text{-}CTX}$ は分類器に入力され，ユーモア検出が行われる．

$$P = softmax(\mathbf{W}_c\mathbf{h}_{CA\text{-}CTX} + \mathbf{b}_c) \tag{9}$$

ここで，P は各ラベルにおける予測確率を表しており，$\mathbf{W}_c \in \mathbb{R}^{2\times d}$ と $\mathbf{b}_c \in \mathbb{R}^2$ はそれぞれ分類器の重みとバイアスである．

3.4 Discrepancy Resolution Loss

心理学研究における，期待とのずれを認知したときに笑いが起こる不一致理論に着目し，この損失関数を導入する．ユーモアには様々なタイプが存在するため，期待とのずれを適切に定式化することは難しいと考えられる．そこで，非ユーモアにおけるコモンセンスから得られる期待とコンテクストのずれを解消する Distance Type の損失関数を提案する．また，不一致理論のモデリング方法の違いによる影響を検証するため，ユーモアを認識できる期待とのずれの程度を定義した Direction Type

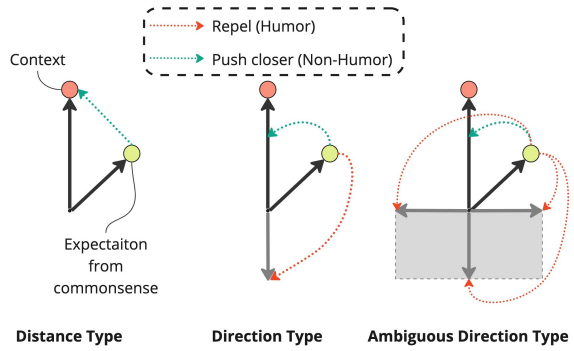


図 3: Discrepancy Resolution Loss のモデリングのイメージ

及び Ambiguous Direction Type を提案する．合計 3 タイプの損失関数を定義し，それぞれのモデリングのイメージを図 3 に示す．

3.4.1 Distance Type

Distance Type では，非ユーモアデータの場合，感受するコンテクストとコモンセンスから得られる期待のずれが小さいと仮定し，意味空間における向きだけでなく，意味の強さを表すノルムも近づけることを試みる．そこで，非ユーモアにおいてコンテクストとコモンセンスの埋め込みを平均二乗誤差により近づける．ただし，ユーモアにおけるコンテクストと期待のずれ方を一意に定めることは難しいため，ユーモアに対するモデリングを行わない．次式により，損失 $\mathcal{L}_{dist}$ を計算する．

$$\mathcal{L}_{dist} = \|\mathbf{h}^{-}_{CA\text{-}MHA} - \tilde{\mathbf{h}}^{-}_{CTX}\|^2 \tag{10}$$

ここで，$\tilde{\mathbf{h}}^{-}_{CTX}$ は勾配から切り離した非ユーモアラベルを持つコンテクストの埋め込み，$\mathbf{h}^{-}_{CA\text{-}MHA}$ は CA-MHA から得られるコモンセンスの埋め込みを表す．

3.4.2 Direction Type

Direction Type では，Distance Type と異なり，意味空間における向きのみに着目し，ユーモアデータに対してコンテクストと期待のずれを大きくするモデリングを試みる．ここでは，「ずれ」を正反対の意味であると仮定し，非ユーモアおいて cos 類似度を 1 に近づけ，ユーモアにおいて cos 類似度を -1 に近づける操作を行う．この操作は次式で示される．

$$sim = \frac{\mathbf{h}_{CA\text{-}MHA} \cdot \tilde{\mathbf{h}}_{CTX}}{\|\mathbf{h}_{CA\text{-}MHA}\|\|\tilde{\mathbf{h}}_{CTX}\|} \tag{11}$$

$$sim_{norm} = \frac{sim + 1}{2} \tag{12}$$

$$\mathcal{L}_{dir} = -y \log(1 - sim_{norm}) - (1 - y) \log sim_{norm} \tag{13}$$

ここで，sim はコンテクストとコモンセンスの埋め込みの cos 類似度，$sim_{norm} \in [0, 1]$ は確率表現化された類似度，y は正解ラベルである．$\tilde{\mathbf{h}}_{CTX}$ は $\mathbf{h}_{CTX}$ を勾配から切り離したものである．

3.4.3 Ambiguous Direction Type

Ambiguous Direction Type では，Direction Type と同様，cos 類似度に基づく損失を計算する．ただし，ここでは「ずれ」の定義を広くとり，無関係な意味や正反対の意味もずれとみなす．cos 類似度が 0 以下のものをずれとして，全てを同様に扱うことで，ずれの定義を広くする．このモデリングは次式で示される．

表 1: HaHackathon データセットの統計情報

	正例	負例	合計
訓練	4,932	3,068	8,000
検証	632	368	1,000
評価	615	385	1,000

$$sim = \frac{\mathbf{h}_{CA\text{-}MHA} \cdot \tilde{\mathbf{h}}_{CTX}}{\|\mathbf{h}_{CA\text{-}MHA}\|\|\tilde{\mathbf{h}}_{CTX}\|} \tag{14}$$

$$sim_{norm} = max(0, sim) \tag{15}$$

$$\mathcal{L}_{amb} = -y \log(1 - sim_{norm}) - (1 - y) \log sim_{norm} \tag{16}$$

ここで，$max(\cdot, \cdot)$ は 2 つの入力の最大値を選び出す関数である．

3.5 目的関数

ユーモア検出の学習のため，クロスエントロピー誤差をラベル y における予測確率 $P(y)$ に対して使用し，$\mathcal{L}_{CE}$ を計算する．

$$\mathcal{L}_{CE} = -log(P(y)) \tag{17}$$

この損失関数と DRLoss のうち 1 つの損失関数の重み付和を最小化することでモデルのパラメータを最適化する．

$$\mathcal{L} = \mathcal{L}_{CE} + \alpha \cdot \mathbf{g}^T \begin{pmatrix} \mathcal{L}_{dist} \\ \mathcal{L}_{dir} \\ \mathcal{L}_{amb} \end{pmatrix} \tag{18}$$

ここで，α は 2 つの損失関数の貢献度をコントロールするハイパーパラメータであり，$\alpha \geqq 0$ を満たす．また，$\mathbf{g} \in \mathbb{R}^3$ は 3 種類の DRLoss から使用する 1 つのみを選び出す one-hot ベクトルである．

4 実　　験

4.1 データセット

公開データセットである HaHackathon データセットを用いてモデルの学習及び評価を行う．HaHackathon の統計情報を表 1 に示す．このデータセットは SemEval 2021 task 7[3]において使用されたデータセットである．データソースは 80%が Twitter であり，残りは Kaggle Short Jokes[4]である．ユーモア検出のアノテーションのため，各テキストは 20 名のアノテータによって評価されており，「このテキストの意図はユーモアにすることですか？」という質問への回答によってアノテーションされている．ユーモアのラベルはこの回答の多数決によって決定されている．本実験では，公開されている訓練／検証／評価セットを用いて，2 値分類としてのユーモア検出タスクを行う．

3：https://competitions.codalab.org/competitions/27446
4：https://www.kaggle.com/datasets/abhinavmoudgil95/short-jokes

表 2: DRLoss を用いない提案手法の実験結果

PLM	#rel.	Prec.	Rec.	F1
BERT	-	0.923	0.918	0.920
	5	0.916	0.908	0.912
	34	0.914	0.918	0.916
RoBERTa	-	0.948	0.904	0.925
	5	0.940	**0.917**	**0.928**
	34	0.930	**0.917**	0.923
DistilBERT	-	0.925	0.907	0.916
	5	**0.930**	**0.908**	**0.919**
	34	0.924	**0.915**	**0.919**
DeBERTa	-	0.942	0.941	0.942
	5	0.935	0.939	0.937
	34	**0.952**	0.921	0.937

4.2 実験設定

コモンセンスを使用しない PLM をベースラインとして使用し，HuggingFace で公開されている BERT[5]，RoBERTa[6]，DistilBERT[7] [30]，DeBERTa-v3[8] [31, 32] を用いた．提案手法の Context Encoder の重みの初期値にも同様のモデルを使用した．コモンセンスを生成する COMET の実装には comet-commonsense を使用し，ConceptNet により学習された事前学習済みモデルを採用した [9]．オプティマイザーには AdamW [33] を採用し，ベースライン及び Context Encoder として用いる PLM の学習率は 2e-5，その他モジュールは 5e-5 と設定し，COMET の重みは固定した．また，バッチサイズは 32 とした．COMET に入力するリレーションには，“IsA”，“HasA”，“Causes”，“Desires”，“UsedFor” の 5 つか，34 個全てのリレーションを用いた．この 5 つのリレーションは著者らが直感的に理解しやすいと考え，これらを選択した．YAKE!のパラメータには，dedupLim を 0.3，n-gram を 3，使用するキーワードの最大個数 M_{kwd} を 6 とした．HaHackathon データセットはデータ数が少ないため，過学習を防ぐために PLM におけるドロップアウト率を 0.4 に設定した．CA-MHA の head 数を 8 とし，α として 1, 0.5, 0.05 を実験した．学習の最大エポック数を 15 と設定し，検証データにおいて最も損失が小さいエポックにおけるモデルを最良モデルとし，評価データを用いてモデルの性能を評価した．

5 実験結果

提案手法とベースラインの結果を表 2，3，4 に示す．評価指標として，Prec.（Precision），Rec.（Recall），F1 を採用する．PLM を共通するモデルにおいて，最も F1 の高い時の α を最適値として結果を表に示している．表 2 において，ベースラインの PLM に対し，提案手法によって向上した指標を太字で表記している．表 3，4 において，コモンセンスのみを用いる手法に対し，DRLoss を導入することで向上した指標を太字で表記している．また，表 2 に示されたベースラインの PLM における F1 に対し，向上した数値を下線で表示している．

5：https://huggingface.co/bert-base-uncased
6：https://huggingface.co/roberta-base
7：https://huggingface.co/distilbert-base-uncased
8：https://huggingface.co/microsoft/deberta-v3-base
9：https://github.com/atcbosselut/comet-commonsense

表 3: Distance Type を用いた提案手法の実験結果

PLM	#rel.	DRLoss	α	Prec.	Rec.	F1
BERT	5	-	-	0.916	0.908	0.912
		$\mathcal{L}_{dist}$	0.05	0.916	**0.957**	**0.936**
	34	-	-	0.914	0.918	0.916
		$\mathcal{L}_{dist}$	0.5	**0.923**	**0.943**	**0.933**
RoBERTa	5	-	-	0.940	0.917	0.928
		$\mathcal{L}_{dist}$	0.05	0.930	**0.938**	**0.934**
	34	-	-	0.930	0.917	0.923
		$\mathcal{L}_{dist}$	0.05	**0.941**	0.910	**0.925**
DistilBERT	5	-	-	0.930	0.908	0.919
		$\mathcal{L}_{dist}$	0.5, 1	0.930	**0.910**	**0.920**
	34	-	-	0.924	0.915	0.919
		$\mathcal{L}_{dist}$	0.05	0.922	0.915	0.919
DeBERTa	5	-	-	0.935	0.939	0.937
		$\mathcal{L}_{dist}$	0.05	**0.947**	0.917	0.932
	34	-	-	0.952	0.921	0.937
		$\mathcal{L}_{dist}$	1	0.951	**0.960**	**0.956**

表 4: Direction Type または Ambiguous Direction Type を用いた提案手法の実験結果

PLM	#rel.	DRLoss	α	Prec.	Rec.	F1
BERT	5	-	-	0.916	0.908	0.912
		$\mathcal{L}_{dir}$	0.05	**0.917**	**0.957**	**0.937**
		$\mathcal{L}_{amb}$	1	**0.921**	**0.952**	**0.936**
	34	-	-	0.914	0.918	0.916
		$\mathcal{L}_{dir}$	1	**0.917**	0.900	0.908
		$\mathcal{L}_{amb}$	0.5	**0.917**	0.902	0.909
RoBERTa	5	-	-	0.940	0.917	0.928
		$\mathcal{L}_{dir}$	1	0.931	**0.930**	**0.930**
		$\mathcal{L}_{amb}$	1	0.928	**0.928**	0.928
	34	-	-	0.930	0.917	0.923
		$\mathcal{L}_{dir}$	0.5	0.929	**0.936**	**0.932**
		$\mathcal{L}_{amb}$	1	**0.948**	0.892	0.919
DistilBERT	5	-	-	0.930	0.908	0.919
		$\mathcal{L}_{dir}$	0.5	0.929	**0.915**	**0.922**
		$\mathcal{L}_{amb}$	0.5	0.929	**0.915**	**0.922**
	34	-	-	0.924	0.915	0.919
		$\mathcal{L}_{dir}$	0.05, 0.5, 1	0.924	0.915	0.919
		$\mathcal{L}_{amb}$	0.05	0.924	0.915	0.919
DeBERTa	5	-	-	0.935	0.939	0.937
		$\mathcal{L}_{dir}$	0.05	**0.942**	0.908	0.925
		$\mathcal{L}_{amb}$	0.05	**0.944**	0.934	**0.939**
	34	-	-	0.952	0.921	0.937
		$\mathcal{L}_{dir}$	1	0.942	**0.938**	**0.940**
		$\mathcal{L}_{amb}$	0.05	**0.958**	0.912	0.935

まず，コモンセンスの効果を評価するため，DRLoss を用いない手法について評価を行う．表 2 によると，5 つのリレーションを用いた場合，RoBERTa と DistilBER が，全てのリレーションを用いた場合，DistilBERT のみが，PLM に対して F1 が改善している．工夫なしにコモンセンスを扱うだけでは，性能向上を図ることは難しいことがわかる．

次に，DRLoss の Distance Type の影響を評価する．表 3 を見ると，例えば BERT と 5 つのリレーションを用いた手法では，DRLoss を用いない手法に対して，Rec. は 0.908 から 0.957，F1 は 0.912 から 0.936 へと向上している．DRLoss を用いない手法に対して，Rec. の向上に伴って F1 の改善が見られる場合が多く，ユーモアを検出する機会の向上が期待できる．また，全てのリレーションを用いた場合，DistilBERT を用いるモデルにおいて性能改善は見られず，反対に DeBERTa では性能の改善がある．このことから，Context Encoder に用いる PLM の表現力によって，扱うべき知識や知識の選択肢が異なると考えられる．

続いて，DRLoss の比較として提案した手法である Direction Type と Ambiguous Direction Type の影響を評価する．表 4 によると，それぞれの手法は 5 つのリレーションを用いた場合，性能が向上する傾向が見られたが，全てのリレーションを用いると悪化する傾向にある．また，PLM やリレーション数によって，性能の改善傾向が一定ではないため，ユーモアに対するずれの定義や DRLoss のモデリング方法を改める必要がある．

PLM，リレーション数，DRLoss のタイプの組み合わせによって，最適となるハイパーパラメータ α が異なっている．そこで，性能を改善するためにはハイパーパラメータのチューニングが必要である．このチューニングを行えば，Distance Type は PLM やリレーション数の変化に関わらず，比較的安定して性能を改善しており，非ユーモアにおける不一致理論のモデリングは一定の効果がある．

6 分　　析

6.1 コモンセンスの効果

コモンセンスを考慮する CA-MHA の導入がユーモア検出に与える影響を分析する．ここでの分析においてはどの PLM を用いても同様の結果が得られているため，BERT を用いた提案手法における分析を提示し，コモンセンスの影響を考察する．また，コモンセンスの影響のみを評価するため，DRLoss を用いない手法を用いる．

コモンセンスを用いることによる，ベースラインの BERT に対する各ラベルでの予測確率の摂動を図 4，5 に示す．BERT における予測確率の昇順でデータをソートし，BERT とコモンセンスを用いた提案手法の予測確率を比較することで，コモンセンスの効果を観察する．閾値は 0.5 に設定されており，図上の線より上のデータ点は正解，下のデータ点は誤りの予測であることを示している．ラベルがユーモアであるデータにおける図 4a，5a に着目すると，コモンセンスを用いた提案手法では，BERT の予測確率が高いデータに対して出力を維持している．反対に，BERT の予測の信頼度が曖昧または低いデータに対して，コモンセンスを用いることで予測確率を変動させる傾向がある．ラベルが非ユーモアであるデータにおける図 4b，5b においても同様の傾向が見られるが，予測確率が高いもの対しても大きな確率の変動を与えることが多く，非ユーモアの予測の信頼度が下がりやすいことが分かる．このことから，Context Encoder の PLM の予測結果をベースとしてコモンセンスを考慮した推論を行なっており，提案手法は，PLM が正しく予測できないデータの予測の修正を試みる形式で，推論を行うことがわかる．BERT における予測ラベルから変動を伴ったデータにおける予測確率の変化を図 6 に示す．図 6a の上段は提案手法によって正しく予測ラベルを変換したデータ，下段は誤って予測ラベルを変換したデータにおける予測確率の変化を示している．データ点に付与された赤線の長さは，予測確率の変化の大きさを表す．リレーション数の変化によらず，コモンセンスを用いることで，確かに BERT の曖昧な予測を主に変換している．しかし，全体として性能が改善しておらず，その原因は，BERT の適切な予測まで誤って変換するデータも多いことである．コモンセンスを工夫なしに導入することは適切でなく，必要なコモンセンスを抽出するための手法を組み込むことが必要であることが示唆される．

6.2 DRLoss の効果

ここでは 3 タイプの DRLoss の効果の評価を行うため，ベースラインに対する予測確率の変化，attention weights，コンテクスト埋め込み $\mathbf{h}_{CTX}$ とコモンセンス埋め込み $\mathbf{h}_{CA-MHA}$ の類似度の分布を分析する．

6.2.1 予測確率の変化

図 7 は，BERT に対して 5 つのリレーションを用いた提案手法における，PLM のみの BERT に対する予測確率の変化を示し，DRLoss による影響を比較した図である．図 7a と図 7b，7c，7d を比較すると，DRLoss を用いない手法に対し，DRLoss を用いることでユーモアの検出数が増加し，非ユーモアと誤って予測変換するデータがなくなっている（各図右上及び右下）．これによって，BERT では DRLoss の適用により性能が大きく向上していると考えられる．

次に，最も性能の高い DeBERTa において，DRLoss が機能した Distance Type と機能していない Ambiguous Direction Type を比較して分析する．図 8 は DeBERTa に対して全てのリレーションを用いた提案手法における，DeBERTa に対する予測確率の変化と DRLoss の導入による影響を示している．PLM のみの DeBERTa では Recall が高く，元々ユーモアの検出率が高いことが表 2 からわかっており，ユーモアの件数をさらに増加させることは難しいと考えられる．実際に実験結果から，Precision が上がる場合が多く，コモンセンスがノイズとなっている結果が見られている．図 8a 右側を見ると，Distance Type の DRLoss の導入により，ユーモアの検出数はさらに増加し，ユーモアデータにおいて適切だった DeBERTa の予測が誤って予測変換される件数は少ない．しかし，図 8b において，Ambiguous Direction Type の適用により，ユーモアに対

する検出性能が大きく落ちている．DRLoss の適用により検出性能が向上する場合は，ユーモアデータにおいて適切な予測変換を行う傾向が，性能が低下する場合は，ユーモアデータにおいて非ユーモアと誤って予測変換する数が増加する傾向が見られた．この結果から，使用する PLM に合ったリレーション数や α のチューニングができれば，DRLoss はユーモアに対する予測性能の大幅な向上に貢献すると考えられる．

6.2.2 attention weights

次に attention weights の分析を行う．BERT において 5 つのリレーションを用いた提案手法の attention weights の一例を図 9 に示す．各ヒートマップ上段の P/L は予測/ラベルを表現している．この例では，いずれのタイプの DRLoss を取り入れることで，正解を導くことができているが，アテンションが向けられるコモンセンスは大きく変化していない．しかし，Ambiguous Direction Type，Direction Type，Distance Type，w/o DRLoss の順にアテンションが分散しやすい傾向が見られる．コンテクストとコモンセンスの埋め込みを操作する損失を与えることで，必要なコモンセンスを抽出する強度を向上させ，これがモデルの性能向上につながると考えられる．

また，全てのリレーションを用いた場合に Distance Type の導入によって，精度が改善した DeBERTa を例にとって分析を行う．図 9 と同じ入力について示す図 10 を参照すると，Distance Type を用いたモデルだけが正解を導いている．図 10a，10c，10d と図 10b を比較すると，取り出すコモンセンスの傾向が異なっている．コモンセンスの選択肢が増加すると，抽出の組み合わせが増加するため，適切にモデリングされた DRLoss によって抽出する知識の制御を学習することで，多くのコモンセンスから正しい期待を導くことが可能であると示唆される．

6.2.3 コンテクストとコモンセンスの埋め込みに類似度の分布

また，DRLoss の導入によるコンテクスト埋め込み $\mathbf{h}_{CTX}$ とコモンセンス埋め込み $\mathbf{h}_{CA\text{-}MHA}$ の各正解ラベルにおける類似度の分布を観察し，異なるモデリングの特徴の違いと課題を分析する．ここでは一般的によく用いられる BERT，全てのリレーションを用いた場合に DRLoss の導入で性能の変化が起こらない DistilBERT と Distance Type において大幅な改善が見られた DeBERTa の分布を観察する．その分布を図 11，12，13，14 に示す．左側のグラフが平均二乗誤差，右側のグラフが cos 類似度の分布の図となっている．まず，図 11a，12a，13a，14a に着目する．ここでのモデルは DRLoss を用いないため，ユーモアと非ユーモアにおいて同様の分布を示している．

図 11，12，13，14 の (a) を除く図を見ると，3.4 節での DRLoss の各タイプのモデリングの意図に沿った分布の動きが見られる．Distance Type を用いると，非ユーモアは平均二乗誤差の減少に伴い，cos 類似度も大きくなっている．ユーモアに対する損失は計算されないため，ユーモアは非ユーモアへの学習に影響され，分布は類似度の高い方向へ移動し，裾の長い分布を形成している．それに対して，Direction Type または Ambiguous Direction Type を用いると，非ユーモアの分布は cos 類似度の高い方向，ユーモアは低い方向へ移動する．その分布の変化に伴い，平均二乗誤差の分布も大きく移動している．この分布の移動から，平均二乗誤差による操作と cos 類似度による操作は関係性の近い操作であることがわかる．DRLoss によって，コンテクストとコモンセンスの類似度の分布を移動させ，適切な知識の獲得を試みることで性能改善が見込まれる．

しかし，Distance Type における cos 類似度の分布間の位置関係や距離はおおよそ変化しないが，Direction Type と Ambiguous Direction Type における分布間の距離はモデルによって大きく変化する．Context Encoder に用いる PLM によって変化する分布の不安定さが，性能改善への寄与度のモデル依存性につながることが示唆される．また，全く見当違いな言及が面白いユーモアや期待を少し外したユーモアなど様々なタイプがあり，ユーモアにおける期待とコンテクストのずれの定義は難しい．そのため，ユーモアに対する類似度の分布の恣意的な操作は，特定のタイプのユーモアに対応する操作であり，不安定な結果が得られていると考えられる．反対に Distance Type は比較的安定して性能向上に寄与しており，類似度の高い方向にユーモアでの分布が移動していることから，ユーモアは期待との少しのずれを持っていることが多い可能性が挙げられ，より詳細な検証によって傾向を調査するべきである．非ユーモアに対してずれの解消を行うことは効果的であるが，ユーモアに対するずれの認知のモデリングにはさらなる検討が必要である．

最後に，PLM の表現力とコモンセンスに対する DRLoss の影響について考察する．ここでは，提案手法の DRLoss のタイプである Distance Type に着目する．図 11b と図 12b において cos 類似度の分布を比較すると，全てのコモンセンスを用いることで，分布が類似度の高い方向に大きく移動している．図 14b では，非ユーモアの分布が BERT を用いたモデルに対して，より cos 類似度の高い方向に大きく移動している．しかし図 13b によると，分布がほとんど移動していない．これらのことから，DeBERTa のように PLM の表現力が高いほど，より多くのコモンセンスの選択肢を持つことで DRLoss の影響を大きく受けて，分布を移動させることがわかる．反対に，DistilBERT のように PLM の表現力が小さいと，コモンセンスの選択肢の増加によって，必要なコモンセンスを適切に抽出できなくなり，DRLoss の影響が小さくなると考えられる．そこで，パラメータ数や事前学習データが多い PLM に対して，コモンセンスの選択肢を多く与え自動的に抽出することを学習させることが有効だが，それらが小さい PLM に対しては，コモンセンスの選択肢を削減し，必要な知識を事前に決定することが必要である．

7 まとめと今後の展望

ユーモア検出においてコモンセンスを増強する Commonsense-aware Attention を用いた抽象的なアーキテクチャと，心理学研究における不一致理論に基づいた Dircrepancy Resolution Loss を提案した．これらの提案手法により，ユーモア検出性能が向上した．特に Recall の改善傾向が見ら

れたことから，ユーモアに対して反応しやすくなることが示唆された．ユーモアを認知できる期待とコンテクストの「ずれ」を定義することで 3 タイプの DRLoss を提案し，それらの影響を詳細に分析した．これにより，非ユーモアにおける不一致理論のモデリングは有効であり，様々なタイプが存在するユーモアに対するモデリングは不安定で難しいことがわかった．

今後の展望として，ユーモアにおける不一致理論のモデリングの精査，リレーションの選択戦略の設定，コンテクストと期待のずれを認知するモジュールの開発などが挙げられる．これらの追加実験とアーキテクチャの改良を行い，コンテクストの感受とコモンセンスによる期待に基づいた信頼度の高いモデルの構築を行う．

謝　　辞

本研究は JSPS 科研費 JP22K12271，JP19K12230，JP19K11411 の助成を受けたものである．

文　　献

[1] Yudai Tamada, Chikae Yamaguchi, Masashige Saito, Tetsuya Ohira, Kokoro Shirai, Katsunori Kondo, and Kenji Takeuchi. Does laughing with others lower the risk of functional disability among older japanese adults? the jages prospective cohort study. *Preventive Medicine*, 2022.

[2] 山越 達矢, 阪本 亮, 西垣 翔梧, 田中 爽太, 福田 隆文, 金留 理奈, 鈴木 久仁厚, 梁 弘一, 小山 敦子, and 阿野 泰久. 笑いによるストレス応答抑制と認知機能改善効果. **日本健康心理学会大会発表論文集**, 34:87, 2021.

[3] Henri Bergson, Cloudesley Shovell Henry Brereton, and Fred Rothwell. *Laughter: An essay on the meaning of the comic*. Macmillan, 1914.

[4] Lambert Deckers and Philip Kizer. Humor and the incongruity hypothesis. *The Journal of Psychology*, 90(2):215–218, 1975.

[5] Lambert Deckers, Steven Jenkins, and Eric Gladfelter. Incongruity versus tension relief: Hypotheses of humor. *Motivation and Emotion*, 1:261–272, 1977.

[6] Lambert Deckers and John Devine. Humor by violating an existing expectancy. *The Journal of Psychology*, 108(1):107–110, 1981.

[7] Stuart Stewart. The many faces of conversational laughter. 1997.

[8] Badri N. Patro, Mayank Lunayach, Deepankar Srivastava, Sarvesh Sarvesh, Hunar Singh, and Vinay P. Namboodiri. Multimodal humor dataset: Predicting laughter tracks for sitcoms. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021.

[9] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[10] Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, July 2019.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 2017.

[12] Rada Mihalcea and Carlo Strapparava. Making computers laugh: Investigations in automatic humor recognition. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, October 2005.

[13] Peng-Yu Chen and Von-Wun Soo. Humor recognition using deep learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. Association for Computational Linguistics, June 2018.

[14] Orion Weller and Kevin Seppi. Humor detection: A transformer gets the last laugh. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, November 2019.

[15] Orion Weller and Kevin Seppi. The rJokes dataset: a large scale humor collection. In *Proceedings of the 12th Language Resources and Evaluation Conference*. European Language Resources Association, May 2020.

[16] Issa Annamoradnejad and Gohar Zoghi. Colbert: Using bert sentence embedding for humor detection, 2020.

[17] Nabil Hossain, John Krumm, Michael Gamon, and Henry Kautz. SemEval-2020 task 7: Assessing humor in edited news headlines. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*. International Committee for Computational Linguistics, December 2020.

[18] J. A. Meaney, Steven Wilson, Luis Chiruzzo, Adam Lopez, and Walid Magdy. SemEval 2021 task 7: HaHackathon, detecting and rating humor and offense. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*. Association for Computational Linguistics, August 2021.

[19] Hang Zhang, Dayiheng Liu, Jiancheng Lv, and Cheng Luo. Let's be humorous: Knowledge enhanced humor generation. *CoRR*, 2020.

[20] Jiangnan Li, Hongliang Pan, Zheng Lin, Peng Fu, and Weiping Wang. Sarcasm detection with commonsense knowledge. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2021.

[21] Somnath Basu Roy Chowdhury and Snigdha Chaturvedi. Does commonsense help in detecting sarcasm? In *Proceedings of the Second Workshop on Insights from Negative Results in NLP*. Association for Computational Linguistics, November 2021.

[22] Kailai Yang, Tianlin Zhang, and Sophia Ananiadou. A mental state knowledge–aware and contrastive network for early stress and depression detection on social media. *Information Processing & Management*, 2022.

[23] Sahand Sabour, Chujie Zheng, and Minlie Huang. Cem: Commonsense-aware empathetic response generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022.

[24] Quan Tu, Yanran Li, Jianwei Cui, Bin Wang, Ji-Rong Wen, and Rui Yan. Misc: A mixed strategy-aware model integrating comet for emotional support conversation. *arXiv preprint arXiv:2203.13560*, 2022.

[25] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, June 2019.

[26] 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.

[27] Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. Yake! keyword extraction from single documents using multiple local features. *Information Sciences*, 2020.

[28] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

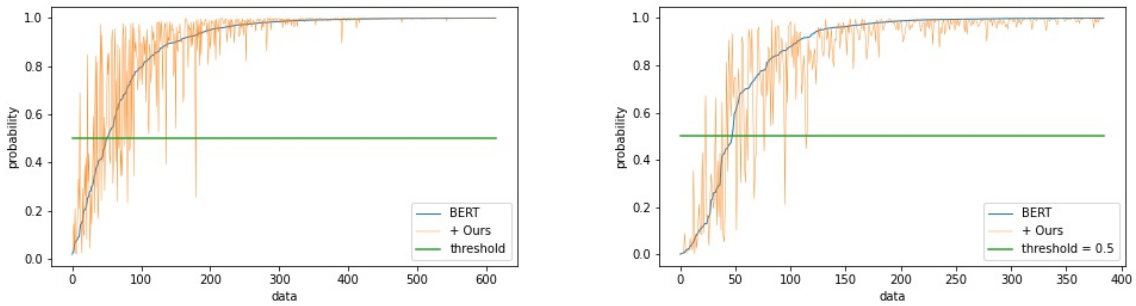
[29] Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI conference on artificial intelligence*, 2019.

[30] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[31] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*, 2020.

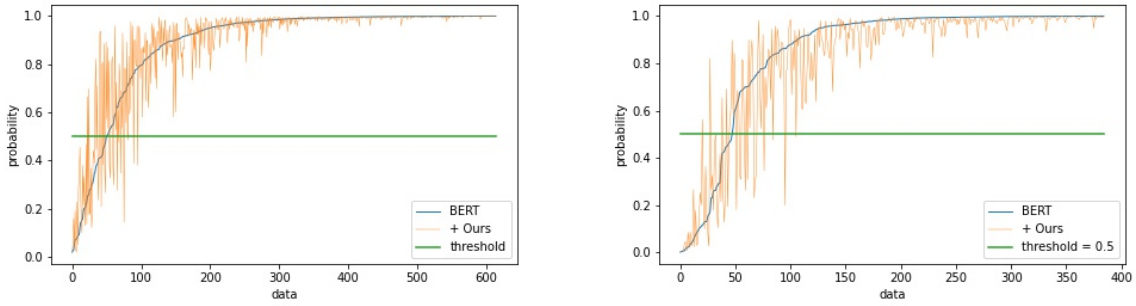
[32] Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*, 2021.

[33] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018.



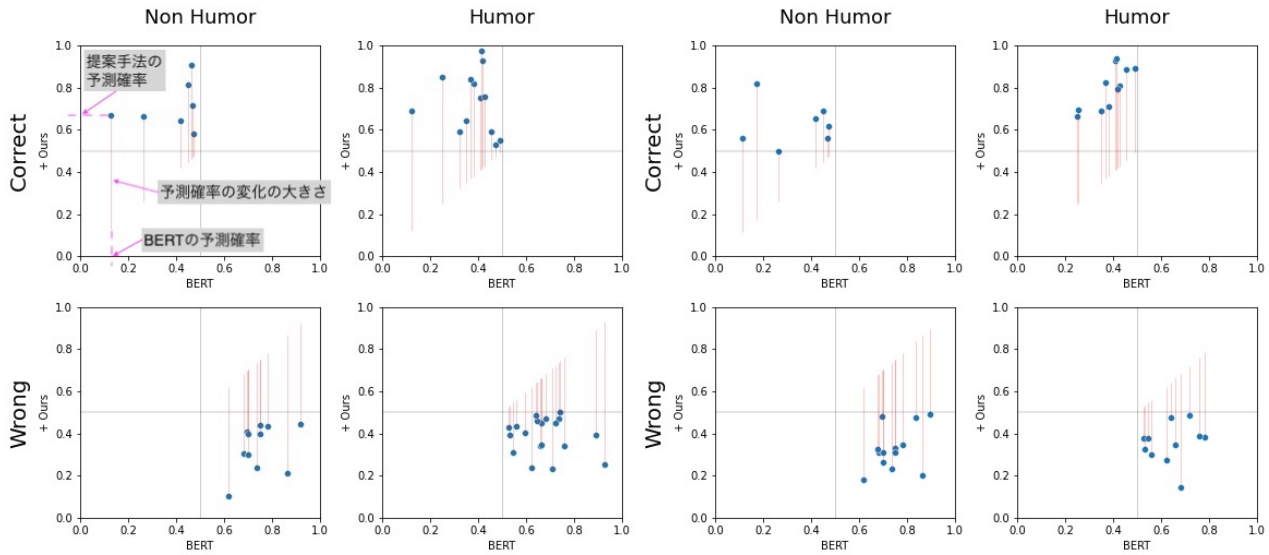
(a) ラベルがユーモアであるデータ　　**(b)** ラベルが非ユーモアであるデータ

図 4: BERT と 5 つのリレーションを用いた提案手法における，BERT に対する予測確率の摂動



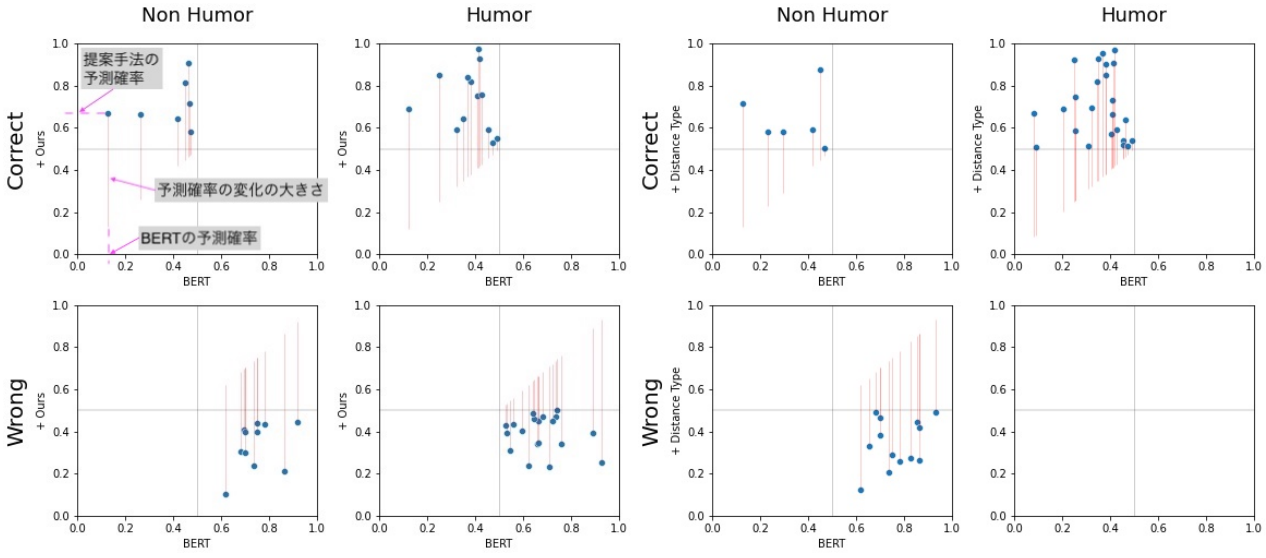
(a) ラベルがユーモアであるデータ　　**(b)** ラベルが非ユーモアであるデータ

図 5: BERT と全てのリレーションを用いた提案手法における，BERT に対する予測確率の摂動



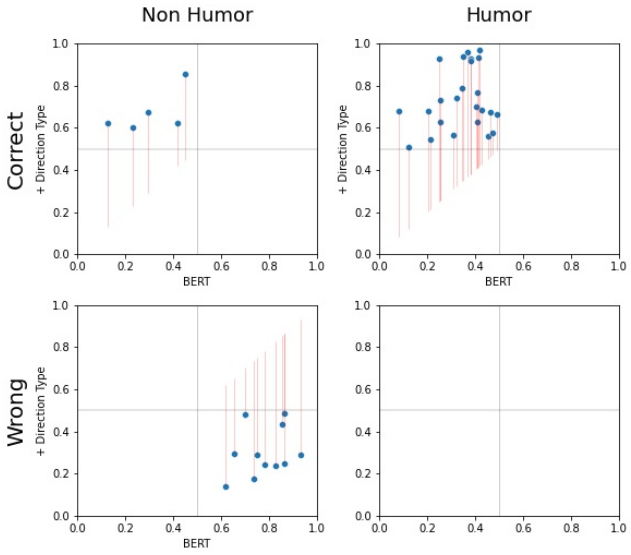
(a) 5 つのリレーションを用いた場合　　**(b)** 34 個全てのリレーションを用いた場合

図 6: BERT を用いて DRLoss を用いない提案手法における，BERT に対する予測ラベル変動時の正解ラベルの予測確率の変化とリレーション数による影響の違い

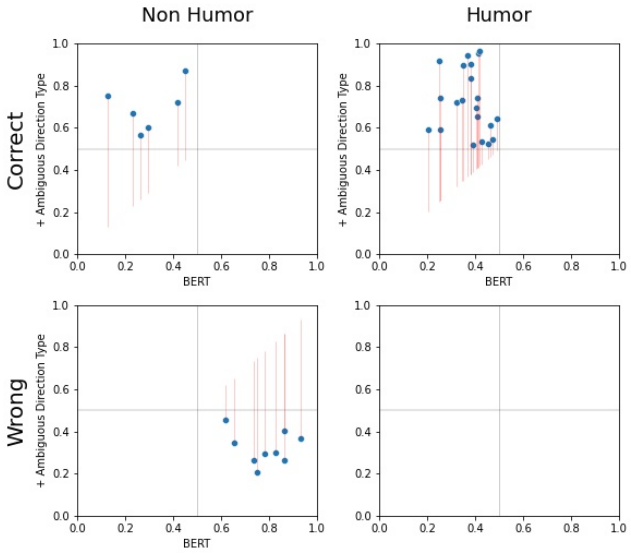


(a) w/o DRLoss

(b) w/ Distance Type

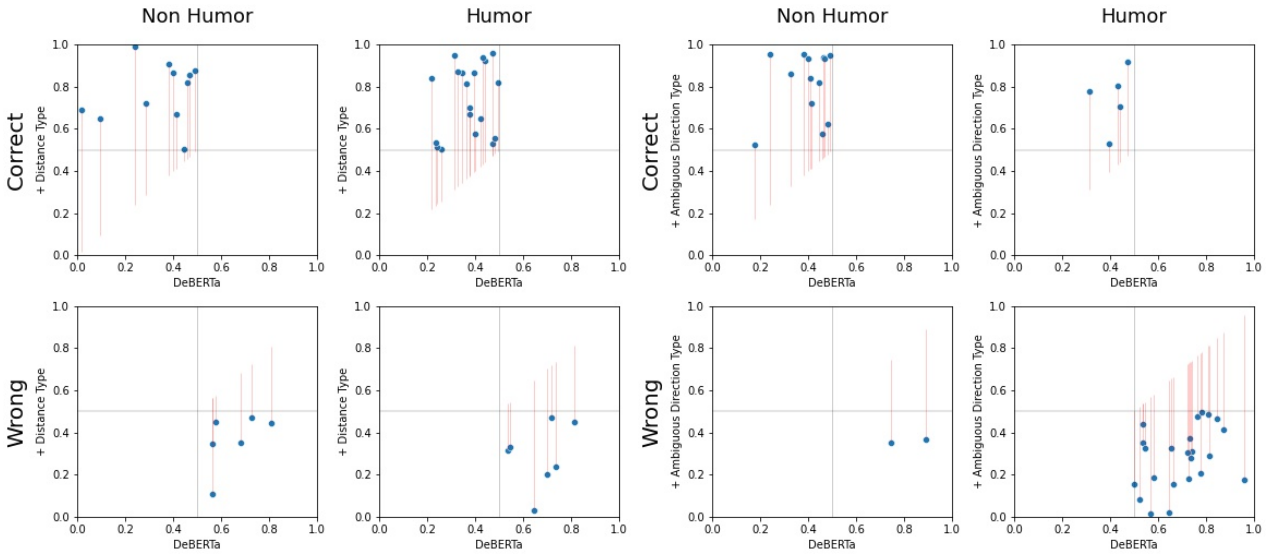


(c) w/ Direction Type



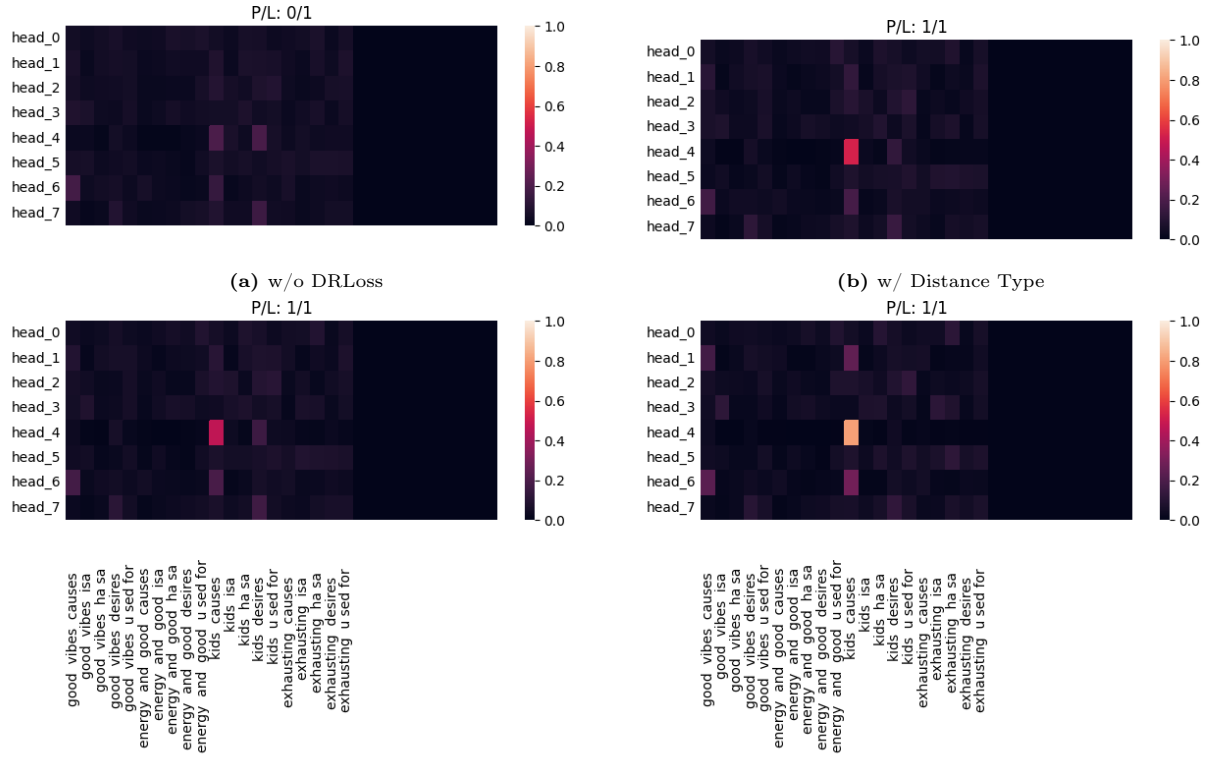
(d) w/ Ambiguous Direction Type

図 7: BERT と 5 つのリレーションを用いた提案手法における，BERT に対する予測ラベル変動時の正解ラベルの予測確率の変化と DRLoss の導入による影響の違い



(a) w/ Distance Type **(b)** w/ Ambigous Direction Type

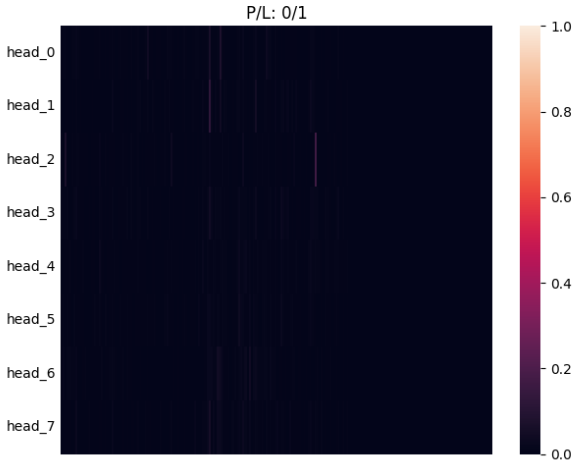
図 8: DeBERTa と全てのリレーションを用いた提案手法における，DeBERTa に対する予測ラベル変動時の正解ラベルに対する予測確率の変化と DRLoss の影響の違い



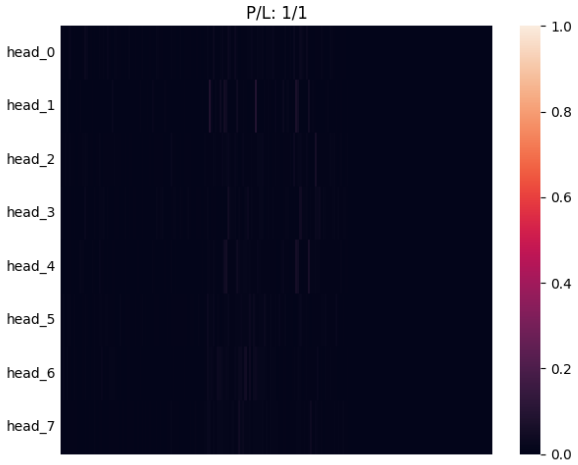
(a) w/o DRLoss **(b)** w/ Distance Type

(c) w/ Direction Type **(d)** w/ Ambiguous Direction Type

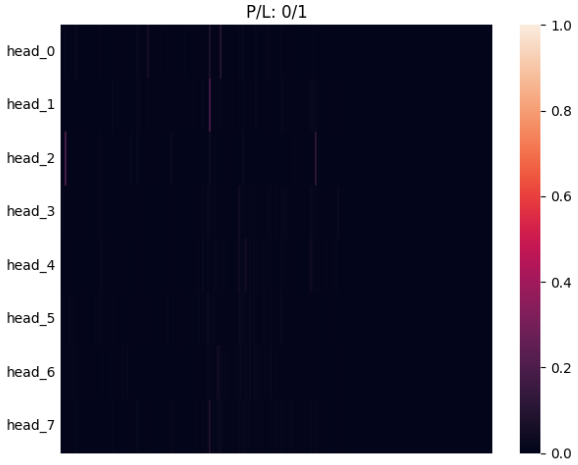
図 9: BERT と 5 つのリレーションを用いた提案手法における，コモンセンスに対する attention weights と DRLoss の導入による影響の違い



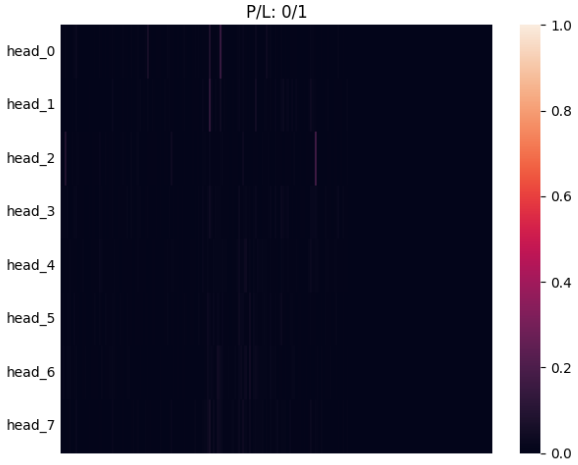
(a) w/o DRLoss



(b) w/ Distance Type

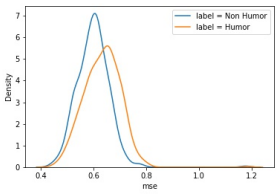


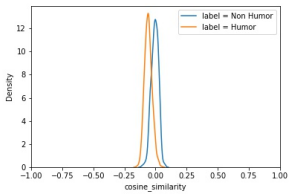
(c) w/ Direction Type



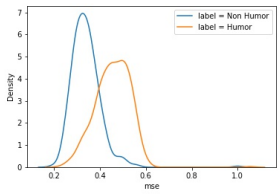
(d) w/ Ambiguous Direction Type

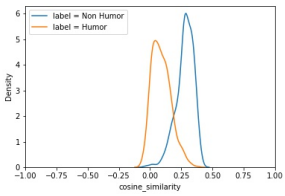
図 10: DeBERT と全てのリレーションを用いた提案手法における，コモンセンスに対する attention weights と DRLoss の導入による影響の違い



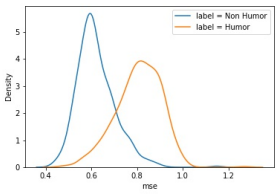


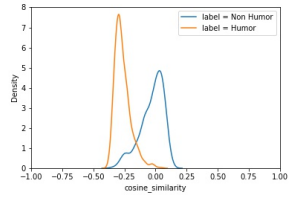
(a) w/o DRLoss



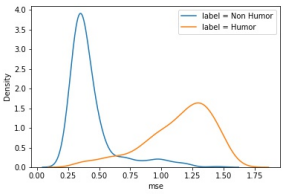


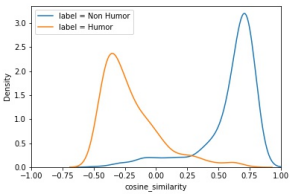
(b) w/ Distance Type





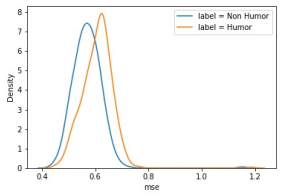
(c) w/ Direction Type

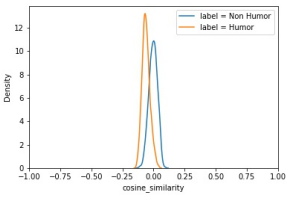




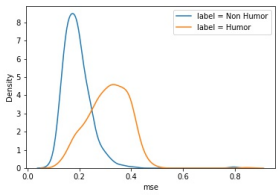
(d) w/ Ambiguous Direction Type

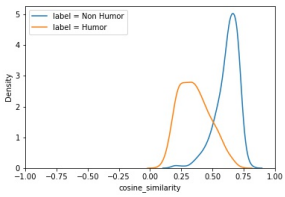
図 11: BERT と 5 つのリレーションを用いた提案手法における，$\mathbf{h}_{CTX}$ と $\mathbf{h}_{CA\text{-}MHA}$ の類似度の分布と DRLoss の導入による影響の違い



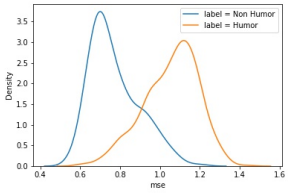


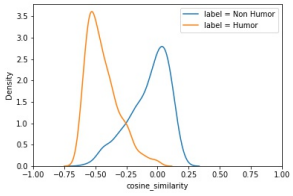
(a) w/o DRLoss



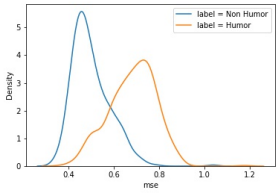


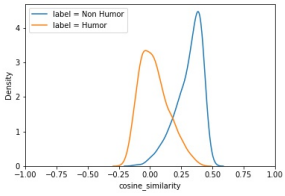
(b) w/ Distance Type





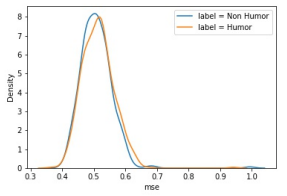
(c) w/ Direction Type

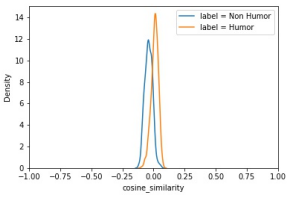




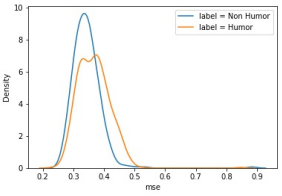
(d) w/ Ambiguous Direction Type

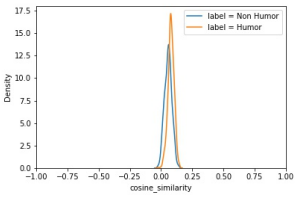
図 12: BERT と全てのリレーションを用いた提案手法における，$\mathbf{h}_{CTX}$ と $\mathbf{h}_{CA\text{-}MHA}$ の類似度の分布と DRLoss の導入による影響の違い



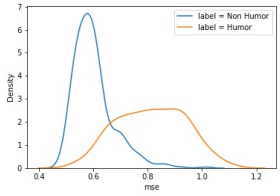


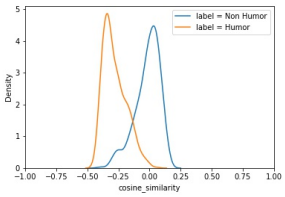
(a) w/o DRLoss



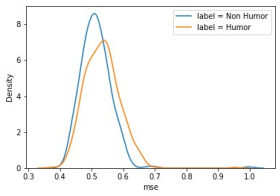


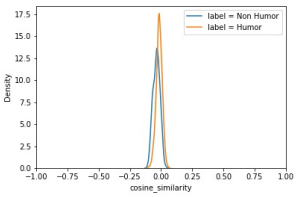
(b) w/ Distance Type





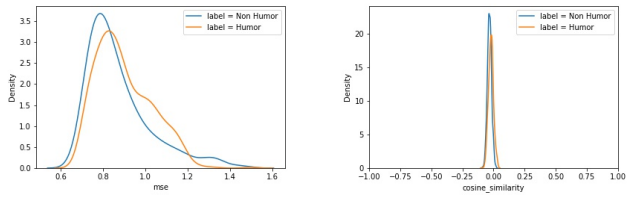
(c) w/ Direction Type



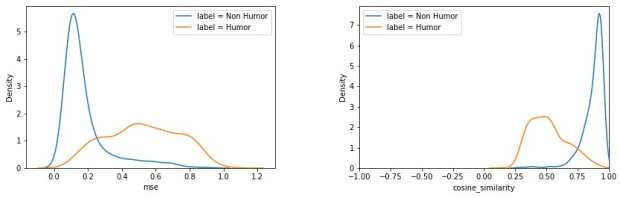


(d) w/ Ambiguous Direction Type

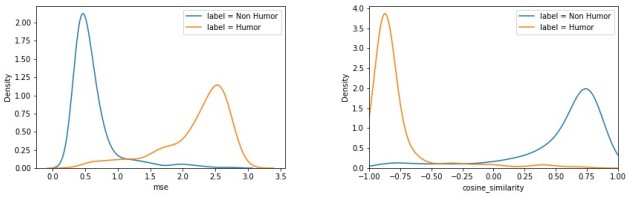
図 13: DistilBERT と全てのリレーションを用いた提案手法における，$\mathbf{h}_{CTX}$ と $\mathbf{h}_{CA\text{-}MHA}$ の類似度の分布と DRLoss の導入による影響の違い



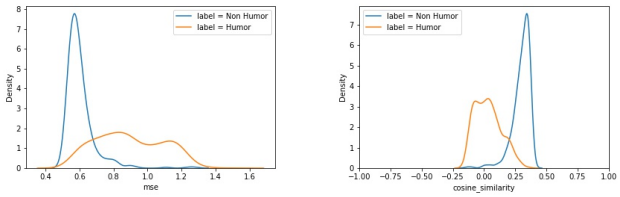
(a) w/o DRLoss



(b) w/ Distance Type



(c) w/ Direction Type



(d) w/ Ambiguous Direction Type

図 14: DeBERTa と全てのリレーションを用いた提案手法における，$\mathbf{h}_{CTX}$ と $\mathbf{h}_{CA\text{-}MHA}$ の類似度の分布と DRLoss の導入による影響の違い