令和 3 年度茨城大学工学部情報工学科卒業研究論文

文書分割を利用した BERT による半教師あり学習

所属 情報工学科

著者 大塚拓海 (18T4020Y)

指導教員 新納浩幸教授

令和 3 年 2 月 5 日 (水)

令和 3 年度茨城大学工学部情報工学科卒業研究論文

文書分割を利用した BERT による半教師あり学習

著者

大塚拓海 (18T4020Y)

指導教員

新納浩幸教授

論文要旨

近年, 機械学習によって自然言語処理能力は大きく発展した.2018 年に Google が BERT [1] を発表したことからも年々自然言語処理タスクの精度は向上している. 事前学習モデルである BERT は,Masked Language Model と Next Sentence により双方向 Transfomer を実現しており, 様々な文書分類タスクにおいて高い結果を出している.

しかしその一方で, データのラベル付けの際のコストの大きさや, そもそも特定の分野ではデータの収集自体が困難などの問題もいまだある. そして BERT 自体も 512 トークンを超える文書を処理できないというようなデメリットも存在している.

このような問題を解決するべく, 本稿では文書を分割しデータの拡張を試みると同時に 512 トークン以上の文書を BERT で扱える半教師あり学習の手法を提案する. 具体的な手法として livedoor ニュースコーパスを用いて少量のラベル付きデータから分類器を作成し, 奇数と偶数で 2 分割したアンラベルデータに信頼度が閾値を超えた場合ラベル付けをする. そして分割したデータを訓練データに加えもう一度学習させる. 実験の結果から精度が上がったことが分かった. さらに行った追加実験により, データ拡張の面からみてもデータを 2 分割したほうが高い精度が出ることを示した.

目次

第 1 章

序論

自然言語とは, 人々が日常的に使う自然言語をコンピュータに処理させるための一連の技術である. 近年 BERT [1] の登場により自然言語処理能力は大幅に向上した. そんな BERT にはその性質上以下の制約がある. 入力される文章は固定長になるようにしなければならず, その最大長は 512 トークンである. 扱う文章が 512 トークン以下の短い文章ならば,512 トークンとなるように意味を持たない単語で埋めるだけでいい. しかし扱う文章が 512 トークンを超える場合, それ以降の単語は捨てられてしまい, 文章すべての情報を受け取ることが出来ない.　また自然言語処理はディープラーニングを用いて　本論文では BERT を用いた文書分類タスクを行う.livedoor ニュースコーパス内の文章を 2 分割にすることにより、512 トークンを超えた文章から情報を得ることを試みる. また 1 つの文章から 2 つのデータを生成することで, データ拡張による半教師あり学習を試みる.

第 2 章

関連研究

2.1 テキスト分類

自然言語処理では機械翻訳, 文書生成, 検索, 文書要約など様々なタスクがあるが, 本稿ではテキスト分類を扱う. テキスト分類とは図 2.1 のようにある文書 A をカテゴリ (1),(2),(3) に分類する場合, あらかじめ各カテゴリの特徴を抽出し, 文書 A の特徴を抽出した後に文書 A に似ている特徴を持つカテゴリに分類していくタスクである.

近年ではこの特徴を抽出するために機械学習が用いられており, その中でも様々な手法が存在する. 文書を機械学習で扱うためには文書のベクトル化する必要があり, このベクトル化のことを埋め込みといい, 埋め込みによって生成されたベクトルを埋め込み表現という. 埋め込み方法も様々な手法があり, 大きく分けて文書の単語の出現頻度をもとに埋め込みを行うカウントベースの埋め込みと, 単語の意味を推測することで分散表現を得る推論ベースの埋め込みの 2 つがある. カウントベースの埋め込み手法の例では BoW があり, 推論ベースの例として word2vec [2] がある. 本稿で扱う BERT も推論ベースである.

2.2 半教師あり学習

機械学習においてデータというのはとても重要である. ラベルありの大量のデータを使い学習を行う教師あり学習においては, データの数が分類精度に直結するからである. しかしデータにラベル付けを行うのは一般的に人の手による作業となるので, 大量のラベルありデータを用意することは非常にコストがかかる. 文章の特徴が似ているか否かでグループ分けを行い学習させる教師なし学習はラベルつけがされていないデータ (アンラベ

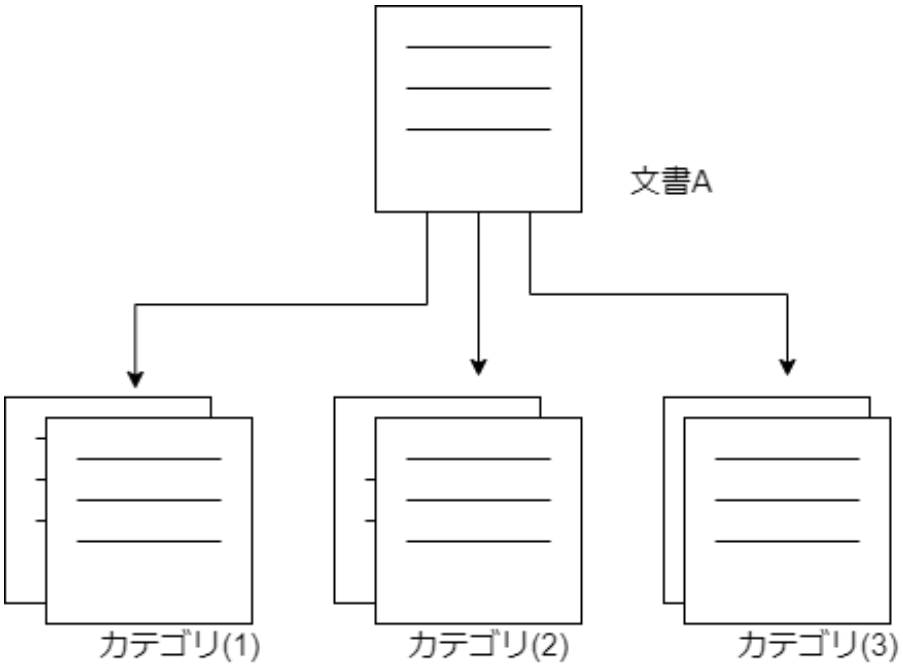


図 2.1: テキスト分類の説明

ルデータ) のみで学習を行えるので, ラベル付きデータを用意するコストが必要がない代わりに一般的に精度が教師あり学習に比べて落ちてしまう. 教師あり学習と教師なし学習のメリット, デメリットをまとめたものを表 2.1 に示す.

そこで近年注目されているのが, 半教師あり学習である. 半教師あり学習とは少量のラ

表 2.1: livedoor ニュースコーパスのファイルごとのファイル数

学習方法	メリット	デメリット
教師あり学習	正解が決まっているので高い精度が期待できる	大量のラベル付きデータを用意しなければならない
教師なし学習	アンラベルデータのまま学習できるので,ラベル付けのコストがかからない	正解が決まってないので教師あり学習に比べ,精度が落ちてしまう

ベル付きデータと大量のアンラベルデータを用いて学習を行う手法である. 文書分類での半教師あり学習は主に 2 つあり,1 つはラベル付きデータで分類器を作成しその分類器によってアンラベルデータを活用する分類器に基づく半教師あり分類学習であり,2 つ目は

データ分布によってデータをクラスタリングし近いラベル付きデータに基づいてアンラベルデータを活用する半教師ありクラスタリングである. 本研究では半教師あり分類学習を参考に分類器を用いて半教師あり学習を行った.

2.2.1　半教師あり分類学習

半教師あり分類学習の主な手順としては以下の通りである.

1. ラベル付きのデータを用いて分類器を作成する
2. 作成した分類器によりアンラベルデータに対しラベリングを行う
3. ラベリングしたアンラベルデータを訓練データに加え再度学習する

この手順をまとめたものを図 2.2 に示す. またこの半教師あり分類学習にも様々な手法が

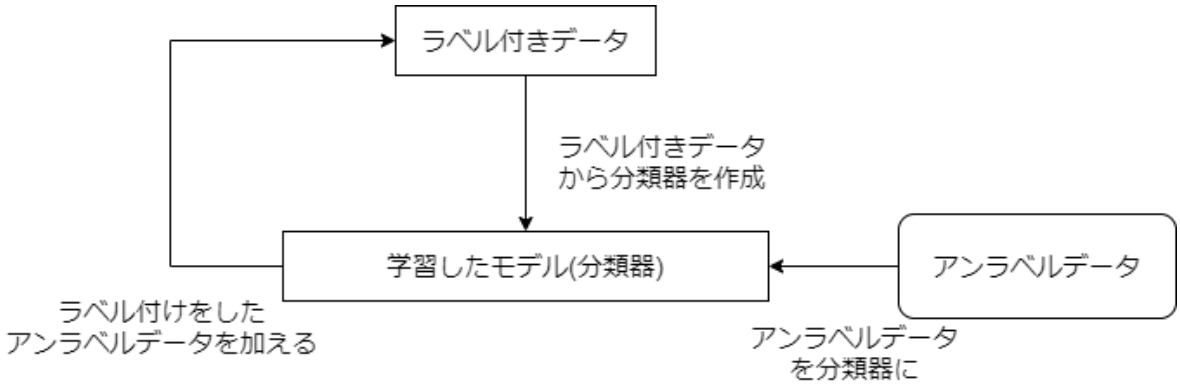


図 2.2: テキスト分類の説明

存在しており,1 つの分類器で学習を行う自己訓練 [3] や複数の分類器を用いる共訓練 [4] などがある.

2.3　データ拡張

2.2 でも述べた通り機械学習においてデータ数は多ければ高い精度が期待できる. 教師あり、教師なし学習において問題だったのはデータのラベル付けの際のコストであったが, そもそもデータ数が少ない分野ではラベル付き, ラベル無し以前にデータを集めるのが難しい. 例えば特定の病気に関するデータなどの狭い範囲ではデータ収集が困難であることが考えられる. そこで近年注目されているのがデータ拡張 (Data Augmentation)

である. 画像処理の分野での研究が多いが, 自然言語処理の分野でもデータ拡張に注目している論文もいくつもある.EDA(Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks) [5] もそのうちの一つである.

2.3.1　EDA

EDA とは Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks の略であり, 文書から同義語の入れ替え, ランダムな挿入, ランダムな入れ替え, ランダムな削除という 4 つの簡単な操作でデータ拡張を行う手法である. それぞれの説明を表 2.2 に示す. この実験の結果,CNN+EDA,RNN+EDA の両方において

表 2.2: EDA の操作の説明

名前	操作
同義語置換	ストップワードではない n 個の単語をランダムに選ぶ。これらの単語をランダムに選んだ同義語に置き換える。
ランダム挿入	文章中のストップワードではないランダムな単語のランダムな同義語を見つける。その同義語を文中のランダムな位置に挿入する。これを n 回行う
ランダムスワップ	文章中の 2 つの単語をランダムに選び、その位置を入れ替える。これを n 回行う。
ランダム削除	文章中の各単語を確率 p で削除する。

精度の向上が見られた. しかし大規模なデータセットや事前学習済モデルでは十分な効果得られなかった.

2.4　BERT

BERT(Bidirectional Encoder Representations from Transformers) は 2018 年に Google から発表された自然言語処理モデルである.BERT は様々な自然言語処理タスクにおいて高いスコアを出した.BERT は事前学習モデルでありラベルが付与されていない分散表現を Transformer [6] が処理することによって学習を行う.Transformer とは 2017 年に発表されたモデルであり,Attention 機構と呼ばれる仕組みのみをつかって様々

な自然言語タスクで使用できる.BERT では事前学習が行われるが, 事前学習とは様々なタスクへの応用なためにに, 有効な特徴量を学習することである. 本来ならば処理するタスクごとにモデルを変える必要があるが,BERT では目的にあわせ事前学習したモデルにファインチューニングによって層を追加するだけで良いので汎用性が高い. またこのモデルの大きな特徴として深い双方向性を実現しているというものがある.

2.4.1　BERT への入力

BERT への入力はタスクごとに違う. よって様々なタスクに応用できるように BERT では入力するトークンに工夫がされている.

具体的な説明として [CLS] という特別なトークンを常に最初におくが, これは「classification embedding」と呼ばれ, 分類タスクをする際に使用され, この単語に位置に対応する隠れ層の値によって分類できる.

また answer-question などの入力が 2 文以上になる場合に対応できるように BERT では文を 2 つの方法で区別している.1 つめは [SLP] トークンと呼ばれる特別なトークンを使用して文を区別する.2 つ目は「Segment Embeddings」と呼ばれる各トークンがどの文に属するかという情報を示す埋め込みによって区別する. ほかにも単語情報を示す「Token Embeddings」やそのトークンがどの位置にあるのかを示す「Position Embeddings」というよな埋め込みを各トークンで行う.

これらのことを説明したものを図 2.3 に示す.

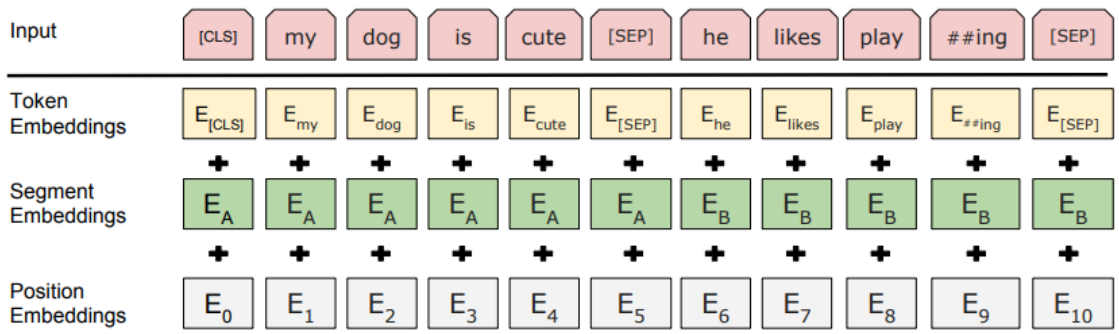


図 2.3: BERT への入力 (元論文 [1] より引用)

2.4.2　事前学習

BERT では事前学習を行い, そのパラメータをファインチューニング時の初期値と設定することで学習を行う. 深い双方向表現を得るために事前学習時に「Masked Language Model」と「Next Sentence Prediction」と呼ばれる BERT 特有の 2 つのタスクが設定されている.

Masked Language Model

BERT 以前の自然言語モデルは単一方向からしか処理できない. そのため目的の単語の予測をするためには, それ以前の文章から予測する必要があった. さらにもし双方向で処理するならば, 従来の方法では双方向であるがゆえに予測する単語の情報を得てしまっているため, 学習がうまくいかない. そのため深い双方向性表現を得るために BERT では, 入力トークンの一部をランダムにマスクし, それらのマスクされたトークンを予測するという手法がとられている. 具体的な手法として

1. 入力文の 15 %の単語を 80 %で [MASk] に変換
   my dog is hairy → my dog is [MASK]
2. 10 %をランダムな別の単語に変換
   my dog is hairy → my dog is apple
3. 残りの 10 %はそのままの単語で残しておく.
   my dog is hairy →　 my dog is hairy

以上の手順で置換された単語を周りの文脈から当てるタスクを解くことにより, 単語に対応する文脈を学習する.

Next Sentence Prediction

自然言語処理タスクの多くが、文同士の関係性を理解することに基づいているということから, このタスクが設定された. 手法としては 2 つの入力された文に対してその 2 文がとなりあっているか当てるように学習させる. 具体的な手法としてつながりのある 2 文の片方を 50 %で他の文に置換し, 隣り合っているなら IsNext, いないなら NotNext とし,

当てるように学習させる.

Input = [CLS] the man went to [MASK] store [SEP] he bought a gallon [MASK] milk [SEP]

output = IsNext

Input = [CLS] the man went to [MASK] store [SEP] penguin [MASK] are flight less birds [SEP]

output = NotNext

2.4.3　ファインチューニング

BERT のファインチューニングは事前学習で得られたパラメータを初期値として, 各タスクごとに層を追加するだけでよい. 各タスクごとの層の説明を図 2.4 に示す.

Next Sentence Prediction

2.4.4　BERT のベンチマーク

GLUE というベンチマークテストにおいて BERT は 8 項目のデータセットすべてで既存モデルよりも高い数値を出した.

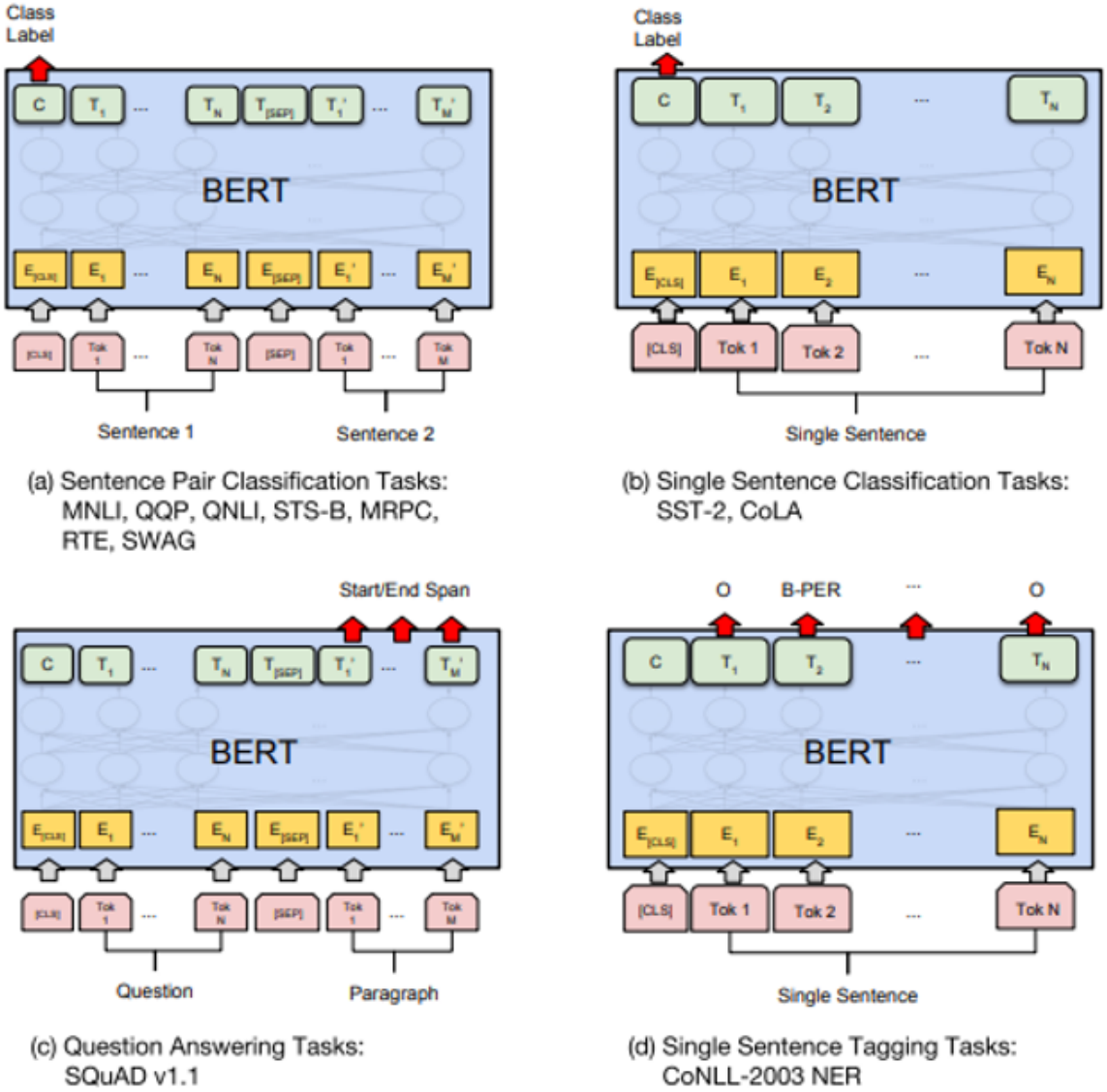


図 2.4: BERT への入力 (元論文 [1] より引用)

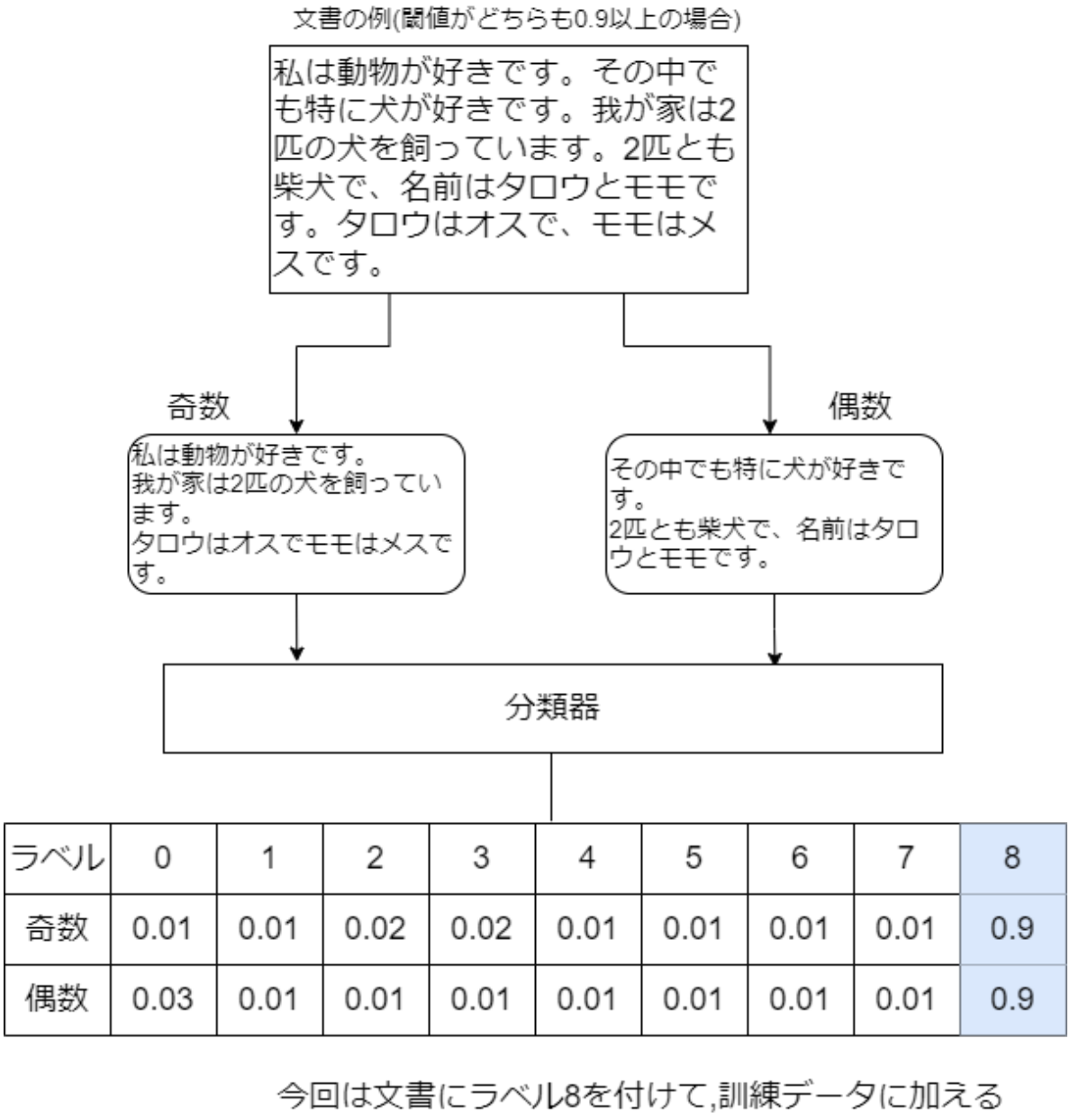
第 3 章

提案手法

BERT では最大文字列は 512 トークンである. よって 512 トークン以上の文書を入力する際には, 工夫が必要である. そこで本研究では 1 つの文書を 2 分割することにより 512 トークン以上の文書を扱える半教師あり学習を行う手法を示す.

1. ラベル付きのトレーニングデータ, およびテストデータを用いて分類器を作成する.
2. アンラベルデータの文書を句点で区切り, 奇数と偶数で 2 分割する. 分類器より奇数, 偶数それぞれの文書のラベルの確率を出す.
3. 本研究では,
   ①片方の値が 0.9 以上, もう片方の値が 0.1 以下
   ②どちらも値も 0.9 以上
   の 2 つで実験を行った
4. 3. で出した確率をもとに奇数, 偶数両方のデータにラベル付けを行う.
5. ラベル付けを行った奇数, 偶数両方のデータを訓練データに加えもう一度学習し文書分類を行う.

なお, 上記の手順を図 3.1 に示す.



ラベル	0	1	2	3	4	5	6	7	8
奇数	0.01	0.01	0.02	0.02	0.01	0.01	0.01	0.01	0.9
偶数	0.03	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.9

図 3.1: 手法の説明 (閾値はどちらも 0.9 以上)

第 4 章

実験

4.1 概要

本実験ではラベルがある訓練データ, テストデータから分類器を作成し, その分類器からアンラベルデータに対しラベル付けを行う. その後ラベル付けをしたデータを訓練データに加え学習を行い, その精度を加える前と比較し本研究手法の効果を調べた.

4.2 条件

4.2.1 使用した BERT モデル

本実験では東北大学乾・鈴木研究室が作成した日本語 BERT のうちの一つである「bert-base-japanese」を使用した.

4.2.2 使用したコーパス

コーパスは livedoor ニュースコーパスを使用した.livedoor ニュースコーパスとは NHN Japan 株式会社が運営している「livedoor ニュース」から, 下記の 9 ジャンルで作成したコーパスである

・ラベル 0: 毒女通信

・ラベル 1:IT ライフハック

・ラベル 2: 家電チャンネル

・ラベル 3:livedoor HOMME

・ラベル 4:MOVIE ENTER

・ラベル 5:Peachy

・ラベル 6: エスマックス

・ラベル 7:Sports Watch

・ラベル 8: トピックニュース

またそれぞれのファイル数を表 4.1 に示す. またこのコーパスの最大とトークン数は 984 トークンであり,BERT の入力上限である 512 トークンを超える文書は 3203 ファイルである.

表 4.1: livedoor ニュースコーパスのファイルごとのファイル数

ラベル	0	1	2	3	4	5	6	7	8
ファイル数	1170	1170	1164	811	1170	1142	1170	1200	1070

4.3　実験手法

4.3.1　データ数

本実験では 4.2.2 で述べた livedoor ニュースコーパスのデータを, 訓練時に使うラベル付きデータ (train), 作ったモデルをテストするときに使うラベル付きデータ (test), そして何のラベルもつけないアンラベルデータ (unlabel) の 3 つに分けた. それぞれのデータ数を表 4.2 に示す.

4.4　作成したモデルの説明

実験で作成した分類器およびアンラベルデータを加え再学習したモデルについての詳細だが,3 層のニューラルネットワークであり, それぞれ 768 次元,768 次元,9 次元となっている. またエポック数は 8 エポックで学習を行った. どちらも最適化関数は確率的勾配降下法 (SGD) を使い, 交差エントロピーによって損失を求めた. 本研究は多ジャンル分類であり, 結果が確率として出るという性質から出力層には softmax 関数を使用した.

softmax 関数は

表 4.2: livedoor ニュースコーパスデータの振分け

ラベル	train	test	unlabel
0	450	50	670
1	450	50	670
2	450	50	664
3	450	50	311
4	450	50	670
5	450	50	642
6	450	50	670
7	450	50	700
8	450	50	570
sum	1350	450	5567

- 0 から 1 で値が出力される
- すべての値の出力の合計値が 1 である

という特徴から他クラス分類の出力層で利用できる. また sofmax 関数を式 4.1 に示す.

$$y_i = \frac{e^{x_i}}{\sum_{k=1}^{n} e^{x_k}} \tag{4.1}$$

4.4.1　分類器の精度

test データと train データから作った分類器を作る. この際の分類器の精度を表 4.3 に示す. この分類器をもとに unlabel データのラベル付けを行う.

表 4.3: 分類器の精度

test データ	正解数	正解率
450	448	0.987

4.5　加えたデータの正解率

今回 unlabel データとして 5567 ファイル用意したが, それらのファイルは本来ラベルがついている. そこで分類器で識別し, 加えたデータの本来のラベルとの正解率を表 4.4 に示す

表 4.4: 加えたデータ

閾値	加えたデータ数	正解数	正解率
片方が 0.9 以上 片方が 0.1 以下	1122	850	0.758
どちらも 0.9 以上	6648	5954	0.923

4.6　比較実験

本手法の有効性を調べるために,unlabel データを奇数, 偶数に分けず閾値 0.9 を超えたデータだけ train データに加えたモデルを作成する. 加えたデータ数と正解率を表 4.5 に示す.

表 4.5: 比較実験の加えたデータ

閾値	加えたデータ数	正解数	正解率
0.9 以上	4018	4441	0.905

4.7　実験結果

閾値が片方 0.9 以上で片方が 0.1 以下 (1), どちらも 0.9 以上 (2), 比較実験 (3) での実験結果を表 4.6 に示す．このことより,2 分割したデータの閾値がどちらも 0.9 以上のとき, 本研究の手法が有効であることが分かる.

表 4.6: 実験結果

条件	test データ数	正解数	正解率
(1)	450	432	0.960
(2)	450	440	0.978
(3)	450	437	0.971

4.8　追加実験

本手法では 2 分割したデータを分割したまま訓練データに加えることにより, データの拡張を行った. 分割したデータの値がどちらも 0.9 以上のとき有効であることが分かった. そこで新たにデータを 2 分割しどちらも閾値 0.9 以上のデータを 2 分割する前の元の文書で訓練データに加えた (4) モデルを作成することにより, データ拡張の面でも本研究手法が有効であることを示す.(4) の説明を図 4.1 に示す. また加えたデータ数は 3224, そのうち正解数は 2977 であり正解率は 0.923 となった.

4.9　追加実験の結果

追加実験の実験結果を表 4.7 に示す (閾値はどちらも 0.9 以上). このことより文書を 2 分割して訓練データに加える方が精度が上がっており, データ拡張の面でも本手法が有効であることがわかる.

表 4.7: 追加実験の結果

加えたデータ	test データ数	正解数	正解率
2 分割	450	440	0.978
元の文書	450	437	0.971

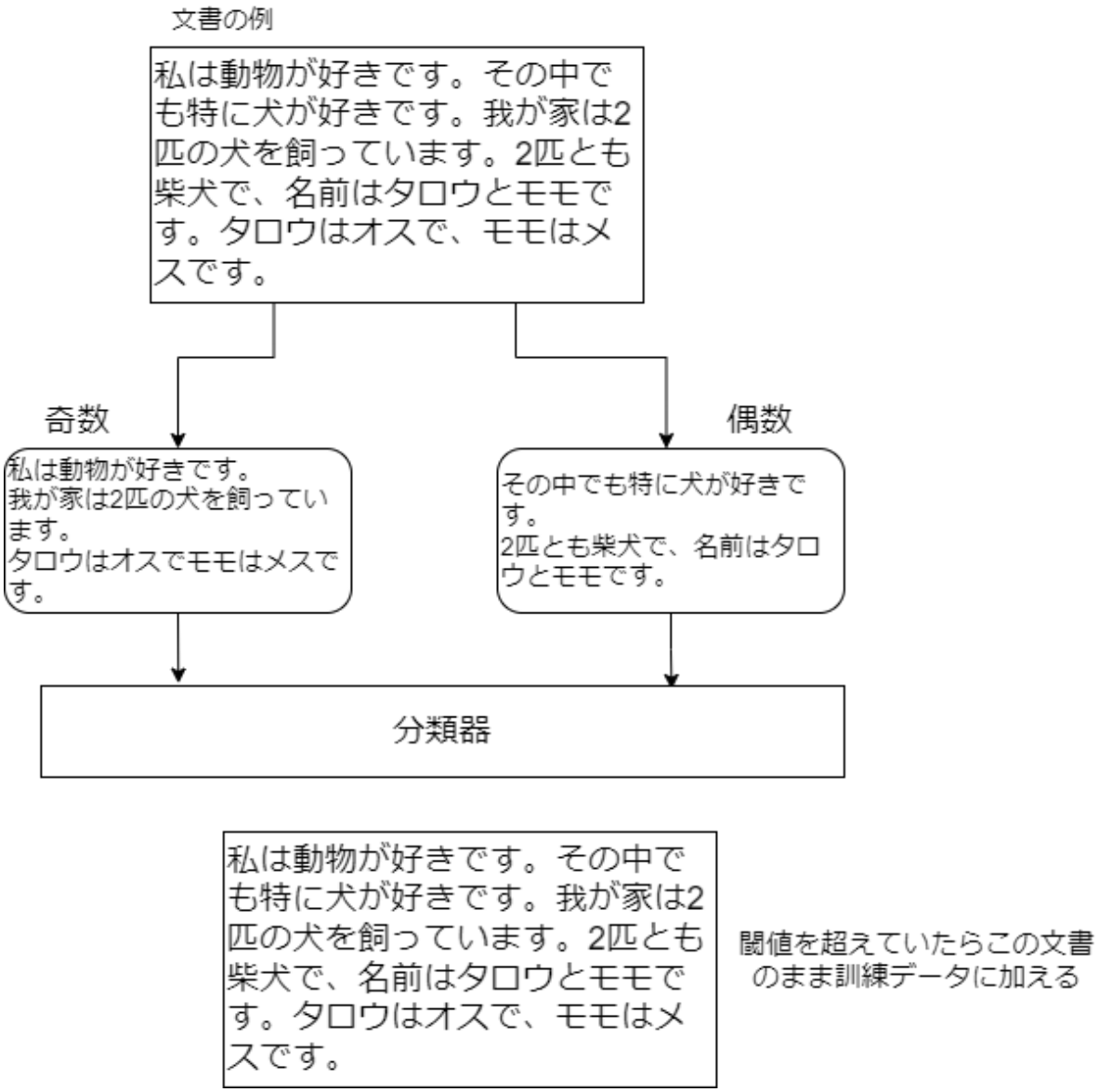

文書の例
私は動物が好きです。その中でも特に犬が好きです。我が家は2匹の犬を飼っています。2匹とも柴犬で、名前はタロウとモモです。タロウはオスで、モモはメスです。
奇数
偶数
私は動物が好きです。
我が家は2匹の犬を飼っています。
タロウはオスでモモはメスです。
その中でも特に犬が好きです。
2匹とも柴犬で、名前はタロウとモモです。
分類器
私は動物が好きです。その中でも特に犬が好きです。我が家は2匹の犬を飼っています。2匹とも柴犬で、名前はタロウとモモです。タロウはオスで、モモはメスです。
閾値を超えていたらこの文書のまま訓練データに加える

図 4.1: 追加実験の説明

第 5 章

考察

(2) と比較実験によりデータを 2 分割することの有効性が, 追加実験により 2 分割することによるデータ拡張の有効性が分かった.

(1) は比較実験と比べると精度が落ちているが, 元の文書は同じであるが分割するとそれぞれ全く違うクラスと判別されることがあるというのは重要である. このことに注目してさらに研究していく価値があると考える. さらに本手法ではデータの分割方法が奇数, 偶数のみであったが分割方法を前半, 後半に分ける手法や 2 分割以上の分割方法などの手法も試してみる必要がある.

第 6 章

結論

本論文では文書分割を用いて BERT による文書分類タスクをする手法を試みた.BERT では 512 トークンしか入力できないため, 文書を分割することにより長い文章でも BERT で入力できるようにするのが目的である. さらにデータを分割することにより機械学習の課題である, データの収集の難易度を緩和することが出来ると考えた.livedoor ニュースコーパスを使い, 分類器を作成し 2 分割した unlabel データにラベル付けを行い訓練データに加えて学習させるという実験を行った. 結果としては分割したデータ両方とも信頼度が 0.9 以上のとき 2 分割したデータを訓練データに加えると有効であることがわかった.

信頼度の閾値の設定や分割方法を変えて実験してみることが今後の課題である.

謝辞

最後に, 本研究を進めるにあたり, ご指導頂いた指導教員である新納教授に心より感謝の意を申し上げます. また, 多くの助言やご指摘を頂きました自然言語処理研究室の皆様に感謝申し上げます.

参考文献

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers forlanguage understanding, 2018.

[2] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, 2013.

[3] Xiaojin Zhu. Semi-supervised learning literature survey. Computer Science, University of Wisconsin-Madison, Vol.2, No.3, p.4, 2006.

[4] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In Pro-ceedings of the eleventh annual conference on Computational learning theory, pp.92 – 100. ACM, 1998.

[5] Jason Wei and Kai Zou.EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks,2019.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones,Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need,2017.