Contrastive Learningを利用した類似特許検索

星野雄毅[1*]　内海祥雅[2]　中田和秀[1]
[1] 東京工業大学工学院　[2] 楽天グループ株式会社・知的財産部
[*] 責任著者 : hoshino.y.ad@m.titech.ac.jp

概 要

近年，知的財産の管理は社会にとって重要となってきている．特に，特許は毎年30万件を超える出願があり，膨大な量の特許を処理する上で多くの課題が存在する．そこで，本研究では特許を扱う上で非常に重要な類似特許検索タスクについて，Contrastive Learningの応用を考えた．一方，特許情報の中で何を入力とするべきかについては，定かではない．また，Contrastive Learningを利用するにあたって，教師データに何を用いるかについては，いまだ研究がなされていない．そこで，本稿では３つの工夫を用いて，類似特許検索を実施した．まず，入力方法について，請求項全文を入力することを提案し，トークナイザー及びエンコーダを全て自作した．次に，Contrastive Learningを実施する教師データについて，引用情報を用いることを提案した．最後に，Contrastive Learningを実施する上でのHard NegativeについてIPCを用いた作成方法を提案した．さらに，実際の特許データを用いて２つの検証を行った．まず，特許の審査の際に用いられた引用情報を用いた数値実験を行いその効果を検証した．さらに，無効審判請求の事例をいくつか用いて，実際に運用した際の結果について検証を行った．

Keywords— Natural Language Processing, Patent, Contrastive Learning

1　序論

1.1　はじめに

特許とは知的財産の一つで，発明を保護するものである．特許は，発明における利益を発明者に還元するために必要であり，社会にとって重要な役割を果たしている．このような特許を活用していく上で，類似特許の検索は特許庁と企業の双方にとって不可欠である．まず，特許庁は特許審査の際に類似特許を検索し，それらとの差分を確認することで，特許を認めるかどうかを判断している．また，各企業にとっても，自社特許が他社のものに比べて何が優位なのか調査することで，特許出願の際の書き方や企業戦略にも影響を与える．したがって，類似特許は特許管理に関する様々な業務に応用できる非常に需要が大きなタスクである．

一方，近年深層学習を中心とした機械学習技術の進歩により，様々な分野で機械学習を応用した業務の効率化が期待されている．特に自然言語分野では，各文の埋め込みベクトルを獲得し様々なタスクに応用する研究が行われており，その手法の一つとしてContrastive Learningが挙げられる．そこで，本研究ではContrastive Learningを利用した，特許の表現ベクトルの作成を行い，これを類似特許検索に活用することを考える．

本稿の貢献は以下のとおりである．

1. 特許の入力方法について全請求項全文を入力することが精度を上げるために重要であることを示し，入力情報をできるだけ増やすためのトークナイズ方法の工夫について提案した．さらに，トークナイザー変更に当たって特許データを利用した事前学習を実施した．
2. 類似特許の教師データとして引用情報を用いてContrastive Learningを実施することを提案した．
3. Contrastive Learningを行うにあたって，IPCを用いてHard Negativeを作成することとその際のサンプリング方法について提案した．
4. 上記提案について実特許データを用いて定量的に評価した．

5. 無効審判請求された二つの事例を用いて，現状のモデルの良い点と問題点について定性的に評価した．

1.2 本論文の構成

本論文の構成は以下のようになっている．まず, 2 節で研究背景として，特許の概要や IPC について紹介したのち，類似特許検索の重要性について解説する．3 節では，関連研究として日本語の自然言語処理においてよく用いられる手法について紹介したのち，Contrastive Learning について説明する．4 節では提案手法として，入力情報，教師データ，Hard Negative のサンプリングについて説明する．続いて 5 節では，提案手法の有効性について実験を行い，定量的に評価する．さらに，6 節では実際に無効審判請求をされた２つの事例を用いて，提案手法がどの程度妥当なのかやその問題点について定性的に検証する．最後に本研究のまとめと今後の課題について述べる．

2 研究背景

2.1 特許とは

特許とは，知的財産の一つである．特許には様々な情報が含まれており，公開公報（登録公報）だけでも以下のものが挙げられる．

1. 公報種別
2. 出願番号，公開番号,（特許番号）
3. 特許分類
4. 出願日，公開日,（登録日）
5. 出願人，発明者,（特許権者）
6. 出願番号，公開番号,（特許番号）
7. 発明の名称
8. 要約
9. 特許請求の範囲
10. 発明の詳細の説明
11. 実施例
12. 図面の簡単な説明
13. 図面

このように，出願日や特許分類などの基本情報のほかに，特許文書の内容を示す項がいくつか存在する．本論文では特に「特許請求の範囲」の部分について詳しく説明する．特許法には,「第二項の特許請求の範囲には，請求項に区分して，各請求項ごとに特許出願人が特許を受けようとする発明を特定するために必要と認める事項のすべてを記載しなければならない．この場合において，一の請求項に係る発明と他の請求項に係る発明とが同一である記載となることを妨げない.」（第三十六条五項）とある．つまり,「特許請求の範囲」は権利を請求する具体的な範囲が記されている部分であるといえるため，特許の内容について調べる場合に中心的に見る必要のある項目であるといえる．

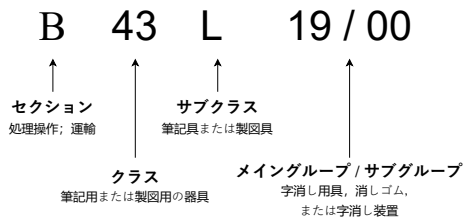

図 1: IPC の例

ここで,「特許請求の範囲」の特徴として複数の請求項に分けられていることがあげられる．例えば，消しゴム付き鉛筆を発明したときの請求項例は以下のようになる．

【特許請求の範囲】

【請求項 1】 鉛筆

【請求項 2】 本体の断面が多角形の請求項１に記載の鉛筆

【請求項 3】 本体の断面が六角形の請求項２に記載の鉛筆

【請求項 4】 本体に小型消しゴムを装着した請求項１～３のいずれか１項に記載の鉛筆

ここで請求項は大きく分けて，独立請求項と従属請求項と呼ばれる２つに分類される．まず，独立請求項とは請求項１のようにどの請求項も修飾しない請求項のことである．一方従属請求項とは請求項２の「請求項１に記載の」のように他の請求項に補足的な情報を加える形で書かれている請求項である．このように請求項に従属関係を持たせることで，例えば請求項２で権利を得られなかったとしても権利範囲を限定した請求項３によって権利を取得できる可能性を残せるようにしている．また，付録 A に乗せたように実際の特許を見ると，権利請求範囲を狭めないようにするために，表現が回りくどく文長が長いことも特徴である．

2.2 IPC とは

国際特許分類 (以下 IPC) とは国際的に統一された分類である．IPC は，技術分野によって与えられ，先行研究の検索などを行う際に用いられる．ここで，IPC には特徴が二つ存在し，階層構造を持っていることと，一つの特許に複数与えられることである．

IPC は「セクション」,「クラス」,「サブクラス」,「メイングループ」,「サブグループ」の 5 つの要素から構成されており，階層構造を持っている．例えば図 1 の場合，セクションに当たる「B」は「処理操作」という大まかな分類

を表している．次にクラスの「43」はセクションの「B」と合わせてより細かい「筆記用または製図用の器具」を表し，最後の「サブグループ」まで合わせると,「字消し用具，消しゴム，または字消し装置」という細かい分類にまで分けられる．このように，階層構造を持っていることで，様々な粒度で分野の絞り込みを行える．IPC は全部で約 7 万通り存在しており，多種多様な分野を表現可能である．

次に，IPC は一つの特許に複数与えられる．例えば，前述した消しゴム付き鉛筆の例で考えると，鉛筆に関係する IPC と消しゴムに関する IPC の両方が付与されることとなる．IPC は一つの特許に平均 3 件程度，多いものでは 70 を超える IPC が付与されている．

2.3 類似特許検索

類似特許検索は特許庁と申告する企業の双方に需要のあるタスクである．まず，特許庁にとっては審査の精度向上と効率化という２つの点から重要であるといえる．審査の際には，審査官が出願された特許に対して先行技術を調査し，比較した上で，新規性と進歩性の両方の観点から違いを検討することになる．ここで，現在先行技術調査は人手で行われており，特定の単語が含まれているかや，IPC などの分類を用いて絞り込みながら行っている．例えば，消しゴム付き鉛筆の例であれば,「鉛筆」や「消しゴム」だけでなく，その上位概念にあたる「筆記具」などを候補としながら検索を進めたり,「B43L19/00」という IPC が含まれているかどうかなども含めて様々な条件を組み合わせて検索を行うことになる．数百万件の特許の中から指定の分野に似た特許を調べ上げるのは大変である．したがって，特許審査の精度向上は非常に重要な課題であるといえる．

一方で，特許を調べる効率も非常に重要な課題である．特許は国内だけでも年間 30 万件も出願されており，これらをすべて人手で処理するには膨大な時間がかかることがわかる．実際，特許の審査には平均 10ヶ月程度かかっており，権利化までは１年以上かかることが多い．権利化されるまでは，特許として権利を行使することができないため早く権利化を行うことは重要な課題とされている．

また，類似特許検索は各技術を有する企業も実施している．前述したとおり，特許の審査は現在確実であると言えず，調査漏れなどを原因として本来認められないような特許も登録されてしまう危険性がある．そのため，各企業も自社特許が侵害されていないかどうかを監視しておく必要がある．また，企業が特許を申請する際も先行研究を調査したうえで，どこに自社技術の新規性があるのかなどを差別化する必要がある．したがって，企業にとっても類似特許検索は重要なタスクであるといえる．

ここで，複数 IPC を与えられる場合，その全ての IPC における観点から類似特許を検索する必要がある．例えば，消しゴム付き鉛筆の特許の場合を考えると，多角形の鉛筆という鉛筆分野の技術を調査すると同時に，消しゴムをその他の文具につけるという消しゴムに関する技術についても調査するべきである．このように，IPC が複数与えられるような，複数の分野にまたがった技術の場合は，すべての IPC からの観点から似ている特許を調べ上げることが重要である．

以上より，類似特許検索は企業にとっても，特許庁にとっても重要なタスクであり，これを補助するツールの作成は有意義である．

3 関連研究

3.1 日本語のトークナイズ

深層学習モデルを用いて自然言語を扱う上ではトークナイズしていくつかの文字をまとめて入力することが多い．中でも，日本語のトークナイズ方法は大きく分けると，形態素解析を行うものと，モデルベースで分割を行うものの２つが存在する．

まず，形態素解析とは，文を品詞に分解することである．日本語の形態素解析ツールとしては，MeCab [25] や Sudachi [28] など複数存在している．これらの手法はいずれも，単語とその品詞の組み合わせを記述した辞書と，すでに分かち書きされた文をもとに作成される．この手法のメリットとしては，日本語の文法に従ってトークナイズすることができることにある．一方でデメリットとして単語の辞書と分かち書きされている文の両方が必要な点が挙げられる．単語の辞書はその品詞も含めて記述する必要があるため，それらを準備するのは大変である．一般に公開されている辞書も存在するが，これを利用するとそこに含まれていない単語を識別できないため，特殊な単語が存在する場合には不適切な場合がある．また，単語の辞書を作成したとしても，それを用いて分かち書きされた文を多く用意する必要があるため，実際に自作の辞書に対して学習することは現実的でない．

もう一つのトークナイズ方法としてモデルベースの手法が挙げられる．こちらは文法などは考えず，意味の一定の文字のかたまりに分解する方法であり，BytePairEncoding [26] や UnigramLanguageModeling [15] などが提案されている．まる．これらの手法のメリットとして，教師データが必要ないため，対象の文データのみからトークナイズ方法を学習できることが挙げられる．特に，対象のデータの単語が一般の単語と比べて偏っていたり，特殊な単語が使われているような場合にも，頻出のまとまりは抽出できるため有効であると考えられる．また，もう一つ大きな特徴として比較的各文のトークン数を少なく表現できることが挙げられる．なぜなら，本来多くの単語からなるような頻出の表現を１トークンとしてまとめて表現できるためである．これは，長文を深層学習モデルで学習する上で重要となる特徴である．

3.2 深層学習モデル

3.2.1 Transformer

近年，自然言語処理の分野では深層学習のモデルが主に用いられている．その中でも，Transformer [29] をベースとした手法が最も精度が高いとされている．Transformer とは，MultiHeadAttention 層と FeedFoward 層を繰り返し用いるような方法で，FeedForward 層は 2 層の Affine 変換と活性化関数からなる．特徴的なのは MultiHeadAttention 層で，これは Attention 機構を分割して行うような手法で，単語間の関係性を考慮する層になっている．

自然言語処理や画像処理などに深層学習を適応する際には，多くの場合事前学習が実行される．事前学習とは，大量のコーパスからデータそのものの特徴を捉えるために学習をすることで，様々な方法が提案されている．

自然言語処理の事前学習の中で最もベーシックな方法として，BERT [7] が挙げられる．BERT とは，Masked Language Model (MLM) と Next Sentence Prediction (NSP) という 2 つのタスクを用いて，Transformer を学習するものである．MLM とは，文中のランダムな単語をマスクし，マスクした単語を予測するというタスクである．一方，NSP とは，名前の通り 2 つの文のペアに対して，それが次の文であるかどうかを推定するモデルである．ただし，NSP は導入することでむしろ精度が下がるという結果も後の研究 [16] で指摘されており，行うかどうかは意見が分かれている．BERT では，出力は $L \times h$ 行列となるが，実際にはその先頭のトークンを文書ベクトルとみなすことが多い．

近年では，他の事前学習方法についても研究が進んでいる．例えば，RoBERTa [20] は BERT の MLM のマスクを動的にすることで精度をわずかに改善している．また，ALBERT [16] では，BERT の NSP を導入すると精度が下がるという問題について言及したうえで，NSP の代わりに Sentence Order Prediction と呼ばれる文の順序が入れ替わっているかを判定するタスクを導入した．さらに，ELECTRA [6] は MLM で Mask する代わりに別の単語で置き換え，置き換えられているかを予測するという Replaced Token Detection タスクを導入し，より少ないステップ数で同程度の精度を出せることを示した．ただし，学習ステップを減らすことやモデルサイズを減らすことが目的であり，これらはいずれも予測精度に大きな影響は無かった．

3.3 Contrastive Learning

Contrasitve Learning とは，画像処理を中心に研究されていた分野である [4]．Contrastive Learning の最も特徴的な点としては，データ間の関連性から学習を進めることにある．つまり，いくつかの画像を入力とした際に，同じ画像か異なる画像かを判別するように学習を進めていく．一方で，ミニバッチ内のデータのみをサンプリングして異なる画像として学習する場合では，学習が進みにくいという問題がある．そこで，場合によっては Hard Negative として予測が難しいであろう負例を考える場合がある．これによって，常に予測難易度が高いタスクを学習することになり，学習効率を上げることができる．

SimCSE は Contrastive Learning を自然言語に応用する方法である [9]．具体的には，意味的な従属関係のある文を近くに，そうでないものを遠くに学習するものである．例えば,「2 匹の犬が走っている」 (“Two dogs are running.”) という文に対して，意味的な従属関係がある文とは「外に動物がいる」 (“There are animals outdoors.”) といった文が対応する．ここで，教師有 SimCSE の特徴として，Hard Negative には意味的に排反なデータ [3] を使用していることが挙げられる．排反な意味のデータとは,「ペットがソファに座っている」 (“The pets are sitting on a couch.”) というように，同時に起こりえない 2 つの事象のことである．このような排反な意味のペアをハードネガティブとして用いることで学習効率を上げている．

3.4 機械学習の特許への応用

機械学習を特許に応用する研究は様々存在している．まず，Fall らは，文章を単語の集合とみなしたうえで，SVM [27] や k-NN [23] といった古典的な機械学習手法によって IPC 分類の一部を予測した [8]．また，Grawe らは，Word2Vec を用いた特許データの分類も行った [10]．Aras らは，同様に Word2Vec [22] を用いて，自作でラベリングした類似特許データに対してその類似度を比較した [2]．

深層学習技術が特許に応用され始めたのは 2018 年頃からであり，特に分類タスクでいくつかの研究がされている．まず，Li らは CNN をベースとしたモデルで IPC のサブクラスを予測した [19]．Lee らは BERT を特許データでファインチューニングし，IPC のサブクラス予測を行った [18]．Risch らは IPC の予測を少し工夫し，文生成タスクとしてモデル化，予測を行った [24]．さらに，Hoshino らは，入力方法とデコーダの階層構造を工夫することで，IPC の予測精度をさらに改善させた [11]．Kim らはオートエンコーダを利用し特許の埋め込みベクトルを作成したうえで，クラスタリングを実施した [14]．Lee らは，特許文を生成するという目的で深層学習を利用し，Transformer ベースのモデルでその可能性を検証した [17]．

深層学習技術によって関連特許を検索する研究も行われている．Abood と Feltenberger は，筆者グループがラベル付けしたデータを用いて，LSTM ベースのモデルを学習し，その精度や人手による評価を実施した [1]．Choi らはこのモデルについてグラフ埋め込みの適応や Transformer モデルの使用により精度を向上させた [5]．このように関連特許を検索する研究はいくつか行われている．しかし，これは一部の自作データセットによってのみ行われており，これらの日本語データは存在しない．したがって，単純にこれらの手法を利用することはできない．さらに．入力として特許の「要約」部分のみを用いている点も問題であり，改善が望まれる．

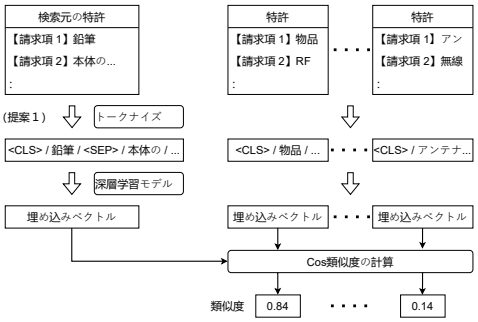

図 2: 提案手法の概要

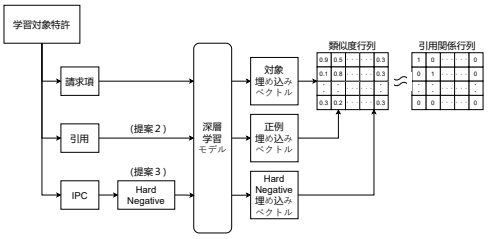

図 3: 提案手法の学習方法の概要

4 提案手法

4.1 提案フレームワーク

提案する，類似特許を検索する際のイメージは図 2 のようになっている．まず，検索したい特許について，請求項を取り出し，トークナイザーによって文を分割する (提案 1)．次に，分割した文を深層学習モデルに入力し，埋め込みベクトルを得る．最後に，作成した埋め込みベクトルと，ほかの特許について同様の方法で作成した埋め込みベクトルとの類似度を計算し，上位のものを類似特許とみなす．

また，推論に用いる深層学習モデルの学習方法のイメージは図 3 のようになっている．まず，学習対象の特許の請求項と，引用による正例 (提案 2)，IPC による Hard Negative(提案 3) の 3 つを深層学習モデルに入力し，それぞれの埋め込みベクトルを作成する．そのうえで，得られた埋め込みベクトルから類似度行列を計算し，引用関係を表した引用行列を正解データとして，Contrastive Learning の損失を与え，学習する．

ここからは提案手法の詳細について，請求項全文の入力 (提案 1)，引用情報の利用 (提案 2)，IPC を用いた Hard Negative(提案 3) の三つに分けて順に説明する．

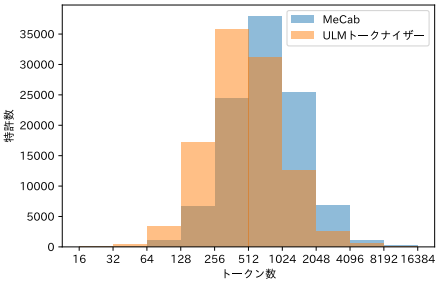

図 4: トークナイズ方法によるトークン数の比較

4.2 請求項全文の入力

既存の研究では，入力に請求項の全文を用いたものは少なかった．なぜなら，請求項の全文は言語を問わず長く，機械学習モデルに入力するのが厳しいためである．

一方で，日本語においてはトークナイズ方法を工夫することで，請求項をより長く入力することができることが期待される．これまで日本語で特許を扱った論文は少なく，主に英語を用いられていたため，トークナイズ方法については言及された研究はなかった．そこで，まず日本語特許の請求項を UnigramLanguageModel を用いてトークナイズし，そのトークン数を数えた．

トークン数のヒストグラムは図 4 のようになった．ただし，横軸が対数スケールであることに注意が必要である．図 4 を見ると，トークナイズ方法を MeCab から UnigramLanguageModel に変更することでトークン数を少なくできていることが分かる．実際トークン数の中央値は 733 程度から 467 程度まで減少しており，1024 トークンまでに収まる特許の割合も 68%程度から 85%程度まで増加した．そこで，UnigramLanguageModel を用いたトークナイズしたうえで深層学習モデルに入力することを提案する．

一方で，トークナイザーを新たに作成した場合，既存の事前学習済み BERT を利用することはできない．なぜなら，事前学習は，事前学習時に使用されたトークナイザーの関係性のみ学習しており，新たに与えられた未知の文字列に対してその構造をとらえられていないためである．そこで，自作のトークナイザーを用いて特許をトークナイズしたのち，事前学習を実施した．事前学習は 2014 年から 2016 年の特許全て（およそ 45 万件）の請求項を用いて実施した．請求項をできるだけ全て入力することを考え，モデルのハイパーパラメータは表 1 のように設定した．ただし，これは基本的に BERT の small のサイズに準拠しており，maxlength のみは特許に合わせて 2048 トークンとしている．また，学習は Tesla p100 の GPU(5.3TFlops) を用い，バッチサイズ 16 × 4 で 3 週間程度 (912000 ステップ) 学習を行った．学習曲線は図 5 のようになった．これは横軸がステップ数，縦軸が損失を表しており，収束していることがわかる．

表 1: 事前学習モデルの詳細

attention_probs_dropout_prob	0.0
hidden_dropout_prob	0.0
hidden_size	512
intermediate_size	2048
max_position_embeddings	2048
num_attention_heads	8
num_hidden_layers	4
vocab_size	16003

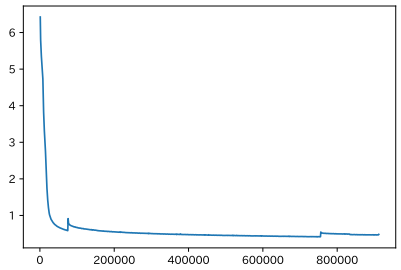

図 5: 事前学習の学習曲線

4.3 引用情報の利用

類似特許検索を行う上では，類似特許とみなせる教師データが決まっていないという問題が存在する．この,「類似特許」として適したデータとしては，無効審判請求及び，異議申し立てのデータである．これらは，実際に厳密に審査した結果，特許が無効になったものであり，その際に参照された特許は確実に類似特許とみなすことができる．一方で，これらのデータは学習データとして用いるにはデータ数が十分でないという課題がある．

そこで，本研究では特許の審査過程で援用された引用情報に注目し，引用されている特許を類似特許とみなすこととした．これは，類似した特許のすべてを網羅していないが，引用された特許は類似した特許であるとみなすことができるためである．さらに，ほとんどすべての特許について，引用情報は取得できることから，学習データとして十分な数存在しているといえる．したがって，引用情報を類似特許の正解データと仮定し，学習評価をすることを考える．

ここで，引用は各特許に対して複数存在していることに注意が必要である．SimCSE は正例が一つの場合を想定したアルゴリズムであったため，単純に応用することはできない．そこで，教師有 Contrastive Learning [13] のように，複数の正例から学習を行うような損失に従うサンプリング

Algorithm 1 特許データミニバッチ作成

Require: I, P, B

Ensure: $\text{batch}_i, \text{batch}_p$

1: $\text{batch}_i, \text{batch}_p \leftarrow [], []$
2: **for** $b \in B$ **do**
3: $\quad i \leftarrow I_b$
4: $\quad$ Choose p from $P(i)$ uniformly randomly
5: $\quad \text{batch}_i \leftarrow \text{concatenate}(\text{batch}_i, [i])$
6: $\quad \text{batch}_p \leftarrow \text{concatenate}(\text{batch}_p, [p])$
7: **end for**

方法で学習を行った．アルゴリズムは Algorithm 1 のようになっており，4 行目で引用特許の集合からランダムに選択をしている．ただし， B はミニバッチで学習したい特許の ID 集合， I_b は特許の ID b に対応する特許， $P(i)$ は特許 i に引用されている特許集合である．

4.4 IPC を用いた Hard Negative

Hard Negative を作成する目的としては，類似度の予測が難しい Negative データについて効率的に学習を行うことである．そこで，予測が難しい Negative として分野が同じものを取得することを考え，IPC を用いた Hard Negative のサンプリング方法を提案する．

ここで，サンプリングする手順として様々なものが考えられるが，最も単純なものは各特許に対して IPC を共有する特許を全て一覧として保持しておき，その中から一様ランダムにサンプリングしてくるというものである．しかし，この手法では二つ問題が存在する．まず，同じ IPC を共有する特許は実際には無数に存在しており，これらをすべての特許の組合せに対して記憶しておくことは，メモリの観点から現実的ではないことが挙げられる．そのため，よりデータの効率の良いサンプリング方法が望まれる．次に，多くの特許が出願されている分野が重点的にサンプリングされてしまい，様々な方面から似ている特許を取得しにくくなってしまうことが挙げられる．例えば，ある特許に A と B という二つの IPC が付与されており，他に A の IPC を持つ特許が 2 件しかないのに対して，B の IPC を持つ特許が 100 件存在したと仮定する．この時，単純なサンプリング方法ではほとんどの場合 B の特許しか取得することができなくなり，A 方面からの hard negative をほとんど再現することがなくなってしまう．そこで，IPC に対して同程度の割合でサンプリングすることが望ましいと考えられる．

そこで Algorithm2 のように，5 行目で各特許に付与されている IPC をランダムに選択し，6 行目で選択した IPC を持つ特許の中からさらにランダムに IPC を選択するという 2 段階のサンプリングアルゴリズムが考えられる．た

Algorithm 2 単一階層を用いた IPC による Hard Negative を加えたサンプリング

Require: I, P, B, N

Ensure: $\text{batch}_i, \text{batch}_p, \text{batch}_n$

$\text{batch}_i, \text{batch}_p, \text{batch}_n \leftarrow [], [], []$
for $b \in B$ **do**
  $i \leftarrow I_b$
  Choose p from $P(i)$ uniformly randomly
  Choose ipc from $IPC(i)$ uniformly randomly
  Choose n from $N(ipc) \setminus (\{i\} \cup P(i))$ uniformly randomly
  $\text{batch}_i \leftarrow \text{concatenate}(\text{batch}_i, [i])$
  $\text{batch}_p \leftarrow \text{concatenate}(\text{batch}_p, [p])$
  $\text{batch}_n \leftarrow \text{concatenate}(\text{batch}_n, [n])$
end for

だし，$IPC(i)$ は特許 i に付与されている IPC の集合であり，$N(ipc)$ は ipc が付与されている特許集合である．この手法であれば，前述した問題点はどちらも解決される．まず，データサイズの問題については，各 IPC に対して所属する特許の情報を保持すればよいため，比較的少ない情報で学習できる．さらに，データ内の分野の偏りについても，あらかじめ IPC を同じ確率でサンプリングすることでバランスよく様々なジャンルの hard negative を取得することが可能である．

ここで，IPC は階層構造を持っているため，様々な粒度の分野情報を包含している．そのため，複数の階層に従ってサンプリングを実行することで，より良い Hard Negative を作成することができるのではないかと考えた．以上の考えのもと，提案するアルゴリズムの疑似コードは Algorithm 3 のようになっている．ただし，H はセクションやクラス，メイングループなどの対象としたい階層の集合で，$IPC_h(i)$ は階層 h における特許 i に付与されている IPC の集合である．IPC の持つデータが Algorithm3 のものと異なることに注意が必要である．負例は 5 から 7 行目で取得しており，それぞれ IPC の階層の選択，階層内の IPC の選択，特許の選択の 3 段階で実行している．このように確率的に様々な分野からのサンプリングを実行することで，確率的に様々な粒度の分野において近いかどうか判別できるようなモデルを作成可能である．

Algorithm 3 複数階層を用いた IPC による Hard Negative を加えたサンプリング

Require: I, P, B, N, H

Ensure: $\text{batch}_i, \text{batch}_p, \text{batch}_n$

$\text{batch}_i, \text{batch}_p, \text{batch}_n \leftarrow [], [], []$
for $b \in B$ **do**
  $i \leftarrow I_b$
  Choose p from $P(i)$ uniformly randomly
  Choose h from H uniformly randomly
  Choose ipc from $IPC_h(i)$ uniformly randomly
  Choose n from $N(ipc) \setminus (\{i\} \cup P(i))$ uniformly randomly
  $\text{batch}_i \leftarrow \text{concatenate}(\text{batch}_i, [i])$
  $\text{batch}_p \leftarrow \text{concatenate}(\text{batch}_p, [p])$
  $\text{batch}_n \leftarrow \text{concatenate}(\text{batch}_n, [n])$
end for

5 数値実験

5.1 実験設定

5.1.1 データセット

今回データセットは有料の特許データ取得サービスを通じて入手した．入手手順としては，学習用データは 2016/07/01~2016/12/31 に出願された特許（104,078 件），テストデータは 2017/01/01~2017/01/15(4,348 件) をすべて取得し，それぞれこれらを検索元データとした．次に，検索元データの中で引用されていた特許をすべて取得し，検索対象データとした．検証用データは学習用の検索元データをランダムに 8:2 で分割して作成した．使用データの詳細は表 2 のようになっている．

5.1.2 学習設定

ファインチューニングは，バッチサイズ 8 で 5 万ステップ実施した．ファインチューニングの設定は表 3 のように設定した．特に，ハードネガティブの重みについては，いくつかの設定で行い，最も良かったものを採用している．

5.2 評価指標

本提案手法の目的は，類似度が高い特許を業務を行う人に推薦することである．そこで評価指標としては，類似度が高かった順に検索対象の特許を並び替えて推薦すること

表 2: データセットの基礎統計量

		学習用データ	テスト用データ
期間		2016/07/01~2016/12/31	2017/01/01~2017/01/15
件数	総数	104,078	4,348
	A	19,567	1,032
	B	26,592	995
	C	16,496	597
	D	1,435	49
	E	4,103	172
	F	12,434	467
	G	31,398	1,183
	H	32,295	1,350
平均文字数		1797.3	1587.9
平均引用数		2.60	3.02

表 3: ファインチューニングのハイパーパラメータ

埋め込み次元	512
最大トークン数	1024
バッチサイズ	8
ハードネガティブ重み (クラス)	e^{-5}
ハードネガティブ重み (サブクラス)	e^{-6}
ハードネガティブ重み (メイングループ)	e^{-7}
ハードネガティブ重み (サブグループ)	e^{-8}

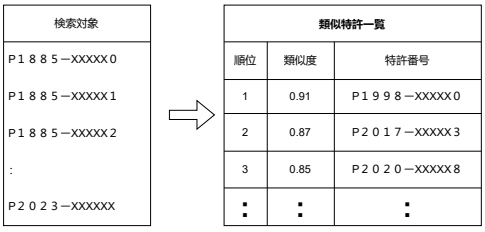

図 6: 類似度を推薦に使うイメージ

を考えた (図 6). そのうえで，一般に推薦システムで用いられる評価指標を用いた.

推薦システムとしての評価指標として，適合率 (Precision)，再現率 (Recall)，NDCG の 3 つの評価指標を用いた [12]. まず，Precision@k とはモデルが推定したスコアの上位 k 個のうち，どの程度を引用していたのかを表した値であり，Recall@k とは実際に引用されていた特許のうちどの程度が上位 k 個に含まれているのかあらわした値である (図 7).

最後に，NDCG@k とは DCG@k を正規化したものである．DCG@k とは以下の式で定義される.

$$DCG_k = \sum_{i=1}^{k} \frac{r_i}{\log_2(i+1)} \tag{1}$$

ここで，各 r_i は i 位と予想されたものに対する評価値を表しており，今回の問題設定では， i 番目の特許が引用されていたかどうかを示したバイナリの値となっている．DCG@k のイメージは図 8 のようになっている．これを，

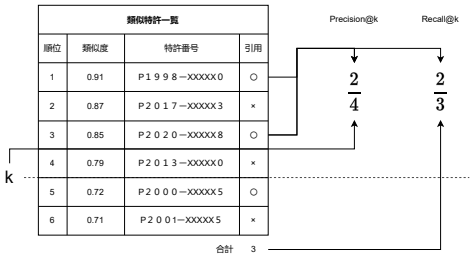

図 7: Precision@k と Recall@k のイメージ

類似特許一覧

順位	類似度	特許番号	引用	DCG@k
1	0.91	P1998−XXXXX0	○	$\frac{1}{\log_2 2}$
2	0.87	P2017−XXXXX3	×	$\frac{0}{\log_2 3}$
3	0.85	P2020−XXXXX8	○	$\frac{1}{\log_2 4}$
4	0.79	P2013−XXXXX0	×	$\frac{0}{\log_2 5}$
~~5~~	~~0.72~~	~~P2000−XXXXX5~~	~~○~~	SUM
~~6~~	~~0.71~~	~~P2001−XXXXX5~~	~~×~~	$\frac{3}{2}$

図 8: DCG@k のイメージ

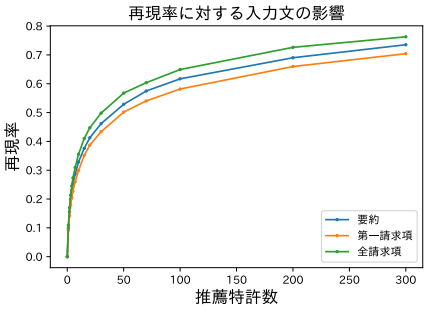

図 9: 再現率に対する入力文の影響

$\frac{1}{\log_2(i+1)}$ で重みづけて足し合わせているため，引用されていた特許が上位で予測されるほど高い値を示す．ただし，DCG@k は引用されている特許が多いほど高い値になってしまうため，これを最適な予測での DCG@k ($DCG_k^{perfect}$ と置く) で割ることで 0 から 1 に正規化したものである．

$$NDCG_k = \frac{DCG_k}{DCG_k^{perfect}} \tag{2}$$

最後に，NDCG を全体に対して計算したものを NDCG@inf としてあらわす．

5.3 全特許を対象とした分類検証

5.3.1 入力内容における検証

提案手法では，入力に請求項の全文を用いていたが，特許にはそのほかにもいくつか自然言語データが存在する．そこで，既存の特許に関する研究で良く用いられていた，「概要」と「第一請求項のみ」を入力とした場合を比較対象として，請求項全文を入力することの効果を検証した．そのために，tf-idf を用いた単純な類似度を用いて引用された特許を上位に選択できるのか検証した．

結果は表 4 のようになった．これを見ると，全請求項が最も良いことがわかった．まず，第一請求項を入力とする場合と比較すると改善しており，少なくとも類似特許を探すという目的においては，全文入力することの重要性がわかる．次に，概要と比較しても各評価指標が高いことから，概要だけでは情報が不足しているのではないかと考えられる．また，各特許数を推薦した時の再現率の変化は図 9 のようになった．これを見ると，再現率も同じ傾向がみられることがわかる．一方で，この手法では 10000 件のうち 300 件（3 パーセント程度）の特許を見たとしても，抽出したい特許の 8 割程度しか入手できておらず，十分でないことがわかる．

5.3.2 モデルに関する検証

提案手法は事前学習済み BERT をファインチューニングする形で使用している．しかし，まず，深層学習モデルを利用することが埋め込みとしてよいのかどうかを検証する必要がある．また，深層学習モデルを用いたとしてもファインチューニングするべきかどうかや，入力トークン数を減少させて汎用的なデータで学習された BERT を利用するべきなのかなどの検証も必要である．したがって，モデルや学習方法を変えた場合について比較を行った．

結果は表 5 のようになった．まず，tf-idf の類似度と比較して，事前学習済み BERT の埋め込みを用いた類似度はいずれの評価指標でもわずかに高くなっていることがわかる．更に，教師有学習を実行することで，いずれの評価指標でも大きく改善していることがわかる．一方で，教師無 SimCSE は大きく評価が落ちていることがわかり，学習を行わなかった BERT よりも低くなっている．推薦特許数を増やした場合の再現率は図 10 のようになった．これを見ると，教師無 SimCSE は著しく評価が悪い一方で，教師有 SimCSE モデルは全ての評価指標がさらに改善していることがわかる．また，BERT については，Wikipedia で学習された BERT [21] の埋め込みをそのまま用いたものはほとんど予測できていないことがわかる．これは，入力トークンが少ないために一部しか入力できていないものが多いことや，特許には特有の表現が存在していることなどが原因として考えられる．

5.3.3 Hard Negative を用いたサンプリングに関する検証

最後に，hard negative の導入の有無と，その導入方法について比較を行った．比較対象としては，Hard Negative を作成しなかったもの (Algorithm1) と，各特許の階層を絞って Hard Negative を作成したもの (Algorithm2)，そして提案手法の階層の深さを変化させたもの (Algotithm3) を用いた．ただし，Algorithm2 と Algotithm3 については，IPC のどの階層で行うかについていくつか設定が考えられるため，複数のパターンについて実験を行った．また，全く異

表 4: 入力文による各種評価指標の変化

	Precision			Recall			NDCG		
	@1	@5	@10	@1	@5	@10	@5	@10	i@nf
要約	0.220	0.122	0.082	0.099	0.253	0.329	0.228	0.258	0.388
第一請求項	0.208	0.113	0.076	0.093	0.228	0.299	0.209	0.237	0.370
全請求項	**0.237**	**0.131**	**0.088**	**0.109**	**0.273**	**0.355**	**0.246**	**0.278**	**0.407**

表 5: 学習モデルによる比較

	Precision			Recall			NDCG		
	@1	@5	@10	@1	@5	@10	@5	@10	@inf
tf-idf	0.237	0.131	0.088	0.109	0.273	0.355	0.246	0.278	0.407
BERT(特許)	0.257	0.134	0.090	0.114	0.274	0.357	0.253	0.284	0.418
教師無 SimCSE	0.018	0.010	0.007	0.005	0.016	0.024	0.015	0.018	0.146
教師有 SimCSE	**0.352**	**0.213**	**0.142**	**0.164**	**0.438**	**0.564**	**0.390**	**0.439**	**0.550**
BERT$_{\text{(cl-tohoku [21])}}$	0.019	0.044	0.058	0.047	0.022	0.015	0.042	0.047	0.175

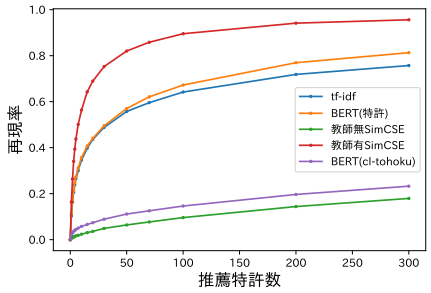

図 10: モデルによる再現率への影響

なる Hard Negative の作成方法として引用特許内で引用されている特許をランダムに選択するものも比較した．各モデル番号とその設定の一覧は以下のとおりに設定した．

1： Algorithm1 Hard Negative 無
2-1： Algorithm2 クラスのみ Hard Negative
2-2： Algorithm2 サブクラスのみ Hard Negative
2-3： Algorithm2 メイングループのみ Hard Negative
2-4： Algorithm2 サブグループのみ Hard Negative
3-1： Algorithm3 クラス，サブクラスの Hard Negative
3-2： Algorithm3 クラス～メイングループの Hard Negative
3-3： Algorithm3 クラス～サブグループの Hard Negative
4： Algorithm4 引用の引用 Hard Negative

結果は，表 6 のようになった．まず，Hard Negative を設定していないモデル 1 と比較して，Hard Negative を設定したモデル 1 以外のモデルは一部は評価が改善している一方で，悪化しているものもある (2-2, 2-3, 2-4, 4)．したがって，Hard Negative は入れたら良いわけではなく，適切な難易度のものを入れる必要があることがわかる．特に，引用の引用やサブグループのような，非常に近い内容の特許を単体で Hard Negative として指定すると評価指標が悪化している．一方，algorithm3 を用いて学習をするとメイングループなどの比較的階層が深く，類似度が高いと想定される特許を入力しても評価指標が向上することがわかる．したがって，Algorithm3 を用いた Hard Negative の作成は効果があると考えられる．また，Algorithm3 の中でもどの階層を入力するのかについては，サブクラスかメイングループまで取得するのが良いという結果となった．特に，最上位の 1 つを取得するようなケースでは，メイングループまで含めたものが適しており，一方で多くの特許を評価対象とする場合はクラスまでで抑えるべきであるという結果となった．これは，より上位のみの評価指標を上げるためには，類似度が高い中での識別を学習するためにより難しい Hard Negative を用意するべきであるという直感に適しており，納得感のある結果となっている．また，再現率をプロットした結果は図 11 のようになった．これを見ると，Hard Negative による影響は，これまでのモデルや入力方法ほどの差が生まれず，重要度は高くないことがわかる．

表 6: Hard Negative のサンプリング方法による比較

	Precision			Recall			NDCG		
	@1	@5	@10	@1	@5	@10	@5	@10	@inf
1	0.352	0.213	0.142	0.164	0.438	0.564	0.390	0.439	0.550
2-1	0.358	0.218	0.144	0.163	0.448	0.573	0.397	0.446	0.553
2-2	0.354	0.213	0.142	0.162	0.436	0.568	0.388	0.439	0.547
2-3	0.345	0.210	0.142	0.158	0.433	0.566	0.384	0.436	0.545
2-4	0.328	0.202	0.138	0.148	0.411	0.548	0.364	0.418	0.530
3-1	0.358	**0.218**	**0.146**	0.167	**0.451**	**0.583**	**0.400**	**0.452**	**0.556**
3-2	**0.361**	0.215	0.145	**0.168**	0.445	0.572	0.396	0.447	0.555
3-3	0.354	0.214	0.145	0.162	0.443	0.579	0.391	0.445	0.551
4	0.350	0.207	0.137	0.161	0.427	0.549	0.381	0.428	0.541

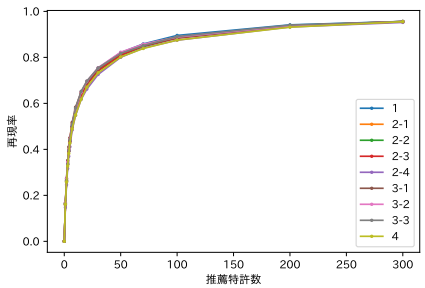

図 11: Hard Negative のサンプリング方法による推薦数に対する再現率の推移

5.4 分野を絞った検証

これまでは，特許庁などの全分野において検索を行う必要がある場合を想定したが，特定の分野のみの研究を調べればよいような企業が用いる場合は，分野を絞って学習を行う方が良いかもしれない．そこで，分野を絞って学習を行った場合と，全分野まとめて学習を行った場合で各評価指標がどのように変化するのか検証した．同時に，分野によって特許の表現は異なっており，予測の難しさも異なると考えられる．したがって，分野による評価指標の違いを比較した．

結果は表 7 のようになった．ただし，上段が分野ごとに学習したモデル，下段が全体で学習をして評価のみ分野ごとに行ったモデルの評価指標をそれぞれ表している．これを見ると，特定の分野に絞った学習を行うことは，ほとんどの分野において予測に良い効果をもたらさないことがわかった．特に，D(繊維) のようなデータが少ない分野において大きくすべての評価指標が悪化している．これは，予測対象の分野を絞ったとしても，引用する対象の分野は必ずしも絞れず，いずれにしても全分野に対する特許に関する表現ベクトルを獲得しなければならないためであると考えられる．例えば，鉛筆に関する特許について，重点的に学習したとしても，消しゴム付き鉛筆の特許を出願するためには，消しゴムに関しても学習する必要がある．更に，鉛筆とは関係のないことを学習する必要もあったために，このような分野を絞った学習に効果が現れなかったと考えられる．

5.5 数値実験のまとめ

本数値実験を通してわかったこととしては，4 つのことがわかった．まず，特許情報の言語情報の中で，請求項全文を入力することが比較的類似度を図る上で良いことがわかった．次に，モデルと学習方法について，教師あり SimCSE を応用するものの評価指標が高かった．更に，Hard Negative について，複数階層を考慮した IPC を用いたサンプリングが最も各評価指標が高い一方で，クラスとサブクラスのみを用いたものが最も良いことがわかった．最後に，分野を絞った学習についてはあまり各種評価指標は改善せず，全体で学習したほうが良いことがわかった．一方で，Recall についてみてみると，現在の予測では Recall@100 でおよそ 9 割程度であった．したがって，実際に引用されていた特許の 9 割をこのモデルを使って推薦するには，全特許のうち 1%程度を見る必要があることになる．

6 無効審判例を用いた定性評価

5 節では，比較手法と比較して，提案手法が最も優れていることを示した．一方で，実際に無効審判などで扱われ

表 7: 分野ごとの各評価指標

セクション	Precision			Recall			NDCG		
	@1	@5	@10	@1	@5	@10	@5	@10	@inf
A	**0.334**	0.169	0.112	0.176	0.394	0.496	0.354	0.395	0.507
(生活必需品)	0.332	**0.182**	**0.117**	**0.184**	**0.414**	**0.511**	**0.370**	**0.407**	**0.524**
B	0.368	**0.238**	**0.156**	0.170	**0.483**	**0.607**	**0.426**	**0.474**	0.573
(処理操作)	**0.394**	0.232	0.155	**0.176**	0.465	0.594	0.420	0.470	**0.576**
C	0.333	0.220	0.146	0.146	0.445	0.567	0.388	0.436	0.543
(化学)	**0.357**	**0.220**	**0.150**	**0.170**	**0.458**	**0.595**	**0.404**	**0.459**	**0.565**
D	0.281	0.144	0.091	0.136	0.346	0.465	0.303	0.345	0.449
(繊維)	**0.406**	**0.256**	**0.159**	**0.199**	**0.629**	**0.746**	**0.534**	**0.585**	**0.645**
E	0.345	0.222	0.144	0.146	0.450	0.590	0.394	0.447	0.529
(固定構造物)	**0.418**	**0.256**	**0.163**	**0.190**	**0.555**	**0.675**	**0.491**	**0.540**	**0.621**
F	0.399	0.232	0.149	0.190	0.484	0.586	0.434	0.476	0.576
(機械工学)	**0.424**	**0.242**	**0.163**	**0.203**	**0.506**	**0.646**	**0.454**	**0.511**	**0.601**
G	0.340	0.201	0.140	0.141	0.387	0.524	0.351	0.405	0.522
(物理学)	**0.351**	**0.208**	**0.142**	**0.150**	**0.400**	**0.525**	**0.365**	**0.415**	**0.534**
H	0.336	0.206	0.142	0.150	0.421	0.561	0.370	0.426	0.536
(電気)	**0.353**	**0.215**	**0.146**	**0.157**	**0.439**	**0.572**	**0.387**	**0.441**	**0.550**

た特許について，どのような場合の時に正確で，どのような場合の時に不正確なのかについてはわからなかった．そこで，本節では提案手法がどのような場合で予測できているのかを，実際に無効審判請求をされた特許を用いて定性的に検証した．

6.1 事例 1．入出力モジュール

6.1.1 事例内容

本事例は，横河電機株式会社が保持している特許 (特許 5565623 号) について，無効審判請求がされたものである．特許 5565623 号の請求項は以下の通りである．

特許 5565623 号の請求項の一部

【請求項１】複数の入出力信号処理系統が設けられた入出力モジュールにおいて、前記モジュールの機能を定義するユーザインタフェースを設け、前記モジュールの機能は、モジュール単位または複数の入出力信号処理系統について個別に定義し、前記ユーザインタフェースは、アナログ出力、アナログ入力、ＯＮ／ＯＦＦのロジック信号、パルスカウンタ、パルス出力、通信を含む機能のうち、少なくとも２つの機能を組み合わせたことを特徴とする入出力モジュール。

請求人の主張の概要としては，無効主張を主張する証拠として甲第１号証：特開 2008-287618 号公報及び甲第２号証：特開平 2-87204 号公報を提示しており，特に特開 2008-287618 は内容が近いことがうかがえる．したがって，特開 2008-287618 が最も類似した特許として抽出され，特開平 2-87204 も上位に来るという結果が望ましいといえる．特開 2008-287618 及び，特開平 2-87204 の請求項は以下の通りである．

特開 2008-287618 の請求項

【請求項 1】複数の外部配線用の接続端子を備えたプログラマブルコントローラにおいて、直流出力回路と、直流入力回路と、交流出力回路と、交流入力回路と、を有すると共に、ユーザの選択操作により、上記接続端子それぞれを上記回路のいずれかに切り替え接続可能として、直流出力仕様、直流入力仕様、交流出力仕様、交流入力仕様のうちのいずれか１つの仕様または複数の仕様に、設定可能とした、ことを特徴とするプログラマブルコントローラ。

【請求項 2】上記接続端子それぞれに対応配列された表示部と、各表示部を対応する接続端子それぞれに割り当てられた仕様に対応した表示色に駆動制御する制御部と、を有することを特徴とする請求項１に記載のプログラマブルコントローラ。

【請求項 3】ＣＰＵモジュールと１ないし複数の周辺

モジュールとからなるプログラマブルコントローラにおいて、上記周辺モジュールの少なくとも１つに、直流出力回路と、直流入力回路と、交流出力回路と、交流入力回路と、を有すると共に、ユーザの選択操作により、複数の外部配線用の接続端子それぞれを上記回路のいずれかに切り替え接続可能として、直流出力仕様、直流入力仕様、交流出力仕様、交流入力仕様のうちのいずれか１つの仕様または複数の仕様に、設定可能としたモジュールを含む、ことを特徴とするプログラマブルコントローラ。

特開平 2-87204 の請求項

【請求項 1】ディジタル計算機への入出力ゲートと、入出力制御回路と、プロセスへの入出力共通コネクタと、プロセス入出力の種別選定回路さらに複数種のプロセス入出力変換回路を有し、前記種別選定回路の設定を自由に行える構成とし、該設定により特定のプロセス入出力インターフェースとして働く構成としたことを特徴とするプロセス入出力装置。

【請求項 2】前記プロセス入出力インターフェースに多種類の入出力機能を含有させ、前記種別選定回路によりプロセスに合つた入出力機能を選択して使えるようにした特許請求の範囲第１項記載のプロセス入出力装置。

【請求項 3】プロセス入出力の種別をＣＰＵ側から設定できるようにした特許請求の範囲第１項記載のプロセス入出力装置。

【請求項 4】多種類の信号変換回路を含有し、選択信号を外部より与え、特定の変換回路のみを動作させるようにした特許請求の範囲第１項記載のプロセス入出力装置。

ここで，特許 5565623 号と特開 2008-287618 の比較をしたときに，明らかに類似しているとは言えないと考えられる．なぜなら，用いられている単語がそれぞれ異なっているからである．例えば，特許 5565623 号では「インターフェース」という単語が多く出現するのに対して，特開 2008-287618 では出現しない．したがって，分野に詳しい人であれば理解できるものの，簡単には類似度が高いといえない例だと考えられる．

一方で，特開平 2-87204 は一見すると特許 5565623 号と類似度が非常に高そうに見える．なぜなら,「インターフェース」や「入出力」といった単語が双方で使われているからである．これは，特開平 2-87204 と特許 5565623 号はどちらも横河電機株式会社が取得した特許であるため，表現が似たのではないかと考えられる．

検証は，検索対象のデータセットの中から，特開 2012-3799 と類似度が高い特許を抽出し，特開 2008-287618 及び特開平 2-87204 が上位に出現するのかを確認した．ただし，検索対象のデータセットは実験で用いた被引用データセットに対象の 2 件の特許を加えた 13125 件からなるデータセットとした．

6.1.2 検証結果

表 8: 事例 2 における類似度の高かった特許

順位	類似度	特許公開番号
1	0.842	**特開平 2-87204**
2	0.751	**特開 2008-287618**
3	0.746	特開 2003-115847
4	0.733	特開 2001-77880
5	0.714	特開 2014-160367
:	:	:

結果は表 8 のようになった．まず，特開 2008-287618 よりも特開平 2-87204 の方が上位に来ており，理想の結果とは異なるものであった．この原因は前述したとおり，特開平 2-87204 の方が似た単語，表現が用いられていることに起因すると考えられる．一方で，一見類似度が高くないように見えた特開 2008-287618 も 2 番目に類似度が高く表現できている．これは，単語レベルではなく，その意味も含めてモデルが学習できていることで得られたものだと考えられる．

また，引用されていない特許も一見関連がありそうな文献が抽出できており，他の類似度もある程度妥当なものを取得できていると考えられる．実際，五番目に類似度が高いとされた特開 2014-160367 の請求項は以下のとおりである．

特開 2014-160367 の請求項

【請求項 1】演算処理を行うＣＰＵ部と外部入出力機器との入出力信号の授受を行う入出力コントローラ部から構成され、前記ＣＰＵ部は複数のＣＰＵコアを備え、前記入出力コントローラ部は複数の入出力コントローラを備え、前記入出力コントローラは前記ＣＰＵ部に対して定周期で割り込み信号を発生し、前記割り込み信号に対して対応する前記入出力コントローラの前記入出力信号の処理を行う前記ＣＰＵコアを予め設定しておく構成の演算処理装置。

【請求項 2】前記割り込み信号に対して前記入出力信号の処理を行う前記ＣＰＵコアとして一部の特定の前記ＣＰＵコアを割り当てる構成とした請求項１に記載の演算処理装置。

【請求項 3】前記ＣＰＵコアは、複数の前記入出力コントローラからの前記割り込み信号に対して前記入出力信号の処理を行う構成において、さらに割り込みタイミング制御部を設け、前記割り込みタイミング制御部は、前記ＣＰＵコアが前記入出力コントローラの前記割り込み信号に対して前記入出力信号の処理を実施している時に他の前記入出力コントローラの前記割り込み信号が発生した場合には、前記ＣＰＵコアへの他の前記入出力コントローラからの前記

割り込み信号の出力を一定時間遅らせる制御を行う構成とした請求項２に記載の演算処理装置。

これを見ると,「信号」や「入出力」といった単語が多く使われていることがわかる．したがって，知識のない一般的な人から見たとき，特開 2014-160367 は特許 5565623 号との類似点もあるように見える．一方で，前述したとおり特開 2008-287618 は一般的な人が見たときに類似度がそれほど高くないと考えるため，特開 2014-160367 と比較して特開 2008-287618 が高く表現されていることは，本事例においてはモデルが専門的な意味まで学習できていると考えられる．

6.2 事例２．セルフレジ

6.2.1 事例内容

この事例で対象としたのは，アスタリスクとファーストリテイリングのセルフレジに関する特許である．この事例は，アスタリスクが新たに出願した特許（特許 6532075）に対してファーストリテイリングが無効審判を求めたものである．特許 6532075 の請求項は以下の通りである．

特許 6532075 の請求項

【請求項１】　物品に付されたＲＦタグから情報を読み取る据置式の読取装置であって、前記ＲＦタグと交信するための電波を放射するアンテナと、前記アンテナを収容し、上向きに開口したシールド部と、前記シールド部の前記開口の上側に設けられ、前記物品が載置される載置部とを備え前記物品が前記載置部に載置された状態で、前記ＲＦタグから情報を読み取ることを特徴とする、読取装置。

【請求項２】　物品に付されたＲＦタグから情報を読み取る据置式の読取装置であって、前記ＲＦタグと交信するための電波を放射するアンテナと、前記アンテナを収容し、上向きに開口したシールド部と、前記シールド部の前記開口の上側に設けられ、前記物品を収容する買物カゴが載置される載置部と、を備え、前記買物カゴが前記載置部に載置された状態で、前記ＲＦタグから情報を読み取ることを特徴とする、読取装置。

【請求項３】　前記シールド部は、前記電波を吸収する電波吸収層と前記電波を反射する電波反射層のいずれか一方または両方を含むことを特徴とする、請求項１または請求項２に記載の読取装置。

【請求項４】　請求項１から請求項３のいずれか１項に記載の読取装置と、前記情報に基づいて前記物品に関する情報を表示するタブレット端末と、を備える情報提供システム。

【請求項５】　請求項１から請求項３のいずれか１項に記載の読取装置と、前記情報に基づいて前記物品に関する情報を表示するモバイル端末と、を備える情報提供システム。

請求の主張では，無効理由として「特開 2008-99266 号公報」,「特表 2007-523563 号公報」,「特開 2007-72681 号公報」の３つを主な文献として主張している．それに付随して,「特開 2015-64673 号公報」,「特開 2010-267010 号公報」,「特開 2015-207119 号公報」,「特開 2007-34789 号公報」を含むいくつかの文献が引かれている．この件について，無効審判の結果としては特許は有効であるという結果となったが，現在は和解によって解決している．本検証では，特許 6532075 を入力した際に，引用された 7 つの特許を引用することができるのかを検証した．また，特に重要とされている,「特開 2008-99266 号公報」,「特表 2007-523563 号公報」,「特開 2007-72681 号公報」の３つについてより高い順位で検索が可能なのかを検証した．ただし，検索対象は事例１と同様に，実験で用いた被引用データセットに対象の７件の特許を加えた 13,130 件からなるデータセットとした．

6.2.2 実行結果

表 9: 事例 1 における類似度の高かった特許

順位	類似度	特許公開番号
1	0.809	**特開 2015-207119**
2	0.796	**特開 2010-267010**
3	0.752	特開 2006-350806
4	0.729	**特表 2007-523563**
5	0.719	特開 2000-149072
:	:	:
7	0.686	**特開 2008-99266**
:	:	:
29	0.587	**特開 2015-64673**
:		:
49	0.552	**特開 2007-34789**
:	:	:
69	0.518	**特開 2007-72681**
:	:	:

実験にて最も多くの評価指標で最高の値であった，クラスまでを Hard Negative として入力したモデル (3-1) を用い，特許 6532075 を入力した際の類似度を高い順に並べたものは表 9 のようになった．ただし，太字のものが実際に引用されていた特許である．これを見ると，データ数全体がおよそ 13000 件程度あることから，すべての特許が比較的上位に来ていることがわかる．

次に，引用されていないが類似度が高いと判定された,「特開 2006-350806」と「特開 2000-149072」の請求項は以下のようになっている．

特開 2006-350806 の請求項の一部

【請求項 1】アンテナの交信領域内に存在する複数の無線タグから情報を非接触通信により読取るタグ情報読取手段と、このタグ情報読取手段により情報が読取られた前記無線タグの数を計数するタグ数計数手段と、前記アンテナの交信領域内に存在する前記無線タグの数を取込む数取込み手段と、この数取込み手段により取込まれた前記無線タグの数と前記タグ数計数手段により計数された前記無線タグの数とを比較する比較手段と、この比較手段により両者の数が一致したことを条件に前記タグ情報読取手段により読取られた前記無線タグの情報を確定する情報確定手段と、を具備したことを特徴とする無線タグ認識装置。（後略）

特開 2000-149072 の請求項の一部

【請求項 1】アンテナ部にかざされた非接触式ＩＣカードとの間でデータ授受を行って電子マネー取引や自動改札処理等の所定の処理を行う非接触式ＩＣカード処理装置において、前記アンテナ部からそのアンテナ部にかざされた非接触式ＩＣカードまでの距離を検出する検出手段と、前記アンテナ部までの距離が所定以内のときに、前記データ授受を開始させる制御手段と、を有することを特徴とする非接触式ＩＣカード処理装置。

【請求項 2】前記アンテナ部までの距離が所定の距離を越えるときに、前記非接触式ＩＣカードをそのアンテナ部へ近付けるようにとの案内を行う案内手段を設けたことを特徴とする請求項１記載の非接触式ＩＣカード処理装置。

「特開 2006-350806」には,「無線タグ」や「非接触通信」,「タグ情報読取」といった表現が含まれており，類似した内容であることがわかる．また,「特開 2000-149072」についても,「RFID」という記載はないものの，RFID は電波の送受信により非接触で IC タグのデータを読み書きするものであるという意味で考えると，類似している分野であることがわかる．実際,「特開 2000-149072」の全文には,「なお、本体１と非接触式ＩＣカードＣとの間の通信方式には、上述の電波方式に限らず、光通信方式、電磁結合方式あるいは電磁誘導方式等を採用することもできる。」という記載があり，RFID における「電波」や「非接触」,「IC タグ」といった要件は満たしており，意味合いとしては類似しているといえる．

また，特に類似度が高いと推定される「特開 2008-99266 号公報」,「特表 2007-523563 号公報」,「特開 2007-72681 号公報」の３つについては，順位がそれぞれ 7，4，69 と比較的悪いことがわかる．特に，類似度が低く見積もられてしまっていた「特開 2007-72681」について，請求項の一部は以下の通りである．

特開 2007-72681 の請求項の一部

【請求項 1】ＲＦＩＤ（RadioFrequencyIdentification）タグとの交信制御にアンチコリジョン方式を用いることで複数のＲＦＩＤタグからデータを一括読取可能なＲＦＩＤタグリーダの制御装置において、一括読取りされたＲＦＩＤタグのデータを破棄するデータ破棄モードと当該データを取込むデータ取込モードとを識別するモード識別情報の記憶手段と、前記ＲＦＩＤタグリーダでＲＦＩＤタグのデータが一括読取りされる毎に前記モード識別情報が前記データ破棄モードを示すのか前記データ取込モードを示すのかを判定するモード判定手段と、前記モード識別情報が前記データ破棄モードを示すとき、前記ＲＦＩＤタグリーダで読取られたデータの中に予め設定された取込許可タグのデータが存在するか検索する破棄モード時タグ検索手段と、この破棄モード時タグ検索手段により前記取込許可タグのデータが存在しないことが確認されると、前記ＲＦＩＤタグリーダで読取られたＲＦＩＤタグのデータを破棄するデータ破棄手段と、前記破棄モード時タグ検索手段により前記取込許可タグのデータが存在することが確認されると、前記モード識別情報を前記データ取込モードの情報に切替える取込モード移行手段と、前記モード識別情報が前記データ取込モードを示すとき、前記ＲＦＩＤタグリーダで読取られたデータを取込み、コントローラへ供給するデータ取込処理手段と、を具備したことを特徴とするＲＦＩＤタグリーダ制御装置。（後略）

これを見ると,「RFID」という単語は含まれているものの,「アンテナ」という単語が出現しないことがわかる．一方で，上位に出現した特許はいずれも,「アンテナ」という単語が含まれていることが挙げられる．したがって，本事例では「アンテナ」という特定の単語の影響が大きな影響を持っており,「アンテナ」という単語が含まれていないものの類似度を下げるように学習していた可能性がある．このことから，特許の分野を認識するうえで，特定の単語に強く依存してしまっている可能性が高いと考えられる．一方で,「特開 2007-72681」の請求項には「アンテナ」という単語が出現しない一方で，明細書本文には以下のような記述がある．

> RFID タグリーダ１２には、タグ交信用アンテナ３０が接続されている。タグ交信用アンテナ３０は、図２に示すように、店の会計場所に設けられたレジカウンタ３１に埋め込まれている。これにより、レジカウンタ３１に RFID タグが近づけられると、ＲＦＩＤタグリーダ１２がその RFID タグとアンテナ３０を介して非接触で交信を行って、RFID タグが有するＩＣチップメモリ内のデータを読取るものとなっている。

ここには，アンテナという記述があり，請求項だけでなく，本文をすべて含めた入力をすることが可能であれば，現状のモデルでも「特開 2007-72681」を上位に出現できる可能性があることがわかる．

7 結論

最後に，本研究のまとめと今後の課題について述べる．

本研究では，Contrastive Learning を応用して類似特許検索を行った．その際に，大きく分けて三つの工夫を行った．まず，Unigram Language Modeling を応用したトークナイザーによって入力トークンを少なくし，請求項の全文を入力した．次に，引用情報を正例として Contrastive Learning を実行した．最後に，IPC を用いたハードネガティブについて，サンプリング方法とともに提案した．また，提案手法について実際の特許データを用いて実験を行い検証した．その結果，まず入力については請求項全文を入力することが比較的良いことが分かり，引用情報を用いた教師有 SimCSE によっても改善していることが分かった．さらに，Hard Negative の選択方法については，クラスとサブクラスを用いたものが最も精度が高いことがわかった．一方で，モデルを単体で用いるためには十分でなく，実用化するにはさらなる改善が求められることが分かった．さらに，実際の無効審判のいくつかの例について，実際に使った場合を想定して検証を行った．その結果，大まかな分野は検出できているものの，特定の単語に依存してしまうケースがあるなどの課題が存在することがわかった．

今後の課題としては，大きく三つ存在する．まず，請求項全文の入力である．本研究では，自作トークナイザーの作成によって入力トークンを減らすことを行ったが，1024 トークンまでしか入力することができておらず，15%程度の特許は全文入力することができていない．近年ではメモリ効率の良いエンコーダについても研究が進んでいるため，これらの応用によって対応できる可能性がある．次に，専門用語の入力である．実験によってうまく予測できなかった文書についていくつか確認すると，分野特有の専門用語が含まれる特許を選択できていない場合があった．したがって，専門用語をトークンに含めることでその特徴を捉えやすくなり，より精度の良い学習が行える可能性がある．最後に，実際の無効審判データを用いた評価についてである．本研究では，一部の無効審判データに対して実行した結果を目視で確認することはしているものの，多くのデータに対して統一的に評価を行ってはいない．したがって，本来の目的である無効審判データを用いた定量的な評価は今後必要である．

参考文献

[1] Abood Aaron and Feltenberger Dave. Automated patent landscaping. *Artificial Intelligence and Law*, Vol. 26, No. 2, pp. 103–125, 2018.

[2] Hidir Aras, Rima Türker, Dieter Geiss, Max Milbradt, and Harald Sack. Get your hands dirty: Evaluating word2vec models for patent data. In *Proceedings of the SEMANTiCS Posters&Demos*, 2018.

[3] Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 632–642, 2015.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the International Conference on Machine Learning*, pp. 1597–1607, 2020.

[5] Seokkyu Choi, Hyeonju Lee, Eunjeong Park, and Sungchul Choi. Deep learning for patent landscaping using transformer and graph embedding. *Technological Forecasting and Social Change*, Vol. 175, p. 121413, 2022.

[6] Kevin Clark, Minh-Thang Luong, Quoc Le, and Christopher Manning. Electra: Pre-training text encoders as discriminators rather than generators. In *Proceedings of the International Conference on Learning Representations*, 2020.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 4171–4186, 2019.

[8] C. J. Fall, A. Törcsvári, K. Benzineb, and G. Karetka. Automated categorization in the international patent classification. *SIGIR Forum*, Vol. 37, No. 1, p. 10–25, 2003.

[9] Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the Empirical Methods in Natural Language Processing*, 2021.

[10] Mattyws Grawe, Claudia Martins, and Andreia Bonfante. Automated patent classification using word embedding. In *Proceedings of the IEEE International Conference on Machine Learning and Applications*, pp. 408–411, 2017.

[11] Yuki Hoshino, Yoshimasa Utsumi, Yoshiro Matsuda, Yoshitoshi Tanaka, and Kazuhide Nakata. Ipc prediction of patent documents using neural network with attention for hierarchical structure. *Research Square preprint DOI:10.21203/rs.3.rs-1164669/v1*, 2022.

[12] Folasade Olubusola Isinkaye, Yetunde O Folajimi, and Bolande Adefowoke Ojokoh. Recommendation systems: Principles, methods and evaluation. *Egyptian informatics journal*, Vol. 16, No. 3, pp. 261–273, 2015.

[13] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Proceedings of the Advances in Neural Information Processing Systems*, 2020.

[14] Jaeyoung Kim, Janghyeok Yoon, Eunjeong Park, and Sungchul Choi. Patent document clustering with deep embeddings. *Scientometrics*, Vol. 123, No. 2, pp. 563–577, 2020.

[15] Taku Kudo. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 66–75, 2018.

[16] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations. In *Proceedings of the International Conference on Learning Representations*, 2020.

[17] Jieh-Sheng Lee and Jieh Hsiang. Patent claim generation by fine-tuning openai gpt-2. *World Patent Information*, Vol. 62, p. 101983, 2020.

[18] Jieh-Sheng Lee and Jieh Hsiang. Patent classification by fine-tuning bert language model. *World Patent Information*, Vol. 61, p. 101965, 2020.

[19] Shaobo Li, Jie Hu, Yuxin Cui, and Jianjun Hu. Deeppatent: patent classification with convolutional neural networks and word embedding. *Scientometrics*, Vol. 117, No. 2, pp. 721–744, 2018.

[20] 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.

[21] Suzuki Masatoshi. cl-tohoku/bert-japanese-v2: Bert base japanese, 2020.

[22] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proceedings of the International Conference on Learning Representations*, 2013.

[23] Leif Peterson. K-nearest neighbor. *Scholarpedia*, Vol. 4, No. 2, p. 1883, 2009.

[24] Julian Risch, Samuele Garda, and Ralf Krestel. Hierarchical document classification as a sequence generation task. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries*, 2020.

[25] Toshinori Sato, Taiichi Hashimoto, and Manabu Okumura. Implementation of a word segmentation dictionary called mecab-ipadic-neologd and study on how to use it effectively for information retrieval (in japanese). In *Proceedings of the Annual Meeting of the Association for Natural Language Processing*, 2017.

[26] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 1715–1725, 2016.

[27] Shan Suthaharan. Support vector machine. In *Proceedings of the Machine learning models and algorithms for big data classification*, pp. 207–235, 2016.

[28] Kazuma Takaoka, Sorami Hisamoto, Noriko Kawahara, Miho Sakamoto, Yoshitaka Uchida, and Yuji Matsumoto. Sudachi: a japanese tokenizer for business. In *Proceedings of the International Conference on Language Resources and Evaluation*, 2018.

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the Advances in Neural Information Processing Systems*, pp. 5998–6008, 2017.