修士論文

BERT の各層出力に着目した
評価者間のラベルの不一致と一致の判別

小林 大記

2024 年 4 月 12 日

岐阜大学大学院 自然科学技術研究科 知能理工学専攻 知能情報領域
鈴木研究室

本論文は岐阜大学大学院 自然科学技術研究科に
修士（工学）授与の要件として提出した修士論文である.

小林 大記

指導教員：
鈴木 優 准教授

BERT の各層出力に着目した 評価者間のラベルの不一致と一致の判別*

小林 大記

内容梗概

本研究の目標は，評価者間でのラベルの一致・不一致の判別することにより，追加のラベル付けにおける作業件数を低減することである．ラベル付けでは，文章の言語の微妙な差異により評価者により付けられるラベルが不一致となる場合があるため，追加のラベル付けが必要となり，作業件数は増加する．そこで，ラベルの一致・不一致を判別することで，この問題を解決する．本研究の目的は，BERT の各層に着目することで，ラベルの一致・不一致の判別を行うことである．本研究では，ラベルの一致・不一致の言語の微妙な差異を捉えるため，BERT の各層の出力を使用した．BERT の各層は，テキストから異なる種類の特徴を捉えることがわかっているため，ラベルの一致・不一致を判別ができると考えた．評価実験では，追加のラベル付け後のデータを用いて，ラベルの一致・不一致のそれぞれクラスターの分離の度合いを各層において比較した．実験の結果，層毎による違いがなく，データの分離がないことがわかった．BERT の各層の出力から，ラベルの一致・不一致を判別することが可能な層を見つけることができないことが明らかになった．

キーワード

BERT，クラウドソーシング

*岐阜大学大学院 自然科学技術研究科 知能理工学専攻 知能情報領域 修士論文, 学籍番号: 1224525033, 2024 年 4 月 12 日.

目次

図目次

表目次

第 1 章　はじめに

既存データセットを教師あり学習に用いる際，一つデータに対してラベルが一意に定まらない場合，追加のラベル付けが必要となる．ラベル付けを行う方法には，クラウドソーシング [1] が挙げられるが，これには費用が多くかかる問題がある．そこでわれわれは費用低減のため，既存データセットにおける追加のラベル付け作業を効率化することに着目した．

ラベル付けは，教師あり学習モデル構築に必要な，データセットを作成するために行われる．本研究では，各データに一つのラベルを与えられたデータセットを想定している．データセットを作成するためには，工程が 2 つある．一つ目は，学習に用いるために必要なデータを収集する工程である．この工程では，目的に沿ったデータを Web API といったツールにより収集をする．二つ目は，収集したデータに対して評価者がラベルを付ける工程である．この工程では，複数の評価者がデータに対して感情や属性に関するラベル付けを，タスクによって与えられた指示に沿って行う．このようにして，既存データセットでは各データに対して，複数の評価者により 1 票ずつラベルを付けられている．評価者によりラベルをつけられた既存のデータセット中のデータは，ラベルの一致と不一致に分けられる．ラベルの一致とは，多数決によりラベルが一意に定まる場合を指す．このデータはラベル付けの結果，各データにおいて最も票数の多いラベルを．一つに絞ることができる．一方，ラベルの不一致とは多数決によりラベルが一意に定まらない場合を指す．このデータはラベル付けの結果，各データにおいて最も票数の多いラベルが複数存在するため．一つに絞ることができない．一般にラベルの一致のデータは，各データにラベルが一意に定められたラベルがあるため，学習に用いることができる．しかし，ラベルの不一致のデータは各データにラベルが一意に定められたラベルがないため，学習に用いることができない．そこで，われわれは既存データセットにおけるラベルの不一致に着目する．そして，既存データセットにおけるラベルの不一致に対して，追加のラベル付けを行うことでラベルの不一致の解消を行う．ラベルの不一致を解消することにより，データセット上のラベルの一致のデータを増加させることができる．そして，学習に用いることができるデータ数を増やすことができる．しかし，追加のラベル付けにはラベル付け件数に応じて，評価者へ報酬を支払

うための費用が掛かるという問題がある．

上記の問題を解決するため，われわれは追加のラベル付けにおける評価者の行うラベル付け件数を減らすことで，費用の低減に繋げる．追加のラベル付けは，データセット上に含まれるラベルの不一致を解消することで，学習に利用できるデータを増加するために用いられる．既存データセット上には，ラベルの不一致のデータは多く存在することが多い．これは，作業指示の不明確さ，文章の異なる表現，悪質な評価者に起因する．ラベルの不一致のデータ数が多い場合，解消したいデータ数が増えるため，追加のラベル付けの件数もまた増加する．追加のラベル付けの前後において，ラベルの不一致のデータは 2 種類の遷移をする．一方が，ラベルの不一致から追加のラベル付け後にラベルの一致への遷移である．もう一方が，ラベルの不一致から追加のラベル付け後にラベルの不一致への遷移である．ここでは後者の追加のラベル付けの前後で，ともにラベルの不一致であるデータに着目した．これらのデータは，追加のラベル付け後であってもラベルを一つに定めることができないため，利用不可なデータであると考えた．われわれは，この利用不可なデータを追加のラベル付けの前に発見したい．そして，利用不可なデータへのラベル付けを減らすことで，追加のラベル付けにおける評価者の行うラベル付けの件数を減らすことを考えた．利用不可なデータのを発見するためには，追加のラベル付け後におけるラベルの一致・不一致を判別する必要がある．われわれは，ラベルの一致・不一致の判別のために BERT の各層に着目することで，ラベルの一致・不一致の判別を行うことを考えた．

従来の研究として，ラベルの不一致の解消を目的とした研究 [2] は存在するものの，ラベルの一致・不一致の判別に関する研究は，われわれの知る限り存在しない．そこでラベルの一致・不一致の判別のため，本研究では BERT(Bidirectional Encoder Representations from Transformers) モデル [3] の各層出力を用いた手法の提案を行う．BERT とは，自然言語処理分野で用いられる機械学習モデルである．このモデルは，12 層の transformer を元に形成しており，テキストを双方向の文脈から学習できる．そのため，文脈に応じた文章の意味を正確に捉えることができる．BERT の各層は，テキストから異なる種類の特徴を捉える能力を持っている．具体的には，BERT の下層は文の表面的な特徴，中層は構文的特徴，上層は意味的特徴を捉えることが明らかになっている [4]．BERT における上記のような各

層で捉える異なる言語の特徴が，ラベルの一致・不一致の判別に活用できると考えた．ラベルの一致・不一致の判別が困難な要因はいくつかある．要因として最も顕著であるものは，言語の微妙な差異の解釈が困難なことにあると考えた．これは作業指示の不明確さと悪質な評価者に関しても，言語の微妙な差異が影響を与えることが考えられるためである．そして，感情や意図を伴う言葉では，文脈や細かな表現の違いがラベルを変化させる．例えば，皮肉や隠喩を含む発言では文字通りに解釈した場合，想定とは異なるラベルが適用される．また，文脈の違いにより同じ単語が異なる意味を持つことがある．そのため，言語の表現の差異を深く捉えることで，複数の評価者が同じテキストに対して異なるラベルを付与した理由となる．そして，BERT の各層の出力を用いることで，文脈に依存する微妙な意味の違いや異なる評価者が捉えた様々な解釈のニュアンスを，識別することが可能になると考えられる．したがって，われわれは利用不可なデータを発見する上で，BERT の各層の出力を用いた．

具体的な提案手法では，2 つのステップに分けられる．一つ目のステップでは，ラベルの一致・不一致についてそれぞれのクラスタを作成する．まず，BERT の 1 層から 12 層での各層の出力から，K-means 法を用いて 2 つのクラスタを作成する．それから，追加のラベル付け後ラベルの一致・不一致に関するデータより，教師あり学習としてそれぞれのクラスタの識別を行う．二つ目のステップでは，ラベルの一致・不一致を判別する．ここでは，一つ目のステップにて，比較的クラスタの分離が大きい BERT の層の出力を用いる．そして，判別したいテキストの BERT の出力が，ラベルの一致または不一致のクラスタに対して，どちらのクラスタ中心に近いのかによりラベルの判別を行う．

提案手法の有効性を検証するために，実際のクラウドソーシングデータを用いた評価実験を行った．評価実験では，追加投票後におけるラベルの一致・不一致のデータの分離の度合いを層毎に比較を行った．比較には，シルエット係数の導出に用いられる凝縮度と乖離度を利用して提案手法の評価を行った．実験の結果，凝縮度と乖離度の評価指標において，層毎による違いがなく，データの分離がないことがわかった．そのため，本研究では，ラベルの一致・不一致の判別をするための BERT の層を見つけることができなかった．本論文の貢献は，BERT の各層の出力に着目することで，評価者間でのラベルの一致・不一致の判別を，層毎に比較す

ることで確認したことにある．

本論文の構成は以下の通りである．2 章では基本的事項について述べる．3 章では関連研究について述べる．4 章では提案手法について述べる．5 章では評価実験について述べる．6 章では本論文のまとめと今後の課題について述べる．

第 2 章　基本的事項

2.1　クラウドソーシング

クラウドソーシングとは，インターネットを活用して特定のタスクを，個人や組織に限定せず，不特定多数の人々に外部委託する手法である．この手法は，21 世紀にデジタル技術が発展する中で，多くの分野で用いられるようになった．クラウドソーシングは，特定の専門知識を持つ人々だけでなく，一般の人々の知識をコスト削減やタスクの効率化のための情報収集として活用できる．クラウドソーシングは，一般にインターネット上での公募を通して行われる．個人や組織は，必要なタスクをオンラインプラットフォームに投稿することで，世界中の人々からアイデアや情報を募る．投稿される作業には，データ収集やアイデア創出，文章作成といった様々な種類の作業が含まれる．クラウドソーシングは，多様な視点や能力を持つ不特定多数の人々を活用するため，従来の組織や個人のみに依存するアプローチと比べて，新たなアイデアや革新的なものを生み出す可能性を持つ．

2.2　機械学習

機械学習とは，コンピュータがデータを元に，パターンを見つけ出すことで，予測や分類を行う手法である．機械学習は，特定のタスクを実行するための，明示的な指示をせずに，データパターンを自動的に学習し理解することを目的としている．

機械学習は，教師あり学習，教師なし学習，強化学習の大きく 3 つに分類される．教師あり学習では，ラベル付きデータセットを用いて，モデルを学習し，新しいデータに対する予測を行う．教師なし学習では，ラベルがないデータからパターンを見つけ出し，データに対してクラスタリングや次元圧縮を行う．強化学習では，エージェントが，与えられたデータを手掛かりに，試行錯誤して学ぶことで，データの価値を最大化する．

機械学習を行う上では，データ収集，前処理，学習，精度の評価，未知のデータでの予測という 5 つの工程を行う．最初に行うデータ収集の工程では，解決したい課題に必要なデータを収集する．自分で収集せずとも，公開データセットを用いる

ことも可能である．次のデータの前処理の工程では，不適切な形式なデータに対して，欠損値の処理，ノイズの除去，データの正規化，特徴量の抽出といった処理を行う．モデルの学習の工程では，先に目的に沿った適切な機械学習モデルを選択する．また，前処理されたデータから訓練データを用意するそれから，選択した学習モデルを，訓練データを用いて学習を行う．精度の評価の工程では，学習したモデルに対して，どのぐらいの精度で予測ができているのかを評価する．ここでは，訓練データとは別に，テストデータを用意する．テストデータをモデルに用いることで評価を得る．最後に，未知データの予測の工程では，実際の環境での予測を行う．評価の結果，良い精度のモデルが得られたのならば，目的に沿った未知のデータに対して予測を行う．機械学習は，言語処理や画像認識，推薦システムといった様々な場面で活用される．

2.3　教師あり学習

教師あり学習とは，機械学習の一種で，ラベル付けされた教師データを使用し，モデルを学習させる手法である．教師あり学習の目的は，与えられた入力に対して，正確な出力を予測することである．教師あり学習では，まず大量のラベル付きデータを収集する．これらのデータは，モデルに特定の入力がどのような出力に関連するかを学習するために用いる．例えば，犬や猫の画像を分類する場合，各画像は犬，または猫というラベルで明示的にマークされる．このようにして，モデルは画像の特徴と犬と猫のラベル間の関連を学習する．

教師あり学習のタスクには，分類と回帰がある．前者は，入力データを事前に定義されたカテゴリに割り当てるタスクである．例としては，画像認識やスパムメールの識別が挙げられる．後者は，教師データから数値を予測するタスクである．例としては，株価の予測や気温の予測が挙げられる．教師あり学習は，明確な正解が存在する問題に対して非常に効果的である．特に複雑な関係やパターンを持つデータセットを扱う場合に有用である．そのため，先述の例のように多くの分野で活用されている．

教師あり学習の利点は，正解ラベルに基づいてモデルを学習させることができるため，特定の予測や分類タスクに対して高い性能を発揮する点にある．しかし，効

果的な教師あり学習モデルを構築するには，十分な量のラベル付きデータが必要となる．そのため，このデータの収集とラベル付けには多大な時間とコストがかかる場合がある．また，モデルが訓練データにオーバーフィッティングするリスクもあり，この問題を避けるためには適切な正則化技術やモデル検証の手法が必要になる．

2.4　教師なし学習

教師なし学習とは，ラベルや正解が指定されていないデータから，パターンや構造を見つけ出すための機械学習の手法である．この学習手法の目的は，データ自体が持つ特性や関係性を発見し，データの隠れた構造を明らかにすることにある．また，データに対する事前のラベルが不要であるため，未知のデータセットの特徴抽出に有効である．

教師なし学習には，クラスタリングと次元圧縮の大きく 2 つの手法がある．教師なし学習は，教師あり学習とは異なり，正解ラベルが与えられていないため，得られる結果の解釈や評価はより主観的になる．そのため，この手法では結果の解釈にデータの理解が必要になる．この手法は，データの隠れた傾向を発見するために用いられる．例としては，異常値の検出や顧客分析が挙げられ，幅広い分野で利用される．

教師なし学習の利点は，ラベル付けされたデータが不要であることである．ラベル付けは時間とコストがかかる作業であるため，教師なし学習はより多くのデータセットに適用可能である．そして，その傾向から新しい洞察や知識を発見する可能性がある．しかし，その結果の解釈は曖昧であることが多く，特にクラスタリングの結果は主観的な評価が必要な場合がある．

2.5　ニューラルネットワーク

ニューラルネットワークとは，人間の脳の構造と機能を模した数理モデルである．このモデルは，多数の相互接続されたノードから成り立っている．これらのノードは，データの処理と情報の伝達を行う．ニューラルネットワークの目的は，データの複雑なパターンを学習し，分類，予測といった様々なタスクを実行することで

ある．

ニューラルネットワークは，複数の層から構成されている．入力層にてデータを受け取り，一つ以上の隠れ層を通して処理を行い，最終的に出力層によって結果を出力する．各ノードは，前の層からの入力に基づいて活性化され，その結果を次の層へと伝達する．ノード間の重みは，学習中に調整され，ネットワークがデータから特徴やパターンを学習する際の要素となる．モデルの学習には，大量のデータとそのラベルが用いられる．ニューラルネットワークは，与えられた入力データに対して，ラベルとして付与された正しい出力を生成するように訓練される．この過程は，誤差逆伝播というアルゴリズムを用いて行う．このアルゴリズムは，ネットワークが生成した出力と正しい出力との間の誤差を最小限に抑える．ニューラルネットワークの設計には，隠れ層の数，ノードの数，活性化関数の種類，学習率といった多くの要素が関わる．これらのパラメータは，特定のタスクに適したものに調整する必要がある．ニューラルネットワークは，幅広い分野にて用いられる．用いられる分野として，画像認識や自然言語処理，医療診断の支援などが挙げられる．

2.6　分散表現

分散表現とは，単語や文章の意味，特性を低次元のベクトルとして表現することである．この表現は，ベクトルによって表された各単語や文章の関係性を空間内の距離や方向として捉える．分散表現は，データの複雑な特徴や意味を捉え，データ間の微妙な違いを数値として表現することができる．分散表現では，類似したデータ点は近接した位置にマッピングされる．そのため，ベクトル間のコサイン類似度やユークリッド距離といった尺度を用いることにより，単語間の意味的類似性を定量的に評価することができる．これにより，単語間の類似度や関係性を数学的に分析することができる．また，分散表現は単語の意味をベクトルの要素に分散させることができるため，複数の単語からなる表現や文の意味を合成的に捉えることも可能である．例えば，"王様"＋"女性"-"男性"という式について，それぞれの分散表現を用いて計算する．このとき，出力される単語は"女王"や"王女"が近い単語として求められる．Word2vec，GloVe，BERT といったモデルは，分散表現を利用することで構築されている．

2.7 Word2vec

Word2vec とは，分布仮説に基づいて単語の分散表現を学習する手法である．この技術は，2013 年に Google の研究チームによって開発された．そして，これはテキストデータから単語の埋め込みベクトルを学習するために広く守りいられている．Word2vec は，大規模なテキストコーパスを用いて学習され，単語間の意味的および文脈的関係を捉えたベクトルを生成する．この学習により得られた分散表現は単語の意味をしている．そのため，単語の意味的な関係を数学的に捉えることが可能である．これらのベクトルは，単語間の類似性を計算するために使用できる．また，意味的に類似した単語はベクトル空間内で互いに近くに配置される．例えば，「王」と「女王」,「男」と「女」といった単語は，類似した関係性を持つ単語としてベクトル空間内で似たような位置関係に配置される．

Word2Vec には，CBOW モデルと Skip-Gram モデルの 2 つのアーキテクチャがある．これらのアーキテクチャはいずれも，単語とその文脈の間の関係を学習することにより，単語のベクトル表現を生成する．Word2vec のモデルは，大規模なテキストコーパス上で効率的に学習を行うため，Negative Sampling や階層的ソフトマックスといった手法を使用して最適化している．これらの手法は学習プロセスを加速し，大量の語彙に対しても学習を可能にする．

Word2vec による単語埋め込みは，自然言語処理分野において，類似単語の検索や単語間の関係性の探索，複合語の意味を推測するといった場面でよく利用される．

2.7.1 CBOW モデル

CBOW(Continuous Bag-of-Words) モデルでは，ターゲットとなる単語の周囲にある文脈単語を入力として使用し，その入力をもとにターゲット単語を予測する．このモデルでは，文脈単語の集合からターゲット単語を推測することに焦点を当てているため，文脈内の単語の順序は考慮されない．

学習プロセスでは，モデルは文脈単語のベクトル表現の平均を取り，その平均ベクトルを使用してターゲット単語の予測を行う．学習の目的は，モデルが正しいターゲット単語を予測する確率を最大化することである．このプロセスを通じて，単語ベクトルは文脈内での単語の使われ方を反映するように調整される．

2.7.2 Skip-Gram モデル

Skip-Gram モデルでは，CBOW モデルの逆のアプローチを取る．こちらでは，ターゲット単語を入力として，その単語の文脈内での単語を予測する．具体的には，1 つのターゲット単語から，その周囲にある複数の文脈単語を予測している．

Skip-gram モデルの学習プロセスでは，各ターゲット単語に対して，その周囲の文脈単語を正確に予測することを目標としている．モデルは，ターゲット単語のベクトル表現を使用して，周囲の単語が出現する確率を最大化するように学習する．この手法は，特にレアな単語やフレーズで良い性能を発揮し，単語間の関係を捉えるのに効果的である．

2.8 BERT

BERT（Bidirectional Encoder Representations from Transformers）とは，自然言語処理の分野で使用される機械学習モデルである．このモデルは，Google によって 2018 年に開発された．BERT の大きな特徴として，双方向性が挙げられる．従来のモデルは，文の左から右へ，またはその逆へと一方向の文脈のみを考慮していた．しかし，BERT では，与えられた単語の前後の文脈を同時に考慮することができる．この能力により，特定の単語が文中でどのように使用されているかを，より正確に把握することを可能にした．例えば，ある文中に出現する bank という単語が，川の岸を意味するのか，金融機関を意味するのかを，文脈に応じて判断することができる．

また，BERT は transformer を元に構築されている．tranformer は，自己注意機構を用いて，文中の各単語が他の単語とどのように関連しているかを学習する．この能力により，BERT は非常に複雑な文脈関係を捉えている．

BERT の学習は二段階かけて行われる．一つ目の段階が，事前学習である．この段階では，wikipedia といった多くのテキストデータを学習することで，言語の一般的な理解を得る．事前学習タスクには、Masked Language Model と Next Sentence Prediction の 2 つがある．二つ目の段階が，ファインチューニングである．この段階では，事前学習されたモデルをベースに特定のタスクの目的に合わせてモデルを調整する．ファインチューニングで利用するタスクの例としては，質問

応答，感情分析が挙げられる．そして，少量のタスク特有のデータセットを用いて学習を行い，モデルを特定のタスクに最適化する．

2.8.1 Masked Language Model

Masked Language Model(MLM) とは，入力文からランダムに単語を選んでマスクする (隠す) ことで，そのマスクされた単語を文脈から予測させるタスクである．具体的には，入力テキストの約 15 パーセントの単語がランダムに選ばれる．そして，それらは特殊トークン [MASK] に置き換えられる．その後，BERT は残りの文脈を用いることで，これらのマスクされた単語を予測する．具体的な例として，入力文を"私は犬と散歩した。"という文を挙げる．この文章をマスクすると"私は [MASK] と散歩した。"を置き換えられる．このとき，このマスクされた単語である"犬"を文脈から予測することを行う．このプロセスにより，モデルは文脈全体を考慮して単語の意味を理解することを学習する．

MLM の主な利点は，双方向の文脈を利用できることにある．従来の言語モデルが文の左側または右側の文脈のみを考慮できた．しかし，MLM は単語の前後の文脈両方を同時に考慮して単語の意味を捉えることができる．

2.8.2 Next Sentence Prediction

Next Sentence Prediction(NSP) は，与えられた二つの文が論理的に連続しているかどうかを予測するタスクである．NSP では，モデルの入力として二つの文 A と B が与えられる．そして，BERT は文 B が文 A に続く文か，ランダムに選ばれた関連性のない文かを予測する．具体的な例としては，A:"今日は天気がいいです。"，B:"公園を散歩しようと思います。"というような 2 つの文があった際，これらが続く文章かを予測する．今回の例では，モデルが B が A に対して論理的に続くことを期待する．もし，B が"一日中家にいます。"という文であった場合，論理的には続かないことを予測することが期待される．このタスクにより，モデルは文間の関係を理解し，より広い文脈での意味を捉えることができる．

NSP は，質問応答や推論といった文の関係を理解する必要があるタスクに有用

である．モデルが文間の論理的なつながりを学習することにより，より複雑なテキスト理解が可能になる．

2.9 Transformers

Transformers は [5] 自然言語処理分野で使用される深層学習モデルのアーキテクチャである．これは，2017 年に Google の研究者たちによって発表された．基本的な構造として，transformer は，self-attention という機構を持っている．この機構を通じて，モデルは入力されたデータ内の位置に対する依存関係を効率的に学習でき，特に長い距離にある依存関係の把握に優れている．

Transformer モデルは大きく，Encoder と Decoder2 つの部分からから構成されている．Encoder は，入力テキストを高次元の特徴ベクトルに変換し，Decoder はこのベクトルを使用することで出力テキストを生成する．この二つの部分は，複数の層から構成されており，Encoder は Multi-head Attention と Feed Forward Network によって構成されている．一方，Decoder は，Encoder は Multi-head Attention と Feed Forward Network に加え，Masked Multi-Head Attention から成り立っている．self-attention の目的は，シーケンス内の各単語が，その他のすべての単語とどのように関連しているかをモデルに学習させることにある．これにより，例えば文脈上の意味をより深く理解したり，文章生成時により関連性の高い単語を選択したりすることが可能になる．

また，Transformer は，単一の attention ではなく，Multi-head Attention を使用している．これにより，モデルは複数の異なる方法で情報を集約することができ，より豊かな文脈表現を得ることが可能になる．Transformer は，順序情報を自然には取り込まないため，入力シーケンスの各位置に対する情報を含む position embedding を導入うしている．こうして，モデルは単語の順序情報も考慮することが可能になる．

Transformer の主な利点には，長距離の依存関係を効率的に捉えることができる点，計算の並列化が可能である点，そして非常に深いネットワークを構築することができる点が挙げられる．この特徴から，Transformer は自然言語処理分野で広く採用されており，BERT，GPT などにも用いられている．

2.10　トークン化

トークン化とは，テキストデータを，単語やサブワード，文字といったテキストデータの最小単位に分割することである．サブワードとは，単語をさらに小さくした意味のある単位である．例えば「ChatGPT」という単語は，「Chat」，「##G」，「##PT」というトークンに分割される．これにより，テキストデータは，トークンとして深層学習モデルがテキストデータを処理するための，基本的な入力単位となる．このトークン化には，トークナイザーを用いる．トークナイザーとは，テキストデータを個々のトークンに分割するツールである．トークナイザーは，各トークンを一意の ID にマッピングする．この ID は，トークンを表す数値である．例えば，dog は「123」，cat は「456」というように各トークンに ID が割り当てられる．

トークン化されたテキストは，モデルが理解できる形式として，数値の ID にエンコードされる．ニューラルネットワークは数値データしか処理ができない．そのため，トークンのエンコード化は，テキストデータをモデルが理解できる形式に変換するために必要である．

2.11　次元圧縮

次元圧縮とは，多次元のデータを低次元に次元数を減らすことである．このとき，データの特徴を維持しながら次元数を減らす．次元圧縮は，データの可視化，ノイズの削減，データ処理の効率化に用いられる．次元圧縮は，特徴選択と特徴抽出の 2 つに大きく分けられる．特徴選択は，元の特徴量の中から最も重要な特徴量を選び出し残りを除外する．これにより，モデルの解釈性が向上し，過学習のリスクが減少する．特徴選択は，統計的手法や情報理論，機械学習アルゴリズムに基づいて行われる．特徴抽出は，元の高次元データから新しい低次元の特徴空間を生成する．この新しい特徴空間は，元のデータセットの重要な情報を保持しつつ，次元数を削減する．特徴抽出の典型的な手法には、主成分分析，線形判別分析，t-SNE といった手法が挙げられる．次元圧縮の利点としては，次元数が減ることによる計算効率の向上や，不要な特徴の削除による精度の向上が挙げられる．

2.12 t-SNE

t-SNE[6] とは，高次元データの可視化に用いられる次元圧縮アルゴリズムである．正式名称は，t-distributed Stochastic Neighbor Embedding(t 分布型確率的近傍埋め込み) である．

t-SNE は，高次元空間におけるデータポイント間の類似性と対応する低次元空間における類似性に着目している．そして，その類似性の保持する確率分布を定義することで，これら二つの分布の類似性を最大化をしている．具体的には 3 つの手順を行う．

まず，高次元空間における類似性の計算を行う．ここでは，各データポイント間の類似性をガウス分布を用いて確率的に計算する．この確率は，ある点が他の点の近傍に選ばれる確率として解釈される．次に，低次元空間での類似性の計算を行う．初期にランダムに配置された低次元のデータ間で，t 分布を用いて類似性を計算する．t 分布を使用することにより，低次元空間におけるクラスター間が密集することを防ぎ，高次元データの構造をより忠実に再現する．最後に，カルバックライブラー発散 (KL 発散) によるコスト関数の最小化を行う．ここでは，高次元と低次元の空間における類似性の分布の差異を，KL 発散を用いて計算する．それから，勾配降下法といった最適化手法を用いてこのコスト関数を最小化する．これにより，低次元空間におけるデータの配置が調整されることで高次元空間のデータ構造をより良く反映する．

t-SNE は，高次元のデータを，低次元の空間にマッピングすることにより，データの構造を可視化しやすくする削除の仕方として，点の間の類似度が反映されるように高次元の点を低次元に埋め込みを行っている．t-SNE の利点としては，局所的な構造の保持が挙げられる．一方，欠点としては，全体的な構造を常に正確に表しているわけではない点や次元の呪いに弱い点が挙げられる．

2.13 主成分分析

主成分分析（PCA: Principal Component Analysis）は，多変量データの分析手法の一つである．この手法は，データセット内の変数間の相関関係を利用して，データの次元数を削減する手法である．この手法では，データの持つ情報をできる

だけ失わないようにしつつ，より少ない数の新しい変数である主成分にデータを再構成する．主成分は，元の変数の線形結合であり，データセットの分散が最大となる方向に相当している．

主成分分析の計算の方法としては，まず，データを標準化するために，各変数が平均 0、分散 1 となるようにする．これにより，異なる尺度の変数間の比較が可能になる．次に，標準化されたデータに基づいて，変数間の相関関係を表している共分散行列を計算する．それから，共分散行列の固有値と固有ベクトルを計算する．このとき，固有ベクトルは新しい軸である主成分の方向を示し，固有値はその軸に沿ったデータの分散の大きさを示している．これにより，固有値が大きい順に主成分を選択する．最初の数個の主成分は，データセットの大部分の分散を捉えることができる．最後に，元のデータを主成分上に射影することで，データを変換する．この変換により，データセットはより少ない数の次元で表現が可能となる．

主成分分析のメリットとしては，次元を削減することができ，高次元のデータを低次元として扱うことができることが挙げられる．そして，主成分によってデータを表現することで，高次元データの構造を視覚的に捉えることができる．また，低次元にすることで，データからノイズや不要な情報を除去することができ，より重要な特徴を強調することができる．

主成分分析のデメリットとしては，用いれるデータが線形の関係に限定されることが挙げられる．主成分分析は，変数間の線形関係に基づくため，非線形の関係を捉えることができない．それから，主成分分析により新しく形成される主成分は，元の変数の線形結合であるため，解釈が難しくなる場合がある．

2.14　クラスタリング

クラスタリングとは，機械学習の一種であり，データ間の類似度に基づいてデータをグループ分けする手法である．クラスタリングでは，データをクラスタとして分割することで，それぞれのクラスタが内部的には似た特性を持ちながら，他のクラスタとは異なる特性を持つようにする．このそれぞれのクラスタの違いを見ることで，データの内在的な構造を明らかにし，データの理解を深めることに繋がる．

適用例としては，市場調査が挙げられる．この場合，消費者をクラスタリングす

ることで，消費者の傾向に合わせたマーケティング戦略を策定する際に使用される．クラスタリング手法にはいくつかの種類がある．大きく 2 種類に分けられ，非階層型クラスタリングと階層型クラスタリングがある．前者の代表には,K-means 法がある．

2.15 K-means 法

K-means 法とは，クラスタリングの手法の一つであり，データを K 個のクラスタに分けることを目的としている．この手法では，データセット中のデータを，それが属するクラスタの中心に基づいて分けられる．

この手法では，まずクラスタ数 K を決定することが必要となる．この K は，事前にデータを分析する，または目的に応じて定めるかにより設定する．それから，K 個のデータを選びこれらの点を初期のクラスタ中心とする．この際の選び方はランダムであることが多い．ランダム以外にも，より良いクラスタリング結果を得るため，K-means++ の初期化のような，より洗練された方法を使用することもある．

次に，各データを最も近い距離のクラスタ中心に割り当てる．このときの距離とは，使用者が定めた距離尺度を用いる．距離尺度の例としては，ユークリッド距離が挙げられる．このような距離尺度により距離を測ることで，各データの最も近い距離のクラスタ中心を見つける．

それから，すべてのデータがクラスタに割り当てられれば，各クラスタの重心を計算し，それを新たなクラスタ中心として更新する．これは，クラスタ内の全データの各次元における平均値を取ることで行われる．この新しい中心は，そのクラスタのデータポイントが形成する空間的な中心に位置する．

最後にこれらのプロセスを，クラスタの割り当てが変わらなくなるか，または反復回数といった事前に定めた基準を満たすまで続ける．反復ごとに，各データのクラスタの割り当てとクラスタ中心の位置は更新される．その結果，データは K 個のクラスタに分割され，各クラスタはその中心に最も近いデータ点の集合として表される．

k-meas 法のメリットとしては，実装がしやすく，また可視化によりクラスタの分かれ方が直感的にわかりやすい点が挙げられる．また，計算コストが低いため，

大規模なデータセットにもも高速で動作できる．そして，多様なデータセットに対して適用可能であるため，幅広い分野で利用できる．

K-means 法のデメリットとしては，クラスタ数を事前に決める必要性が挙げられる．K-means 法では，クラスタ数を事前に設定する必要があるため，最適なクラスタ数の選択が難しい場合が考えられる．また，初期値に影響を受けやすく，ランダムによって決定された初期の中心により，最終的なクラスタリング結果が大きく変わることが考えられる．それから，この手法ではクラスタが凸形でかつ大体同じ大きさであると仮定している．そのため，異なる形状やサイズのクラスタを持つデータセットには適していない．

2.16 凝縮度

凝縮度は，クラスタリングの効果を測定する指標の一つであるシルエット係数 [7] にて用いられる値である．シルエット係数は，クラスタリングがどの程度うまく行われているかを，定量的に評価する．シルエット係数は，各データポイントについて計算され，その平均値でクラスタリング全体の評価を行う．シルエット係数は, -1 から 1 の範囲の値を取る．1 に近い値は良いクラスタリングを, 0 に近い値は重なり合うクラスタリングを，-1 に近い値は誤ったクラスタリングを示している．

凝集度は，あるデータポイントが属するクラスタ内の他のデータポイントとの平均距離を測定する．凝集度はクラスタ内のデータポイント間の密接さを示し，値が小さいほどそのデータは自身が属するクラスタによりよく凝縮していると評価される．あるデータ i に対する凝集度は，以下の式で計算される．

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i, j)$$

ここで C_i はデータ i が属するクラスタ，$d(i, j)$ はデータ i とデータ j の間の距離，$|C_i|$ はクラスタ C_i 内のデータポイントの数を表す．

凝集度は，クラスタ内の一貫性や密接さを評価するために用いる．凝集度が低いほど，各データは自分が属するクラスタ内の他のポイントに近く，良いクラスタリングの傾向と言える．一方，凝集度が高い場合，各データは自身のクラスタ内で孤立している可能性があるため，クラスタリングの質を再評価する必要がある．

凝集度を乖離度と組み合わせることで計算されるシルエット係数は，クラスタリング結果の解釈と評価に有用な指標である．クラスタリングのパフォーマンスを定量的に理解，異なるクラスタリング設定や手法を比較する際に役立つ．

2.17　乖離度

凝縮度は，クラスタリングの効果を測定する指標の一つであるシルエット係数にて用いられる値である．乖離度はシルエット係数において，ある各データが他のクラスタとどれだけ離れているかを示す指標である．具体的には，あるデータが属するクラスタ以外の最も近いクラスタまでの平均距離を測定する．乖離度は，そのデータが他のクラスタに属していた場合にどれだけ乖離しているかを，他クラスタとの分離度として示す．あるデータ i に対する乖離度は，以下の式で計算される．

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i, j)$$

ここで，C_k はデータ i が属していないクラスタ，$d(i, j)$ はデータ i とクラスタ C_k 内のデータポイント j との間の距離，$|C_k|$ は，クラスタ C_k 内のデータの数を表している．乖離度は，データ i から最も近いクラスタまでの平均距離として定義される．そしてデータがその最も近いクラスタにどれだけ近いかを示している．

乖離度は、データが他のクラスタに対してどれだけ乖離しているかを評価するために用いる．乖離度が高い場合，データは他のクラスタから遠く離れていることを意味している．この場合，そのデータが属するクラスタの分離度が良いことを示す．一方，乖離度が低い場合，データは他のクラスタに近く，クラスタ間の区別が不明瞭になる可能性がある．

乖離度を凝縮度と組み合わせることで計算されるシルエット係数は，クラスタリング結果の解釈と評価に有用な指標である．乖離度は，特にクラスタ間の分離度を評価する上で重要な役割を果たす．クラスタリングの結果，データが適切なクラスタに分類されているかを理解するのに役立つ

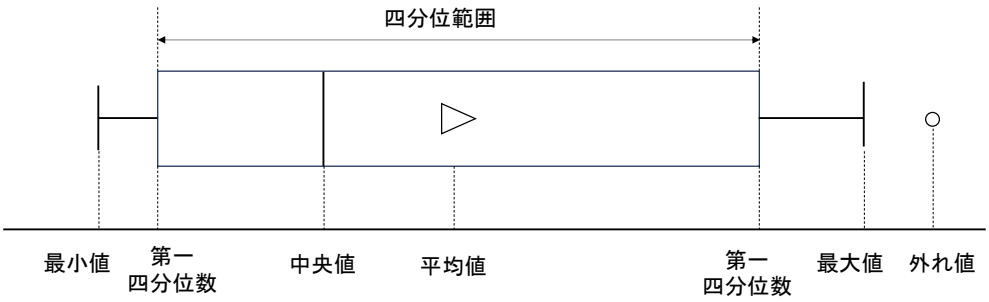

図 2.1　箱ひげ図の例

2.18　箱ひげ図

箱ひげ図とは，データの分布を視覚化するためのグラフの一つである．この図は，図 2.1 のように複数のデータの中央値，四分位数，外れ値といった情報を簡潔に示すことができ，データのばらつきや傾向を知ることができる．箱ひげ図は，小規模なデータの要約統計を表示するのに適しているため，カテゴリ間のデータを比較する際にも用いられる．

箱ひげ図の箱の部分は，データセットの下位四分位数から，上位四分位数までの範囲を表す．この範囲を，四分位範囲といい，データの半数が含まれている．箱の中央に描かれる線は，データセットの中央値を示す．中央値は，複数のデータを二等分する値である．箱の中央に位置することで，データの中心傾向を視覚的に捉えることができる．箱の上下に伸びるているひげの部分は，通常，上位と下位の外れ値の閾値までを示す．これらのひげは，データの最小値と最大値を示すことが多い．また，外れ値を除外した範囲を示す場合もある．外れ値は，ひげの外に個別の点としてプロットされる．この値は，複数のデータに含まれる異常値やを識別するのに役立つ．

箱ひげ図の利点は，データの分布に情報を簡潔に可視化できる点にある．可視化することにより，データのばらつきの大きさ，中央値の傾向，外れ値の有無について素早く把握することができる．例えば，複数のグループ間でデータを比較する際には，それぞれの箱ひげ図を並べて表示することで，グループ間の違いや類似性を把握しやすくなる．

2.19　距離関数

距離関数とは，二つのデータ間の距離を測定するために使用される関数である．この関数は，解析対象のデータの性質や適用されるアルゴリズムによって異なる．そして，データ間の類似性などを定量的に評価するために必要となる．距離関数として適切なものを選択することで，機械学習モデルの性能と精度を大幅に向上させることができる．距離関数の例には，ユークリッド距離やマンハッタン距離，チェビシェフ距離，コサイン類似度が挙げられる．

2.19.1　ユークリッド距離

ユークリッド距離は，最も基本的な距離の尺度の一つで、平面上または空間内の 2 点間の直線距離を測定する．2 点間のユークリッド距離は，ピタゴラスの定理を用いて計算することができる．2 点の座標が与えられた場合，それらの点を結ぶ最短経路の長さを求めるために使用される．2 次元空間のユークリッド距離では，2 次元の平面上で、点 $A(x_1, y_1)$ と点 $B(x_2, y_2)$ 間のユークリッド距離は次の式で計算される．

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

3 次元空間でのユークリッド距離では，3 次元空間内での 2 点間のユークリッド距離は、以下の式により求められる．

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

n 次元空間でのユークリッド距離では，n 次元空間内での 2 点 A と B のユークリッド距離は、より一般化された形で以下のように表される．

$$d = \sqrt{\sum_{i=1}^{n} (B_i - A_i)^2}$$

2.19.2　マンハッタン距離

マンハッタン距離（Manhattan Distance) とは，基本的な距離の尺度の一つで二点間の距離を格子状の道路に沿って測る手法である．この名称は，ニューヨークのマンハッタンの都市計画に由来し，道路が直角に交差するグリッド状のパターンを指す．2 点間のマンハッタン距離は，各座標軸に沿った距離の絶対値の合計で表される．式としては以下のように表される．

$$d = \sum_{i=1}^{n} |p_i - q_i|$$

ここで，p と q は二つの点を表し，n は次元数である．マンハッタン距離は，都市のブロックを移動する際のような直線的ではないルートで測る場合に適している．また，計算が単純であるため計算コストが低いことから，ユークリッド距離よりも計算がしやすい．

2.19.3　チェビシェフ距離

チェビシェフ距離（Chebyshev Distance) とは，基本的な距離の尺度の一つであり，2 点間の距離をその点が異なる座標軸に沿って持つ差の最大値として定義される．チェスでの王の動きに例えられることが多く，王が移動するために必要な最小の手数に相当する．

$$d = \max_{i} |p_i - q_i|$$

ここで，p と q は二つの点を表し、i は座標軸のインデックスである．チェビシェフ距離は，各方向に自由に移動できる場合の最適な移動距離を反映しているため，特にゲーム理論や最適化問題といった特定の種類の問題設定において有用である．

2.19.4　コサイン類似度

コサイン類似度は，距離尺度というより二つのベクトル間の類似性を測定する指標である．この尺度は，ベクトルの向きの類似性に基づいており，その大きさは考

慮されない．ベクトルが指す方向が同じであれば，コサイン類似度は 1 となる．逆に，完全に反対の方向を指していれば-1 となる．

$$\text{コサイン類似度} = \frac{\mathbf{p} \cdot \mathbf{q}}{\|\mathbf{p}\|\|\mathbf{q}\|}$$

ここで ($\mathbf{p} \cdot \mathbf{q}$ は二つのベクトルのドット積であり，$\|\mathbf{p}\|$ と $\|\mathbf{q}\|$ はそれぞれのベクトルの大きさである．コサイン類似度は，特にテキストデータの処理や情報検索において広く用いられる．ベクトルがテキストドキュメントを表す場合，この尺度はテキスト間の内容の類似性を示すのに適している．

第 3 章　関連研究

本章では，本研究の関連研究を紹介する．関連研究では，大きく 2 つのトピックに触れる．

3.1　BERT の言語理解に関する研究

BERT の各層出力に関する研究については，これまで多くの研究が行われている．Jawahar ら [4] は，BERT が言語の構造に関してどのような情報を捉えるかを解明することを目的とした研究を行った．そこで，BERT の各層から得られる表現を分析し，表面的な特徴、構文的特徴、意味的特徴がどのようにエンコードされているかを調査した．表面的な特徴とは，文の長さや文中の単語の存在といった情報を指す．構文的な特徴とは，単語の順序に対する感度や構文木の深さといった情報を指す．意味的な特徴とは，時制や主節の主語，名詞/動詞のランダムな置換への感度を指す．調査の結果，BERT が言語の構造に関して豊かな階層的情報を捉えていることを示した．具体的には，BERT の下層では表層的な特徴が，中層では構文的特徴が，上層では意味的特徴がエンコードされていることが判明した．また，長距離の依存関係情報をモデル化するためには，より深い層が必要であることを示した．

Coenen ら [8] は BERT というモデルが自然言語処理においてどのように言語情報を内部的に表現しているかを調査した．BERT ベースモデルを使用し，各文中のトークンペア間のモデル全体の attention ベクトルを取得した．これらのラベル付き埋め込みを使用して，2 つの L2 正則化された線形分類器を訓練した．一つは、2 つのトークン間に依存関係が存在するかどうかを予測する二値分類器である．もう一方は，依存関係の存在が与えられた場合にその依存関係のタイプを予測する多クラス分類器である．実験の結果，二値分類器は 85.8 %の精度を，多クラス分類器は 71.9 %の精度を達成しました．これらから，彼らは BERT のモデル全体の注意ベクトルが構文的特徴を比較的単純に表現していることを示した．

白静らは [9]，BERT の下位階層の単語埋め込み表現列を利用した，感情分析の教師なし領域適応に関する研究を行った．この研究では，BERT の各層から出力される単語埋め込み表現列を利用し，特に下位層の単語埋め込み表現列が領域適応に

おいて最上位層よりも有効である可能性を検証した．BERT は通常，最上位層の単語埋め込み表現列が各タスクで利用されるが，領域適応の文脈ではこれが必ずしも最適とは限らないため，より下位の階層を利用することを提案した．実験の結果，全体の半数で標準手法よりも高い正解率を達成したが，全体の平均では標準手法にわずかに劣る結果となった．これにより領域適応においては最上位層が必ずしも最良でないため，下位層の情報を併用する手法の可能性が示唆された．

これらの研究では，BERT の言語理解に関する研究を行っている．われわれは，これらの研究で示された BERT の言語理解に関する知見に基づき，ラベルの一致・不一致の言語の特徴の違いを明らかにする．

3.2 評価者間でのラベルの曖昧性に関する研究

Venanzi ら [10] は，クラウドソーシングデータセットから，正確なラベルを抽出するための研究を行った．従来の，個々の作業者の信頼性をモデリングすることに焦点を当てると，作業者あたりのラベルが少ない場合効果がない．そのため，コミュニティベースのベイジアンラベル集約モデル CommunityBCC を提案した．このモデルは，クラウドワーカーがいくつかの異なるタイプに従い，各タイプが類似した混乱行列を持つ作業者のグループを表すと仮定する．このモデルでは，各コミュニティの混乱行列，各ユーザーのコミュニティメンバーシップ、各アイテムの集約ラベルを学習することができる．実験結果は，CommunityBCC モデルが，既存手法よりも一貫して高い精度を達成していることを示した．

Chang ら [2] はクラウドソーシングによる機械学習データセットのラベル付けに曖昧性の研究を行った．従来のラベルの品質向上手法は，曖昧なラベル付けガイドラインにより，概念の解釈の相違やラベルの不一致が発生する問題があった．この問題に対処するため，Revolt は専門家のアノテーションの流れのアイデアをクラウドベースのラベリングに導入する共同作業手法を導入した．Revolt は，クラウドの意見の相違を利用してあいまいな概念を特定し，事後的なラベル決定のための意味に関連するグループを作成する．実験では，Revolt を従来のラベリング手法と比較し，ラベルに関するガイドラインを必要とせずに高品質のラベルを生成できることを示した．

これらの研究では，クラウドソーシングにより収集したデータのおいて，ラベル付けの曖昧性に着目している．そして，その解決のために正確なラベルの提案をしている．そのため，異なるラベル与えられているデータにおける特徴の違いという観点で類似している．われわれは，BERT の各層に着目することで，ラベルの一致・不一致の言語の特徴の違いを明らかにすることを目指している．

第 4 章　提案手法

本研究では，クラウドソーシングにり収集されたデータにおける評価者間でのラベルの一致・不一致を判別するために，BERT の各層の出力を比較することで判別を行う．ここで目指す判別とは，一定数投票した結果ラベルが不一致であるデータに対して，追加のラベル付け後にラベルが一致するのかどうかを判別することを指す．

4.1　ラベルの一致・不一致の定義

本研究におけるラベルの一致・不一致を定義する．ラベルの一致とは，多数決によりラベルが一致に定められる場合を指す．n 種類のラベルへの投票における各ラベル得票数 v について，ラベルの一致の式を以下のように示す．

$$v_i > v_j \quad (\exists i \in \{1, ..., n\}, \forall j \in \{1, ..., n\}, j \neq i)$$

ここで，i は最も票を得たラベル得票数のインデックスを，j はその他のラベル得票数のインデックスを示している．これは投票の結果どれか一つのラベル得票数 v_i が，他のそれぞれのラベル得票数 v_j よりも多くの票を得ている状況を表している．よって，最も多くの票を得たラベル得票数が，他のどのラベル得票数よりも票数が多い場合．一致とする．

例えば，ラベルの一致の例を図 4.1 に示す．テキストがどのような感情ラベルを持つのかについて，ポジティブ，ネガティブ，ニュートラルの 3 種類からの選択肢から 5 人の評価者によりラベル付けを行う．このとき，ポジティブに 3 票，ネガティブに 3 票，ニュートラルに 1 票のとき入ったとする．この場合，一番票数の多いラベルはポジティブである．そのため，ポジティブラベルに一意に定めることができ，ラベルは一致となる．

一方，ラベルの不一致とは多数決によりラベルが一意に定められない場合を指す．n 種類のラベルへの投票における各ラベル得票数 v について，ラベルの不一致の式を以下にように示す．

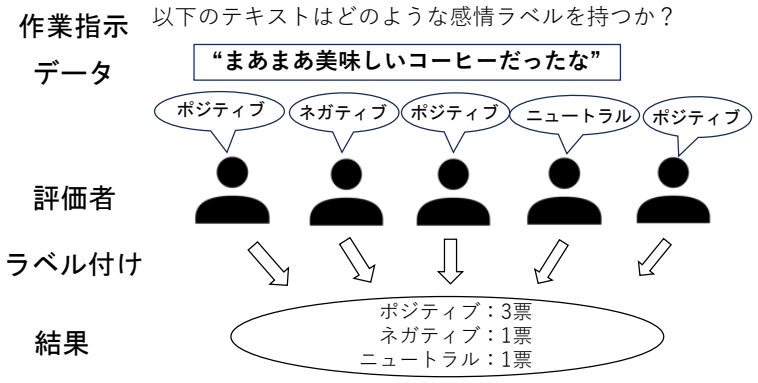

図 4.1　ラベルの一致の図

$$neg(v_i > v_j) \quad (\exists i \in \{1, ..., n\},\ \forall\ j \in \{1, ..., n\}, j \neq i)$$

これは，一致の条件の否定である．つまり，投票の結果どれか一つのラベル得票数 v_i が、他のそれぞれのラベル得票数 v_j よりも明確に多い票数を得ていない場合，不一致となる．この状況では，最も多くの票を得たラベル得票数は，他のラベル得票数と同数までの票数しか得ることができない．

例えば，あるテキストがどのような感情ラベルを持つのかについて，3 種類の選択肢から 5 人の評価者によりラベル付けを行う．このとき，ポジティブに 1 票，ネガティブに 2 票，ニュートラルに 2 票のとき入ったとする．この場合，最も票数の多いラベルはネガティブとニュートラルである．そのため，多数決によりラベルを一意に定めることができずラベルは不一致となる．

これらのラベルの一致・不一致の定義を踏まえ，以降では追加のラベル付けの前に不一致であるデータを追加前データと呼称する．また，追加のラベル付けの後にラベルの一致・不一致であるデータは，そのままラベル一致データとラベル不一致データと呼ぶ

4.2　BERT の各層の出力

BERT からの出力は，各入力テキストに対する各層の [CLS] トークンの特徴ベクトルを利用する．イメージとしては，図 4.2 として表す．例えば一層の出力は，

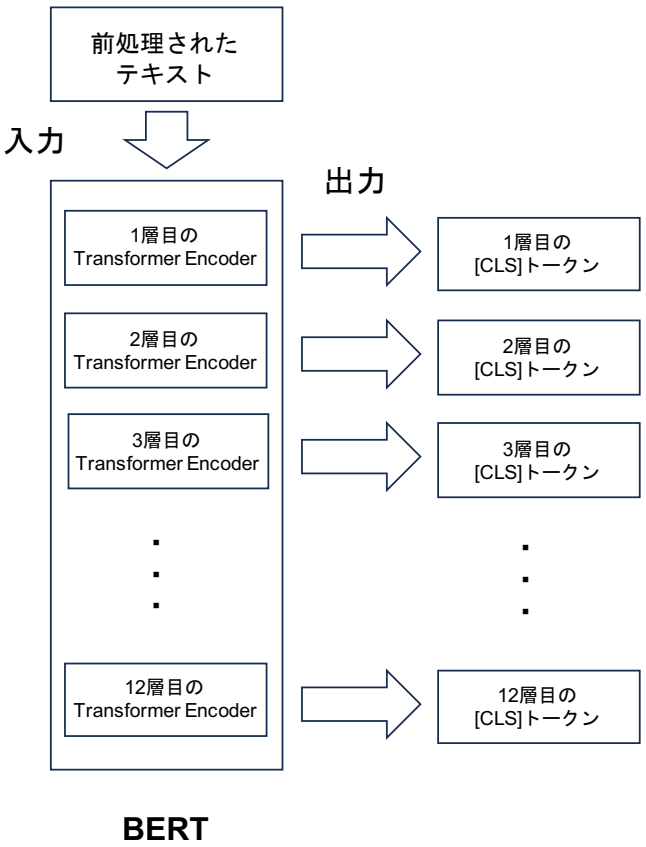

図 4.2　BERT の各層の出力

前処理，およびトークン化したテキストを入力，BERT に入力することで，1 層目の Transformer Encoder にて計算された 1 層目の [CLS] トークンの値出力することで得ることができる．[CLS] トークンのベクトルは，入力されたテキスト全体の意味を総合的に表現している．評価者間でラベルが一致または不一致となるテキストの理解において，単語や文章の表現の違いを捉えるため，BERT の各層ごとの出力に着目した．

提案手法では，BERT モデルの各層が捉える言語の特徴の違いを活用する．この層毎の言語特徴の違いにより，BERT は評価者間でのラベルの一致・不一致の背後にある，言語的特徴の違いを明らかにすることができる．よってわれわれは，この層別の言語特徴を，ラベルの一致・不一致を識別する際の指標として活用することを考えた．これにより，文章における単語選択や構文の違いによって生じる文の意

味の差異が現れると考えた．そして BERT の各層が示す言語的特徴を用いることで，ラベルの一致・不一致のデータの違いを比較する．

4.3　ラベルの一致・不一致の判別手法

提案手法では，クラスタを作成する段階とラベルの一致・不一致を判別する 2 つの段階に分けられる．

4.3.1　クラスタの作成

クラスタを作成する段階では，ラベルの一致のクラスタと不一致のクラスタについて，それぞれのクラスタの作成を行う．

まず，ラベルの一致・不一致のテキストデータを前処理したものを BERT に入力することで，BERT の 1 層から 12 層までの各層の出力を獲得する．そして，各層の出力を 1 層ずつ，K-means 法を用いることで 2 つのクラスタを作成する．この 2 つのクラスタのうち，どちらがラベルの一致または不一致のクラスタであるかを判別するため，各クラスタにおけるラベルの一致・不一致のデータ数を確認する．各クラスタにおいて，ラベルの一致と不一致において，どちらのデータ数が多いかを確認することにより，教師あり学習としてクラスタの識別を行う．

この結果，ラベルの一致のデータ数が多いクラスタはラベルの一致クラスタ，ラベルの不一致のデータ数が多いクラスタはラベルの不一致クラスタと識別される．クラスタを識別するための一連の流れを，1 層から 12 層までそれぞれの層で行う．

4.3.2　ラベルの一致・不一致の判別

2 つ目の段階では，ラベルの一致・不一致の判別を行う．ここでは，1 層から 12 層までの BERT の出力を用いて作成した 2 つのクラスタについて，比較的クラスタの分離が大きい BERT の層の出力を用いる．

まず，ラベルの一致・不一致のどちらかを判別したいテキストを用意する．用意したテキストを，プロットした 2 つのクラスタと同じように，BERT の層の出力を

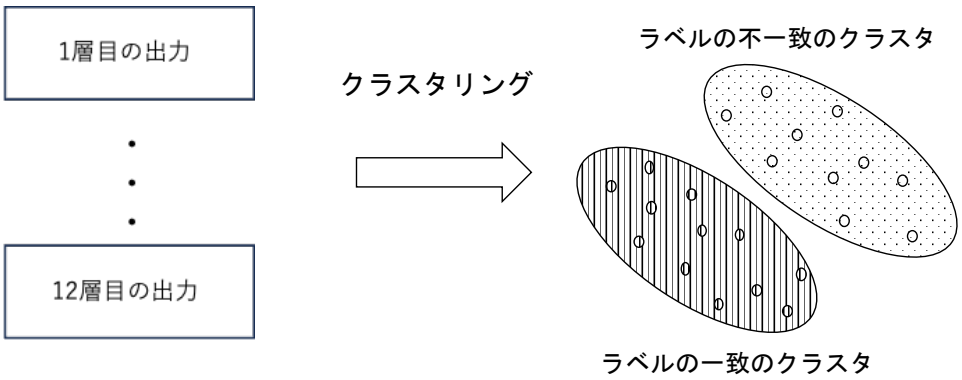

図 4.3　クラスタの作成の図

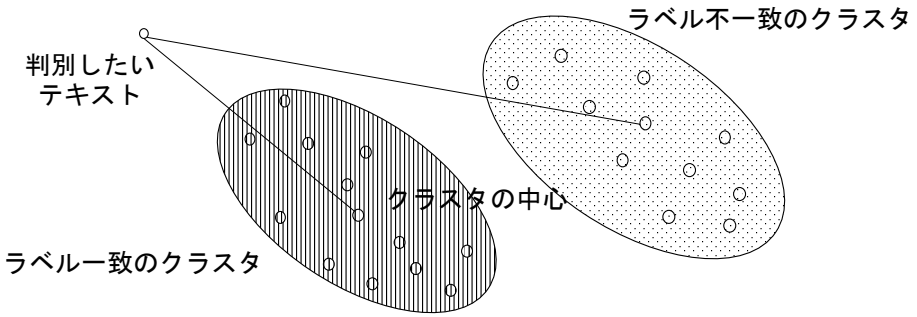

図 4.4　ラベルの一致・不一致の判別の図

用いてプロットを行う．この判別したいテキストは，クラスタを作成するために用いた BERT の同じ層の出力を用いることで，プロットを行う．それから，判別したいテキストデータのプロットについて，ラベルの一致のクラスタとラベルの不一致のクラスタそれぞれの，クラスタ中心とのユークリッド距離を求める．このとき，判別したいテキストデータのプロットが，どちらのクラスタ中心と距離が短いかによりラベルを判別する．

判別したいテキストごとに，このラベルの一致・不一致の判別の一連の流れを行う．

第 5 章　評価実験

本研究では，クラウドソーシングにより収集されたデータにおける評価者間でのラベルの一致・不一致を判別する手法の提案した．提案手法が有効であるかを確かめるために，BERT の層を比較を比較するための実験を行った．

5.1　使用データセット

本研究では，クラウドソーシングによって作成された岐阜大学鈴木研究室のデータセットを使用する．このデータセットは 605 人のクラウドワーカーにより，ラベル付けされたデータで作成されている．このデータはワーカ ID，ツイート ID，ツイート内容，評価ラベルの 4 つのカラムで作成されている．このデータセットが作成された目的は，ツイート内に含まれる「笑」がどのような感情を持つのかを知ることにある．着目している感情は 6 種類あり，ポジティブ，ネガティブ，ニュートラル，ポジティブ＋ネガティブ，その他，判断できないのそれぞれのラベルが存在する．

われわれはこのデータセットにおいて，ポジティブ，ネガティブ，ニュートラル，ポジティブ＋ネガティブの 4 種類のラベルのいずれかに投票されているものに着目した．このとき，その他，判断できないの 2 つのラベルに関しては，1 票でも投票されている場合，利用しないこととした．この 2 つのラベルは，クラウドワーカーが判断が困難だった場合に付与されるラベルである．一方，ポジティブ，ネガティブ，ニュートラル，ポジティブ＋ネガティブの 4 種類のラベルは，確信を持っているラベルと考えられる．われわれが考えている評価者間のラベルの不一致とは，評価者が確信を持って着目している 4 種のラベルに入れた結果，ラベルが不一致となるときである．そのため，5 票のうち 1 票でも判断が困難であるラベルに投票された場合，確信はなく曖昧と判断でき，想定と異なるため排除した．

データセットにおける投票数毎のデータ数を表 5.1 に示す．この表はデータセットの合計 33020 件のデータが，それぞれ投票の合計で何票を与えられているのかを投票数として表している．表から 5 票以下や 11 票以上の投票数のデータは少ない一方，5 票から 10 票のデータは多い．そして，多くの票は 5 票から 10 票に集中し

ている．中でも特に 5 票のデータは突出して多くなっており，26848 件存在した．データ数が多い場合，より多くのデータの特徴を考慮できると考えられる．そのため，実験ではこの最もデータ数の多い投票数が 5 票であるデータを用いる．

表 5.1　データセットにおける投票数毎のデータ数

投票数	1 票〜4 票	5 票	6 票	7 票
件数	275 件	26848 件	2900 件	1097 件
投票数	8 票	9 票	10 票	11 票〜24 票
件数	666 件	466 件	289 件	479 件

5.2　実験設定

追加投票に応じたラベルの一致・不一致の遷移を図 5.1 のように示す．各データが獲得した投票数は，n 票を獲得した段階と $n+\alpha$ 票を獲得した段階の 2 つの段階がある．本研究では，最初に n 票でラベルが不一致だった追加前データに着目する．このとき，追加の α 票を含めた $n+\alpha$ 票の結果においても不一致のままである場合と，一致に遷移する場合を判別することを目指している．この実験では，最初の投票の段階で各データが獲得していた総票数を 3 票とした．そして，次の時点において追加投票後に各データが獲得している総票数を 5 票とした．

このとき，各データが獲得している総票数の 5 票は使用データセットからそのままのデータ数を用いた．一方，3 票はデータをそのまま用いることができない．これは，総票数を 5 票獲得している各データにおいて，3 票時点ではどのラベルに投票されているかがわからないためである．そこで，われわれは 3 票時点でどのラベルに 3 票分が投票されているのかを，ランダムとして設定した．クラウドソーシングにより評価者がデータにラベルを付与する際に，最初の 3 票でどのラベルに投票されるかはランダムと言えると考えたためである．

総票数が 5 票の場合から，ランダムでラベルを付与した総票数が 3 票の場合でのラベル得票数の比率毎のデータ数を示したものが表 5.2 である．ラベル得票数比率とは，今回使用している 4 種類のラベルの内ラベルの種類毎のラベル得票数の比率

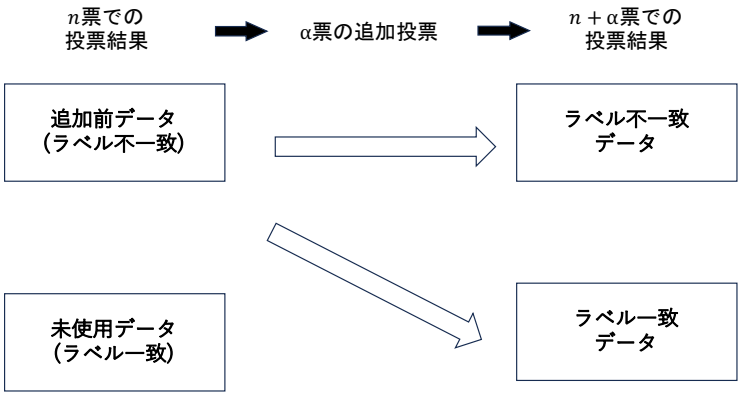

図 5.1　ラベル一致・不一致の状態の遷移図

を表している．これにより，ラベル得票数比率は 3 票すべてが 1 種類のラベルに付与された場合に 3:0:0 となる．一方，3 票すべてが異なるラベルに投票した際にはラベル得票数比率は 1:1:1 となる．

また，このラベル得票数比率に対して評価者間でのラベルの一致・不一致を判断する．本研究における評価者間の一致・不一致の式は 4 章にて示している．この式を適応することで，3 票時点の場合にはラベル一致とはラベル得票数比率が，3:0:0 または 2:1:0 であるときを示す．そして，ラベル不一致とはラベル得票数比率が 1:1:1 であるときのみを示す．その結果，3 票時点でのラベルの一致・不一致それぞれのデータ数は表 5.3 をとして表される．

同様に，総票数が 5 票の場合におけるラベル得票数の比率毎のデータ数を示したものが表 5.4 である．この場合における，ラベル得票数比率対する評価者間でのラベルの一致・不一致を判断する．ラベル一致とは，ラベル得票数比率が，2:1:1:1 または 3:1:1 であるときを示す．そして，ラベル不一致とはラベル得票数比率が 2:2:1 であるときのみを示す．また，ラベル得票比率が 1:1:1:1:1 の場合は 4 種類のラベルに投票する想定上あり得ないため省く．

最初の 3 票でラベルが不一致だった追加前データが，追加の 2 票を含めた 5 票の結果不一致データに遷移する場合と，一致データに遷移する場合に着目をする．

表 5.2　3 票時点でのラベルの得票数比率毎のデータ数

ラベル得票数比率	得票数の件数
3:0:0	8596 件
2:1:0	14941 件
1:1:1	3311 件

表 5.3　3 票時点でのラベル一致・不一致毎のデータ数

ラベル一致・不一致	件数
ラベル一致	23537 件
ラベル不一致	3311 件

表 5.4　5 票時点でのラベルの得票数比率毎のデータ数

ラベル得票数比率	得票数の件数
3:1:1	1189 件
2:1:1:1	732 件
2:2:1	1390 件

表 5.5　5 票時点でのラベル一致・不一致毎のデータ数

ラベル一致・不一致	件数
ラベル一致	1921 件
ラベル不一致	1390 件

5.3　評価指標

本研究の評価指標としては，シルエット係数に付随する凝縮度，乖離度を用いた．一方，シルエット係数は用いなかった．シルエット係数は，クラスタ内は密に凝集されているほど良い．そして異なるクラスタは遠く離れているほど良いという考え

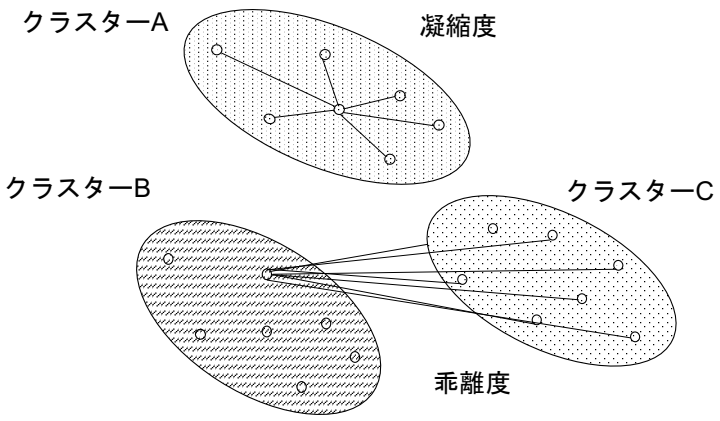

図 5.2　凝縮度と乖離度の図

方に基づいている．この計算式によって最も評価されるのは，2 つのクラスタが完全に距離を空けて分離している，かつ 2 つのクラスタがそれぞれ凝縮した分布を形成している場合である．この場合は，もし 2 つのクラスタのうち片方のクラスタが丸く凝視しており，一方のクラスタがその周りに分布していた場合，うまく分離を評価できない．そのため，本研究では乖離度と分離度の 2 つを用いて評価を行った．凝縮度と乖離度においては，それぞれ図 5.2 のように考えられる．また k-meas 法における，ラベルの一致クラスタとラベルの不一致クラスタの評価には，正解率を定義した．正解率とは各クラスタ中のデータ数のうち，ラベルの一致データと不一致のデータのそれぞれのデータ数の割合を求め，その内の大きい値と定義する．この指標により，ラベルの一致・不一致がどの程度正しくクラスタリングされるかを評価する．

それぞれの具体的な定義は 2.16 節，2.17 節に示した．

5.4　実験内容

実験では，5 票時点で評価者間でのラベルが一致，不一致のデータを用いる．これらは，データ数を揃えるために元々ラベルが一致であるデータを削除することで数を合わせた．このとき，数を合わせるために削除するデータはランダムにより決定した．これより，ラベルの一致・不一致のデータはそれぞれ 1,390 件を用いる．2

種類のデータ数の合計である，2,780 件を入力データとした．これら 2 種類のデータは，3 票時点ではどちらも不一致として追加前データであったものである．これらを用いて，各層の BERT の出力から得られる文章ベクトルから 2 種類のデータの比較を行う．

入力するテキストは，前処理およびトークン化されたテキストを BERT に入力する．BERT で扱えるトークンの最大長は 512 とし，バッチサイズは 32 とした．

また，実験では次元圧縮による次元数を 2 次元，3 次元，768 次元の三つのパターンで行う．この場合分けは，次元圧縮による情報の損失を考慮するため行う．

5.4.1 データの分析の手順

提案手法のステップは以下となる．

1. 追加投票後のテキストを，前処理してトークン化
2. トークン化したテキストを BERT に入力し，各層から出力を獲得
3. t-SNE により各層の出力を次元圧縮
4. K-means 法により 2 つのクラスタへクラスタリング
5. 作成したクラスタについてラベルの一致・不一致を識別
6. クラスタ間におけるデータの分離を評価し，各層にて比較

BERT に入力するテキストには，事前に前処理を行う．具体的には，カッコや引用符，URL，ピリオド，カンマ，改行，空白の記号の除去や全角文字を半角へ，大文字を小文字への変換を行った.．また数字はすべて 0 に統一した．テキストは，BERT に入力する前にトークン化する必要がある．本研究では，日本語のテキストを扱うため BertJapaneseTokenizer を利用した．BERT への入力とは，クラウドソーシングデータにおける個々のテキスト項目といった，評価者によってラベル付けされたテキストデータである．これらのテキストデータを，BERT が処理できる形式に適切に前処理し，トークン化したものを用いる．本研究では，自然言語処理の深層学習モデルである BERT を利用した．BERT の事前学習済みモデルは，東北大学により公開されている BERT-base モデルを利用した．そして，BERT の事前学習済みモデルについてファインチューニングはせずに，そのままのモデルの

出力を利用した．ラベルの識別では，2 つのクラスタにおいて 5 票が与えられた際のラベルの一致データとラベルの不一致データを用いた．ラベルの一致と不一致のデータのうち，どちらのデータ数が多いかを確認することにより，クラスタの識別を行う．これにより，2 つのクラスタはラベルの一致のクラスタとラベルの一致のクラスタとして識別される．

5.4.2　次元圧縮

BERT により出力された [CLS] トークンの特徴ベクトルは 768 次元あるため，そのまま構造の可視化のためにそのまま利用することは困難である．そのため，特徴ベクトルを低次元へ次元圧縮をする必要がある．本研究では，t-SNE を次元圧縮の手法として用いた．t-SNE の次元圧縮では局所的な構造の保持をすることができ，[CLS] トークンの高次元のベクトルに用いる上で適切と考えた．

5.4.3　クラスタリング

BERT の出力を 2 つのクラスタとするため，本研究ではクラスタリングの手法として K-means 法を用いた．K-means 法より次元圧縮した特徴ベクトルを，ラベル一致のクラスタとラベル不一致のクラスタの，2 つのクラスタへとクラスタリングをする．

5.5　実験 1:2 次元の出力での比較

表 5.6　2 次元での K-means 法による正解率

層	1	2	3	4	5	6
正解率	0.534	0.541	0.542	0.548	0.545	0.553
層	7	8	9	10	11	12
正解率	0.542	0.540	0.540	0.534	0.542	0.536

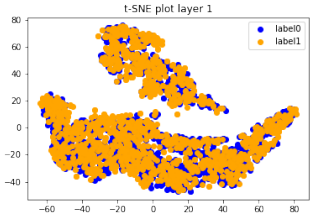

図 5.3　1 層目での出力

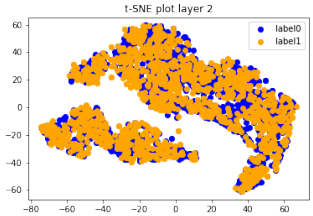

図 5.4　2 層目での出力

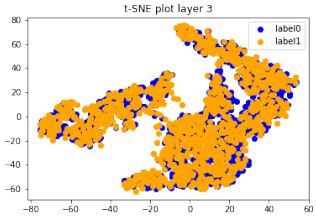

図 5.5　3 層目での出力

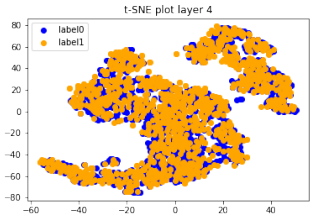

図 5.6　4 層目での出力

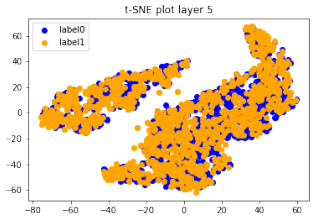

図 5.7　5 層目での出力

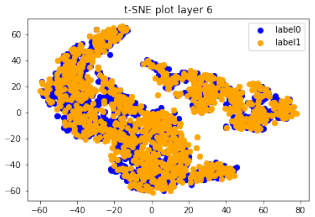

図 5.8　6 層目での出力

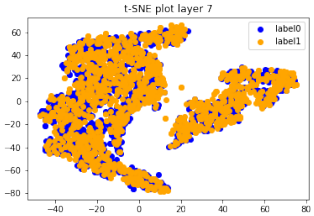

図 5.9　7 層目での出力

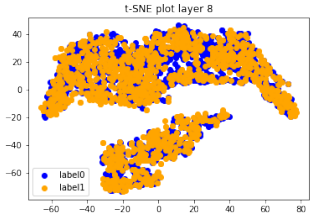

図 5.10　8 層目での出力

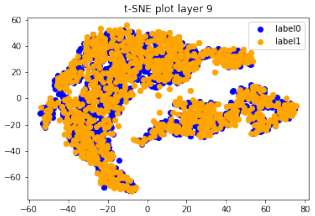

図 5.11　9 層目での出力

BERT の出力を t-SNE により，2 次元に圧縮した際の可視化の結果を図 5.3 から図 5.14 に示した．これらの図において，label0 が 5 票での評価者間のラベルが一致である点，label1 が 5 票での評価者間のラベルが不一致である点を表している．これらの可視化では，2 つの点は分布が重複していることが読み取れる．そのため，t-SNE による 2 次元への次元圧縮では，ラベルの一致・不一致は分離できなかった．

また，表 5.7 には 1 層から 12 層までの K-means 法によるクラスタリングの正解率を示した．各層の正解率は，いずれも約 0.54 であり層毎の比較において差はな

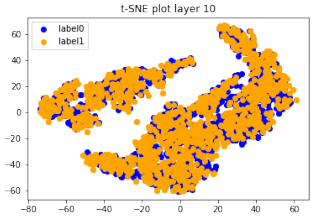

図 5.12　10 層目での出力

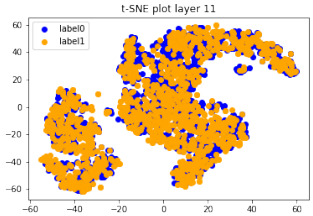

図 5.13　11 層目での出力

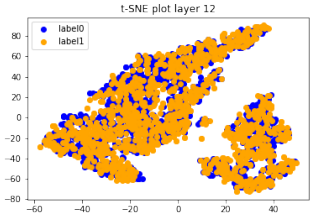

図 5.14　12 層目での出力

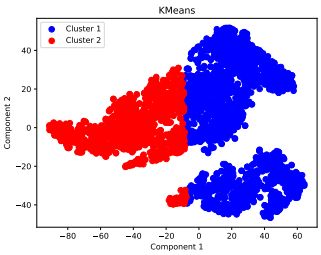

図 5.15　1 層目でのクラスタリング

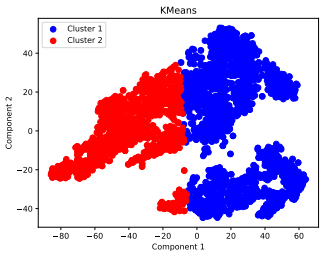

図 5.16　2 層目でのクラスタリング

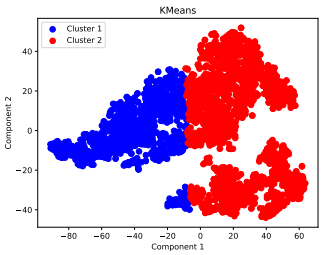

図 5.17　3 層目でのクラスタリング

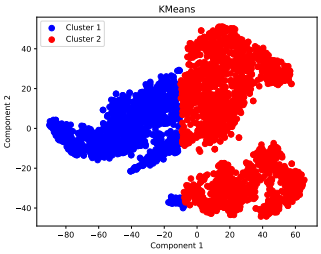

図 5.18　4 層目でのクラスタリング

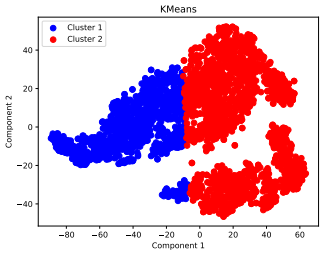

図 5.19　5 層目でのクラスタリング

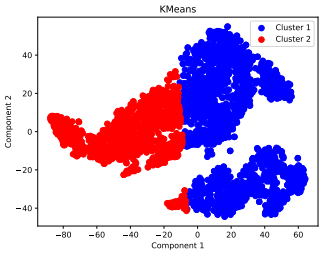

図 5.20　6 層目でのクラスタリング

い．また，これらは 2 つのクラスタでの正解率であるため，クラスタにはうまく分かれていないことが言える．このことは，図 5.15 から図 5.26 までの K-means 法によるプロットの結果からも読み取ることができる．

2 つのクラスタ間の凝縮度の結果を，図 5.27 に示した．それぞれの図で，横軸は BERT の 12 層の層番号，縦軸は凝縮度を示す．平均値や中央値に着目するとどの

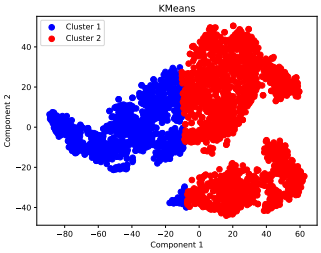

図 5.21　7 層目でのクラスタリング

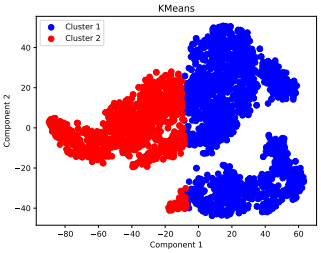

図 5.22　8 層目でのクラスタリング

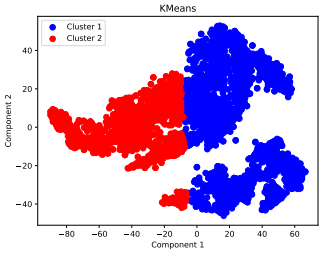

図 5.23　9 層目でのクラスタリング

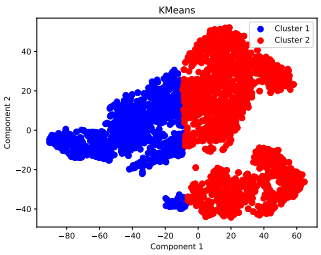

図 5.24　10 層目でのクラスタリング

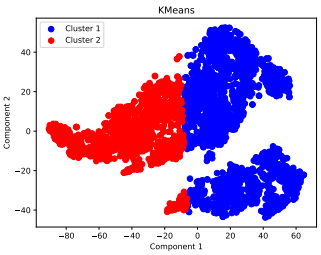

図 5.25　11 層目でのクラスタリング

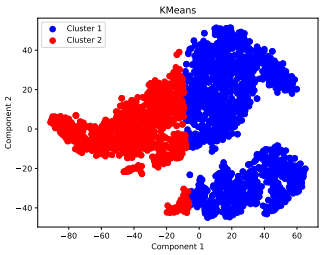

図 5.26　12 層目でのクラスタリング

層においても値は約 60 を取っており，差がないと言える．第一四分位数に着目するとどの層においても，値は約 50 を取っている．また，第三四分位数に着目するとどの層においても値は約 70 を取っている．四分位範囲については，各層約 18 の値を持っておりデータの散らばり方も類似していることが読み取れる．これらから実験の結果，2 つのクラスタにおける凝縮度には層毎による違いがないことがわかる．

また 2 つのクラスタ間の乖離度の結果を図 5.28 に示した．それぞれの図で，横軸は BERT の 12 層の層番号，縦軸は乖離度としている．平均値や中央値に着目するとどの層においても，値は約 100 を取っており，差がないことが言える．第一四分位数に着目するとどの層においても，値は約 83 を取っている．第三四分位数に着目するとどの層においても，値は約 118 を取っている．四分位範囲については，各層約 35 の値を持っていた．そのため，データの散らばり方は類似していることが読み取れる．これらから実験の結果，2 つのクラスタにおける乖離度には層毎に

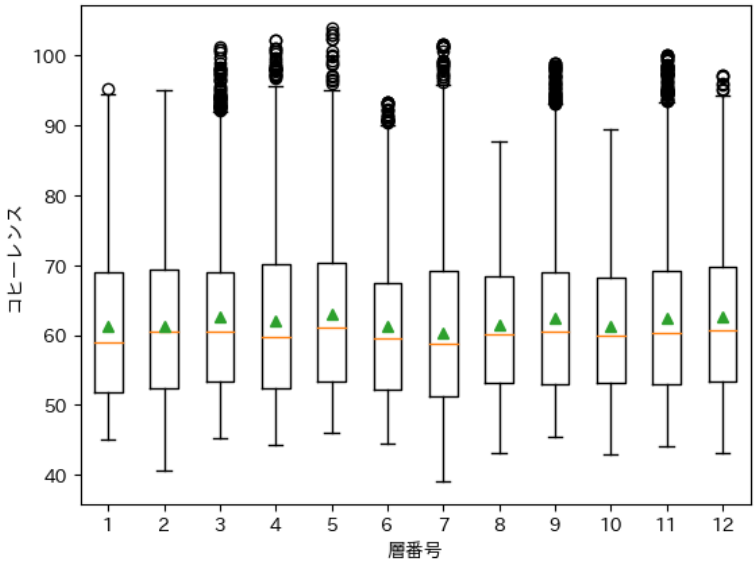

図 5.27　2 次元における凝縮度の箱ひげ図

よる違いがないことがわかる．

凝縮度と乖離度での各層の比較を行うことで，凝縮度が低く，乖離度が高い層があることを期待したが，層毎による違いが読み取れなかった．そのため，BERT の出力を 2 次元に次元圧縮した際の，ラベルの一致・不一致では，データの分離に違いがないと言える．

5.6　実験 2:3 次元の出力での比較

表 5.7　3 次元での K-means 法による正解率

層	1	2	3	4	5	6
正解率	0.536	0.538	0.540	0.536	0.537	0.538
層	7	8	9	10	11	12
正解率	0.547	0.541	0.538	0.540	0.542	0.549

BERT の出力を t-SNE により，3 次元に圧縮した際の可視化の結果を図 5.41 か

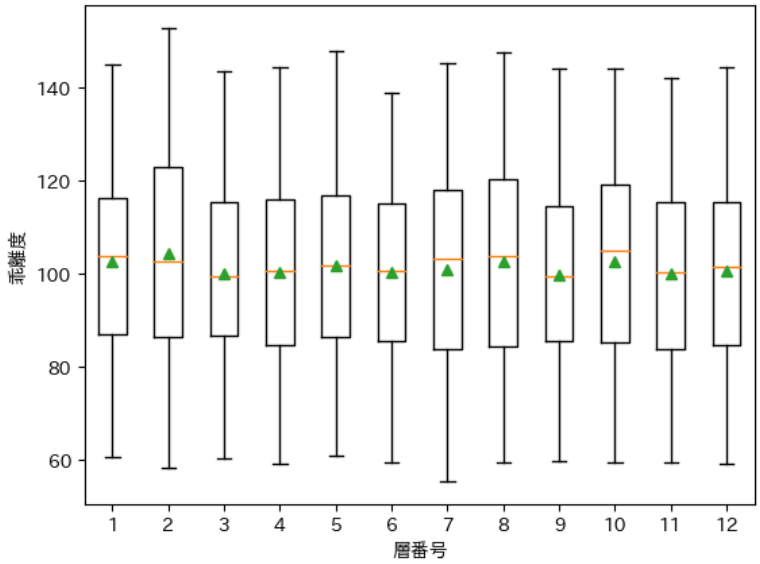

図 5.28　2 次元における乖離度の箱ひげ図

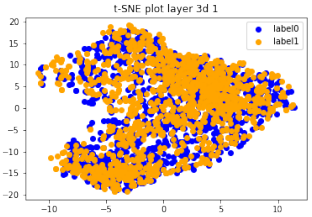

図 5.29　1 層目での出力

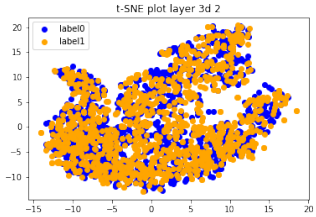

図 5.30　2 層目での出力

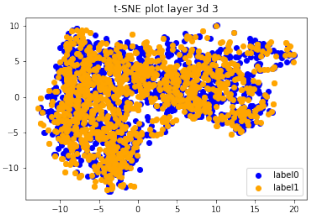

図 5.31　3 層目での出力

ら図 5.52 に示した．これらの図において，label0 が 5 票での評価者間のラベルが一致である点，label1 が 5 票での評価者間のラベルが不一致である点を表している．これらの可視化では，2 つの点は分布が重複していることが読み取れる．そのため，t-SNE による 3 次元への次元圧縮では，ラベルの一致・不一致は分離できなかった．

また，表 5.7 には 1 層から 12 層までの正解率を示した．各層の正解率は，いずれも 0.54 前後であり，層毎の比較において差はないと考えた．また，これらは 2 つのクラスタでの正解率であるため，クラスタにはうまく分かれていないことが言える．このことは，図 5.41 から図 5.52 までの K-means 法によるプロットの結果か

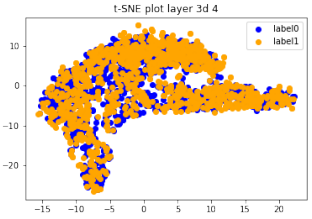

図 5.32　4 層目での出力

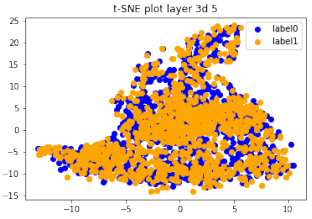

図 5.33　5 層目での出力

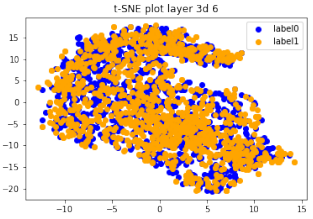

図 5.34　6 層目での出力

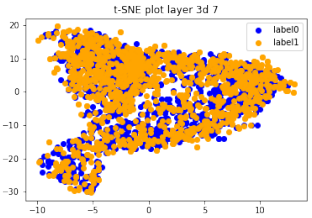

図 5.35　7 層目での出力

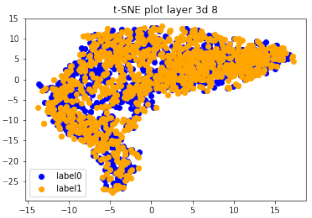

図 5.36　8 層目での出力

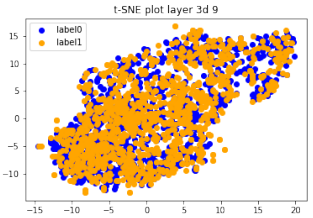

図 5.37　9 層目での出力

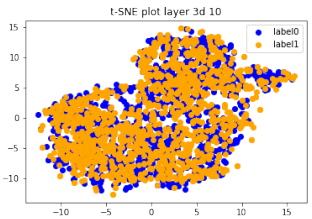

図 5.38　10 層目での出力

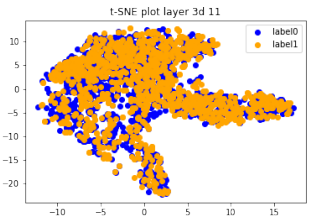

図 5.39　11 層目での出力

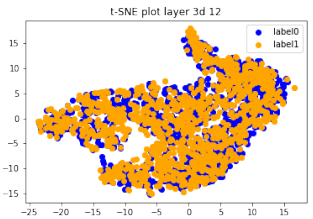

図 5.40　12 層目での出力

らも読み取ることができる．

BERT の出力を 3 次元に圧縮した際の凝縮度の結果を図 5.53 に示した．それぞれの図で，横軸は BERT の 12 層の層番号，縦軸は凝縮度としている．. 平均値や中央値に着目すると 3 層から 12 層においては，値は約 13 を取っているのに対して，1 層，2 層では値は約 10 を取っており違いが読み取れた．第一四分位数に着目すると 1 層，2 層では，値は約 9.5 を取っており，それ以外の層では，約 12 を取っている．第三四分位数に着目すると 1 層，2 層では，値は約 12 を取っており，それ以外の層では，約 14 を取っている．また四分位範囲では，3 層から 12 層において

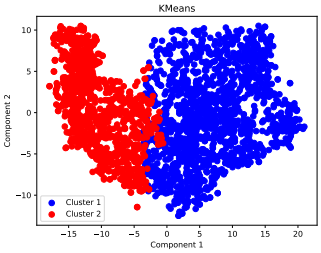

図 5.41　1 層目でのクラスタリング

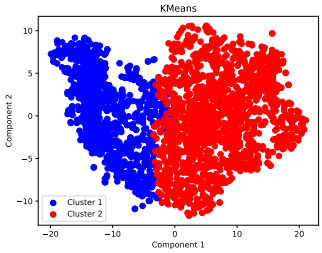

図 5.42　2 層目でのクラスタリング

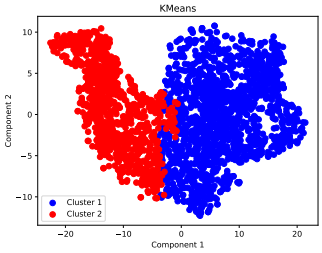

図 5.43　3 層目でのクラスタリング

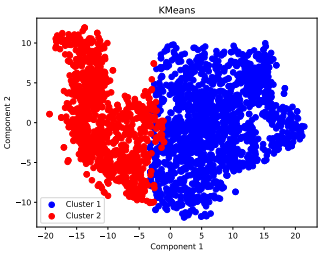

図 5.44　4 層目でのクラスタリング

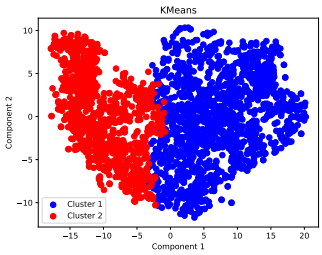

図 5.45　5 層目でのクラスタリング

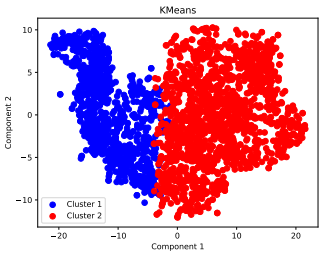

図 5.46　6 層目でのクラスタリング

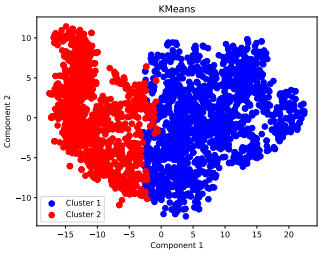

図 5.47　7 層目でのクラスタリング

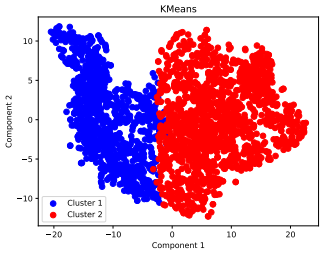

図 5.48　8 層目でのクラスタリング

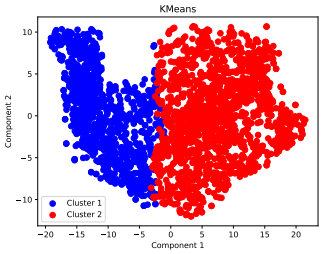

図 5.49　9 層目でのクラスタリング

は，約 12 から 14 の範囲で値を持っているのに対し，1 層，2 層では約 10 から 12 の範囲であった．これらから実験の結果，2 つのクラスタにおける凝縮度には，層毎による違いあることがわかる．

BERT の出力を 3 次元に圧縮した際の乖離度の結果を図 5.54 に示した．それぞ

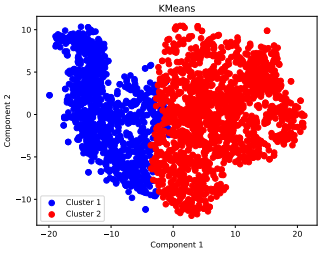

図 5.50　10 層目でのクラスタリング

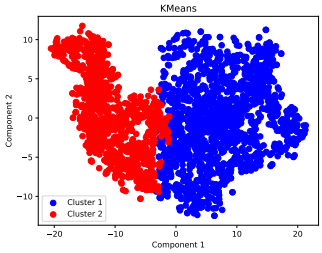

図 5.51　11 層目でのクラスタリング

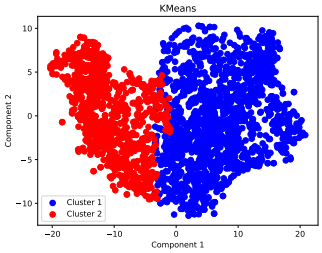

図 5.52　12 層目でのクラスタリング

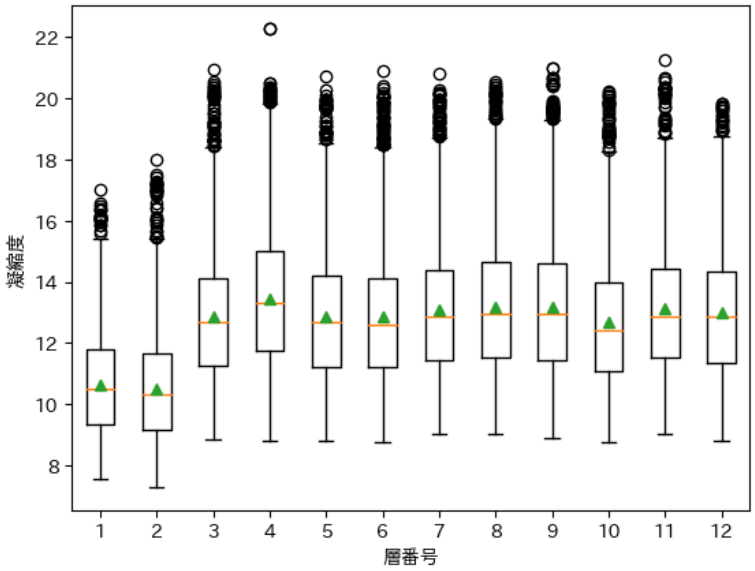

図 5.53　3 次元における凝縮度の箱ひげ図

れの図で，横軸は BERT の 12 層の層番号，縦軸は乖離度としている．平均値や中央値に着目するとどの層においても，値は約 20 を取っており，差がないと言える．第一四分位数に着目するとどの層においても約 18 を取っている．第三四分位数に着目すると 1 層，2 層では，値は約 22 を取っており，それ以外の層では，約 24 を取っている．また四分位範囲では，3 層から 12 層においては，約 18 から 24 の範囲で値を持っているのに対し，1 層，2 層では約 18 から 22 の範囲であった．これらから実験の結果，2 つのクラスタにおける凝縮度には，層毎による違いあること

がわかる．

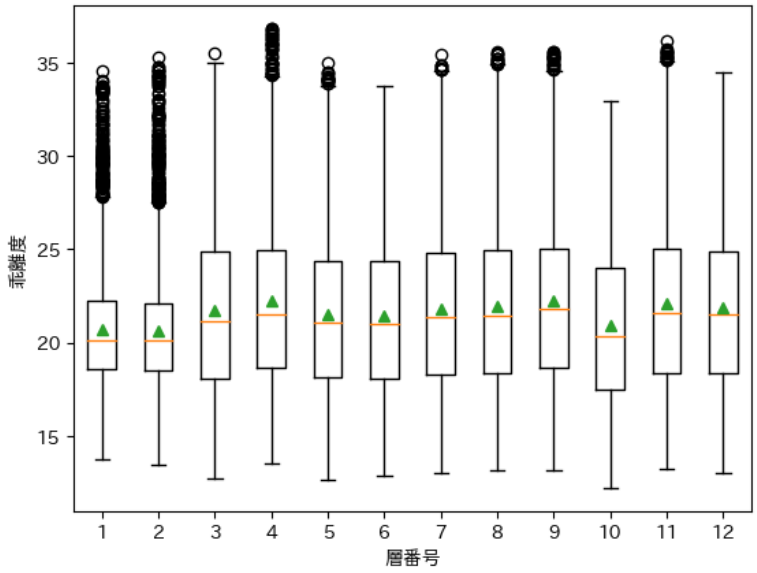

図 5.54　3 次元における乖離度の箱ひげ図

凝縮度と乖離度での各層の比較を行右ことで，凝縮度が低く乖離度が高い層があることを期待した結果，1 層，2 層の分布が他の層より低いことが読み取ることができた．低いことが読み取れた 1 層，2 層では他の層と比べて，クラスタ内の密度が低く，他クラスタとの区別が不明瞭であると考えられる．これらから，BERT の出力を 3 次元に次元圧縮した際の，ラベルの一致・不一致におけるデータの分離は，1 層，2 層は他の層よりも小さいことがいえる．

5.7　実験 3:768 次元の出力での比較

表 5.8 には 1 層から 12 層までの正解率を示した．各層の正解率は，いずれも 0.54 前後であり，層毎の比較において差はないと考えた．また，これらは 2 つのクラスタでの正解率であるため，クラスタにはうまく分かれていないことが言える．

2 つのクラスタ間の凝縮度の結果を，図 5.55 に示した．それぞれの図で，横軸は BERT の 12 層の層番号，縦軸は凝縮度を示す．平均値や中央値に着目するとどの層においても，値は約 1.6 を取っており，差がないと言える．第一四分位数に着目

表 5.8　2 次元での K-means 法による正解率

層	1	2	3	4	5	6
正解率	0.546	0.535	0.543	0.537	0.546	0.546
層	7	8	9	10	11	12
正解率	0.544	0.541	0.540	0.539	0.535	0.537

するとどの層においても，値は約 1.3 を取っている．第三四分位数に着目するとどの層においても，値は約 1.8 を取っている．四分位範囲については，各層 0.5 の値を持っており，データの散らばり方も類似していることが読み取れる．これらから実験の結果，2 つのクラスタにおける凝縮度には，層毎による違いがないことがわかる．

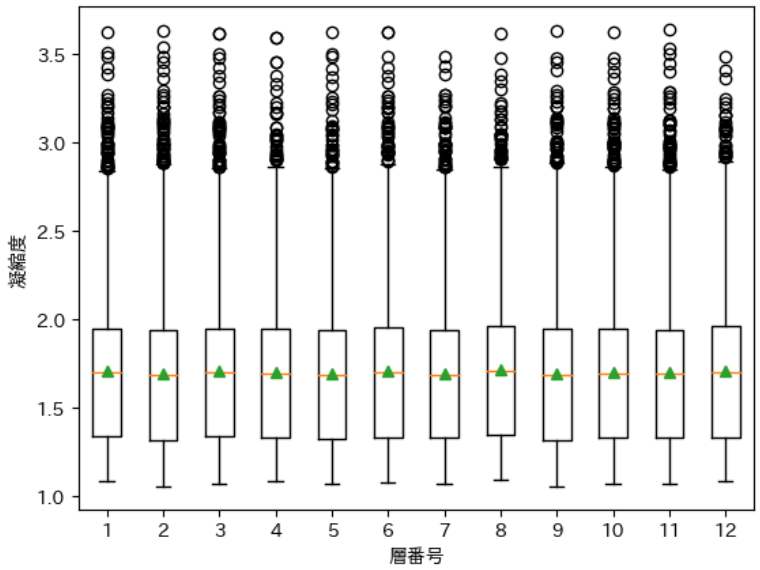

図 5.55　768 次元における凝縮度の箱ひげ図

また 2 つのクラスタ間の乖離度の結果を図 5.28 に示した．それぞれの図で，横軸は BERT の 12 層の層番号，縦軸は乖離度としている．平均値や中央値に着目するとどの層においても，平均値は約 2.5，中央値は約 2.6 を取っており，差がないことが言える．第一四分位数に着目するとどの層においても，値は約 2.1 を取ってい

る．第三四分位数に着目するとどの層においても，値は約 3 を取っている．四分位範囲については，各層約 0.9 の値を持っていた．そのため，データの散らばり方は類似していることが読み取れる．これらから実験の結果，2 つのクラスタにおける乖離度には，層毎による違いがないことがわかる．

凝縮度と乖離度での各層の比較を行った．どちらに評価指標においても，層毎による違いを読み取ることができなかった．そのため，BERT の出力を 768 次元のまま用いた際の，ラベルの一致・不一致では，データの分離に違いがないことが言える．

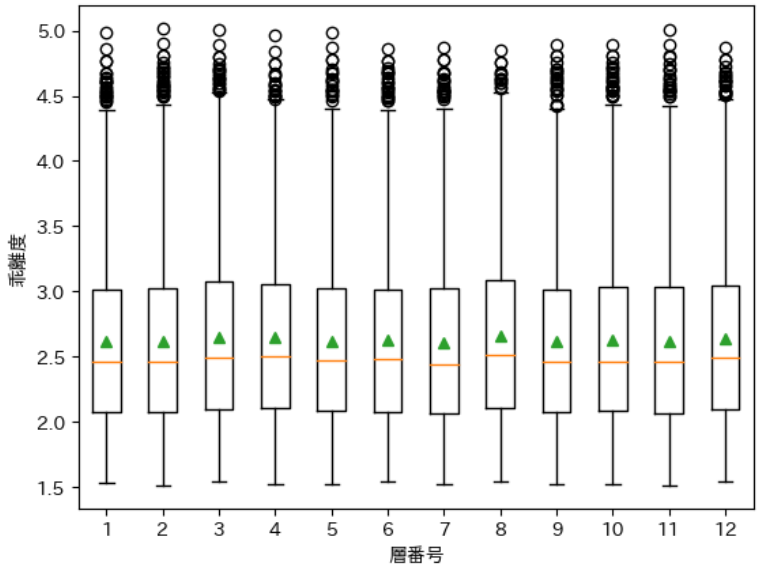

図 5.56　768 次元における乖離度の箱ひげ図

第 6 章　おわりに

本研究では，評価者間でのラベルの一致・不一致の判別することにより，追加のラベル付けにおける作業件数の低減を目的としている．既存のデータセットのラベルの不一致の解消のために，ラベル付けを行うには，作業件数に応じた費用が掛かる．この費用を減らす上では，ラベル付けの作業件数を減らす必要がある．そのため，追加のラベル付けの前後で共にラベルの不一致のデータを事前に判別することでラベル付けの件数を減らすことを考えた．本研究では，ラベルの一致・不一致の判別のために，BERT の各層に着目することで，ラベルの一致・不一致の判別が可能な層を見つけることを考えた．BERT の各層は，テキストから異なる種類の情報を捉える能力を持っている．また，ラベルの一致・不一致の判別が困難が要因として，言語の微妙な差異の解釈が難しいことが挙げられる．この能力を用いることで，複数の評価者が同じテキストに対して，異なるラベルを付与した理由となる言語の差異を深く理解することができると考えた．

評価実験では，まず追加投票後のラベルの一致・不一致のテキストデータを前処理してトークン化し，BERT に入力する．次に，BERT の 12 層の各層それぞれから，[CLS] トークンの特徴ベクトルを得る．それから，t-SNE を使用してこれらの特徴ベクトルを低次元に次元圧縮する．最後に，k-means 法により 2 つのクラスタへとクラスタリングを行う．これにより，ラベルの一致・不一致それぞれのクラスタがどのように分離しているかを各層にて比較した．比較する上で，次元圧縮による情報の損失を考慮して，3 種類の次元数を用いた．

実験の結果，凝縮度と乖離度での各層のどちらに評価指標においても，層毎による違いがなく，データの分離がないことがわかった．また，これは用いる次元数を変えたとしてもほとんど同様の結果となった．3 次元に次元圧縮した場合の 1 層と 2 層についてのみ，分布に違いが生じた．これらの結果から，BERT の各層の出力の比較から，ラベルの一致・不一致を判別することが可能な層を見つけることができなかった．これは，事前学習の汎用的な言語理解では，ラベルの一致・不一致の判別に必要となる言語の差異を学べていないことを示している．

本研究では，ラベルの一致・不一致の判別をするための BERT の層を見つけることができなかった．そのため，ラベルの一致・不一致の判別まで行うことができな

かった．今後の展望としては，教師あり学習として，BERT の全ての層のパラメータを変更可能にした状態で学習を行ったモデルを用いることで，データの分離が見られる層を見つける．そして，その層を用いることで実際にラベルの一致・不一致の判別を行えるかを確認する．そして，判別したいテキストの BERT の層の出力が，ラベルの一致のクラスタとラベルの不一致のクラスタの，どちらのクラスタが中心に近いのかを調べることで，ラベルの一致・不一致を判別する．

謝辞

本研究を進めるにあたり，指導教員の鈴木優准教授にはたくさんのご指導をしていただきお世話になりました．研究が進まない際には，長い場合 6 時間もの多くの時間をかけて相談に乗ってくださり研究を進めるために必要なアドバイスをして下さいました．ありがとうございました．

秘書の佐野さん，井尾さんには事務的な手続きの際にお世話になりました．おかげさまで研究活動を円滑に進めることができました．ありがとうございました．

研究室の方々には，日常において雑談から相談まで様々な面でお世話になりました．雑談という面では食事などに行く際に日常的な他愛もない話をすることで仲を深め，いい雰囲気で研究活動を行うことができました．研究の相談という面でも，研究室の方々には様々の意見をいただくことでたくさんの協力をしていただきました．ありがとうございました．

自分は大学院からの鈴木研究室への加入となりましたが，他大出身であるため，友達も全く居らず，何もかもがわからない状況で最初は不安が大きかったです．しかし，先生や秘書さん，研究室の方々に支えていただいたおかげで，日々を楽しく過ごすことができました．皆様の支援と協力に心より感謝いたします．

参考文献

[1] Jeff Howe, et al. The rise of crowdsourcing. *Wired magazine*, Vol. 14, No. 6, pp. 176–183, 2006.

[2] Joseph Chee Chang, Saleema Amershi, and Ece Kamar. Revolt: Collaborative crowdsourcing for labeling machine learning datasets. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pp. 2334–2346, 2017.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[4] Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does bert learn about the structure of language? In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[5] 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*, Vol. 30, , 2017.

[6] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, Vol. 9, No. 11, 2008.

[7] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, Vol. 20, pp. 53–65, 1987.

[8] Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B Viegas, Andy Coenen, Adam Pearce, and Been Kim. Visualizing and measuring the geometry of bert. *Advances in Neural Information Processing Systems*, Vol. 32, , 2019.

[9] 白静, 田中裕隆, 曹類, 馬ブン, 新納浩幸ほか. Bert の下位階層の単語埋め込み表現列を用いた感情分析の教師なし領域適応. 研究報告自然言語処理 (NL), Vol. 2019, No. 17, pp. 1–6, 2019.

[10] Matteo Venanzi, John Guiver, Gabriella Kazai, Pushmeet Kohli, and Milad

Shokouhi. Community-based bayesian aggregation models for crowdsourcing. In *Proceedings of the 23rd international conference on World wide web*, pp. 155–164, 2014.

発表リスト

[1] 小林大記，鈴木優『トピックモデルとシソーラスから評価項目を特定する意見集約システムの構築手法』東海関西データベースワークショップ 2022，2022
[2] 小林大記，鈴木優『評価観点に着目した意見集約システムの構築』2023 年電子情報通信学会総合大会，2023
[3] 小林大記，鈴木優『評価者間の票が一致しない原因の分類によるクラウドソーシングのインストラクション改善』東海関西データベースワークショップ 2023，2023