

IPC と要約文を用いた特許 Encoder と教師なし分類手法の提案

東将己¹ 内海祥雅² 中田和秀¹

¹ 東京工業大学大学院 ² 楽天グループ株式会社

higashi.m.ac@m.titech.ac.jp

概要

特許には、その特許の技術分野を表す IPC (国際特許分類) が付与されている。しかし、この分類は粒度は荒いため、企業が経営戦略の立案に利用するには、その企業が行っているビジネス・技術分野を踏まえた特許分類の方が望ましい。そのため本研究では企業独自の詳細な特許分類を機械学習を用いて行う。1つ目として、特許の IPC と要約文を入力とした特許 Encoder からなる教師あり学習を行い、さらに IPC の表現力を高めるために、IPC と要約文の関連性を利用した事前学習法と組み合わせる手法を提案する。2つ目として、アノテーションを行わず一般的な特許文書データのみを用いた教師なし学習を行い、その結果を利用して企業独自の特許分類を行う手法を提案する。

1 はじめに

特許は発明を保護するものであり、社会において重要な役割を果たしている。そのような特許の主な分類法として、国際的に定められた IPC や地域特有の CPC, FI などがある。それらは現状、特許審査官が手作業で分類しているため、作業コストや人為的ミスが大きな課題となっている。そのため、IPC などの特許分類を機械学習によって自動化する研究が行われてきた。しかし、これらの汎用的な分類は粒度が荒く使いにくい。企業が経営戦略(知財戦略)を立案する際には、その企業が行っているビジネス・技術分野を踏まえた独自の観点での特許分類の方が有用である。そのため IPC などの分類情報を活用しつつ、企業独自の観点での粒度の細かい特許分類を自動で行う手法を開発する。

本研究では2つの問題設定を考える。1つ目は企業独自の観点で特許分類されたアノテーションデータがある状況であり、2つ目はアノテーションデータが全くない状況である。1つ目の問題設定の方が特許分類の精度は期待できるが、手作業でアノテ

ションする作業コストは高いため、それを必要としない2つ目の問題設定も考える。

1つ目の問題設定では、教師あり学習を行うことになる。そのため、特許の IPC と要約文を入力とする特許 Encoder を提案する。IPC を入力に用いる既存研究は存在するが、それらは IPC の共起性のみに着目していた。本研究では IPC の表現力向上のために IPC の共起性だけではなく、要約文にも着目したのものとなっている。また、入力に使う IPC の表現力を高めるために、3種類の事前学習手法を提案する。事前学習では、アノテーションされていない多数の特許における IPC と要約文の関連性を用いる。

2つ目は問題設定では、教師なし学習を行うことになる。まず、アノテーションされていない多数の特許を用いて対照学習を行う。そして、企業が想定する特許分類の内容を表すクラス名に LLM を適用して得られたテキストを利用して、企業独自の特許分類を行う。そのとき、1つ目の問題設定で提案した特許 Encoder も利用する。

2 関連研究

機械学習を用いた特許の自動分類として多くの研究があるが、ここではその中でも深層学習を用いた代表的な研究を紹介する。まずは IPC を分類する既存研究を3つ紹介し、その後 IPC 以外のものを分類する既存研究を2つ紹介する。

IPC 分類における先行研究の1つ目は DeepPatent[1] である。これは特許の要約文を skip-gram を用いて数値化し、CNN を用いて特許の要約文から IPC 分類を試みたモデルである。2つ目は PatentBERT[2] で、特許のタイトル、要約文、請求項を入力として、BERT[3] を用いて IPC, CPC 分類を行ったモデルである。その結果 DeepPatent と比較すると、precision@1, F1@5 において精度が向上したという結果を得ている。3つ目は Hoshino et al.[4] が提案した単語の出現頻度と IPC の階層構造を考慮しながら IPC 予測を行ったモデルである。入力には特

許文書の請求項を用いている。PatentBERT と比較して precision, recall, F1, AUC の精度が向上したという結果を得ている。

IPC 以外の分類における先行研究の 1 つ目は Choi, et al. [5] が提案した、開発した技術が特許侵害していないか調べるモデルである。IPC, CPC, USPC の特徴量を diff2vec[6] を用いて作成した上で、要約文の入力モデルとして (事前学習されていない) Transformer[7] を用いている。2 つ目は Higashi, et al. [8] が提案した IPC と要約文を用いて、企業の経営戦略に用いるために独自に作成した技術分野別のカテゴリに分類するモデルである。IPC の特徴量は共起グラフから Node2vec[9] によって作成し、要約文の入力モデルとして BERT を用いている。

3 提案手法

3.1 提案 Encoder

本研究では IPC と特許文書の (自然言語的) 関係性を捉えるために Asymmetric Attention[10] を参考に図 1 のようなモデル Asymmetric Patent Encoder を提案する。Asymmetric Patent Encoder は TransformerEncoder を用いた言語モデルと Multi-Head-Attention を組み合わせたモデルになっている。TransformerEncoder で採用されている Attention 機構により、単語間の関係の重要度を考慮することが可能となる。そして、それらは IPC 側のモデルにも採用した。なお、TransformerEncoder 部分のパラメータについては東北大が公開している大規模データで学習された BERT base モデル [11] を使用する。これにより言語の汎用的な特徴を利用することができる。ただし、学習時間の削減のために TransformerEncoder のパラメータは更新しないようにする。

そして、要約文のテキスト情報、IPC と要約文の関係性、IPC 間の共起性の 3 つを同時に考慮するために、図 2 のように BERT と Asymmetric Patent Encoder と Higashi, et al. [8] が提案した共起グラフと Node2vec を用いた Encoder の 3 つの出力を concat したものを提案 Encoder とする。

3.2 IPC の事前学習手法

3.1 節で Asymmetric Patent Encoder を提案したが、そこで使う IPC の表現力を向上させるためにアノテーションされていない多数の特許データを用いた 3 つの事前学習手法を提案する。

1 つ目は特許文書と IPC が紐づいているかどうかの二値分類タスクを学習するものである。ある特許に対して各 IPC の Asymmetric Patent Encoder の出力を 1 次元に変換する。そのあとそれらを Sigmoid 関数と Binary Cross Entropy に入力することで、その特許に付与されている IPC は 1, そうでないものは 0 に近づくように学習を行う。

2 つ目は cos 類似度を回帰タスクで学習するものである。ある特許の Asymmetric Patent Encoder の BERT 部分の出力に対して各 IPC の Asymmetric Patent Encoder の出力との cos 類似度を計算する。そしてその特許に付与されている IPC は 1, そうでないものは -1 に近づくように回帰を行う。

3 つ目は triplet loss[12] を用いたものである。ある特許文書の埋め込みとその特許に付与されている IPC の埋め込みのユークリッド距離を近くし、そうでないものは遠くなるように学習を行う。本研究では特許文書の埋め込みについては Asymmetric Patent Encoder の BERT 部分の出力、IPC の埋め込みについては各 IPC の Asymmetric Patent Encoder の出力を用いる。

3.3 教師なし分類手法

教師あり学習を行うには、企業独自の観点での特許分類データを必要とするため、事前に手作業でアノテーションを行う必要がある。この作業コストは非常に高く、実用上の制限となる。そのため本節では、アノテーションを必要としない教師なし学習を用いた特許分類法を提案する。

教師なし分類手法として自由なクラス名を入力できる対照学習に着目する。対照学習は SimCLR[13], SimCSE[14], CLIP[15] などに用いられている手法であり、ペアであるデータの分散表現の距離を近くするものである。本研究では図 3 のような学習フレームワークを提案する。3.1 節で説明した特許の要約文と IPC を入力する提案 Encoder に加え、特許の背景技術の項目の文書を入力とする BERT を用いて学習する。損失関数には InfoNCELoss を使用する。InfoNCELoss とはバッチ内の二つの出力について類似度行列を作成し、対角成分の類似度を高くしそれ以外の部分を低くするように softmax 関数を用いた学習を行うものである。具体的には、以下の式のように表される。

$$\text{InfoNCELoss} = - \sum_{i=1}^r \log \frac{e^{\text{sim}(x_1^{a_i}, x_2^{b_i})}}{\sum_{j=1}^r e^{\text{sim}(x_1^{a_i}, x_2^{b_j})}} \quad (1)$$

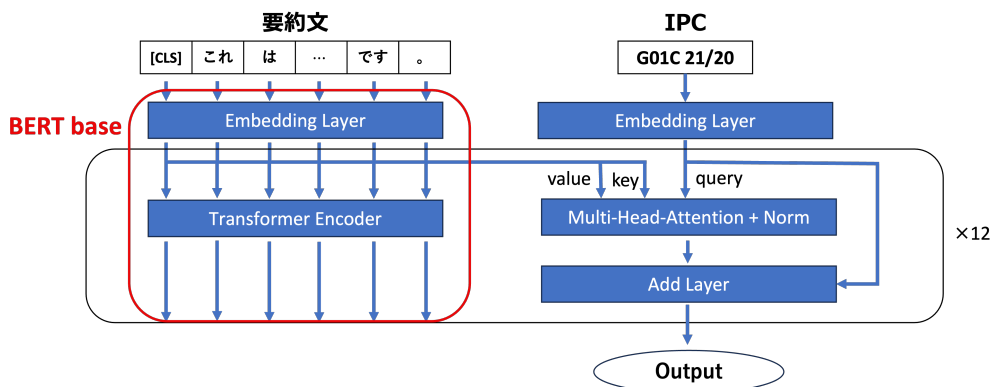


図 1 Asymmetric Patent Encoder

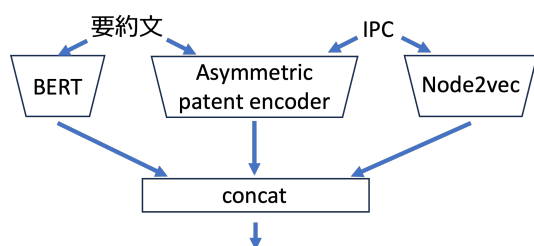


図 2 提案 Encoder

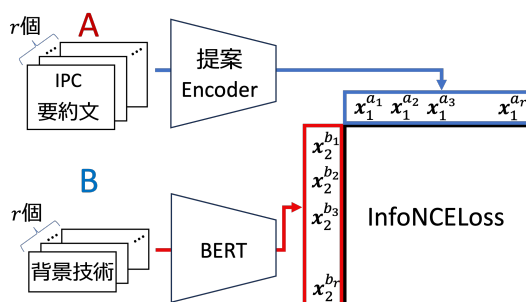


図 3 教師なし学習のフレームワーク

続いて推論では図 4 のように特許の要約文と IPC を提案 Encoder に入力し、「ペイアプリにおける利用促進」などのクラス名を BERT に入力する。ただし、クラス名をそのまま入力すると表現力が乏しいという問題が生じたため、GPT3.5[16] を用いてクラス名を説明文に変換した。変換の方法については例を提示せずに生成させる zero-shot 変換と、例を提示する few-shot 変換を用いる。

4 数値実験結果

4.1 IPC 事前学習の検証

3.2 節で説明した IPC 事前学習の有無、並びに 3 つの事前学習方法の精度検証を行う。Encoder は BERT と Asymmetric Patent Encoder を組み合わせたモデルである。事前学習は 951,340 件の特許データ、

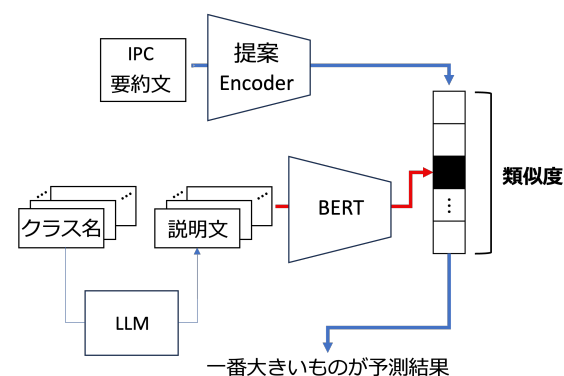


図 4 推論時のフレームワーク

教師あり学習はアノテーションされた特許データ 8,160 件で行い、訓練、検証、テストデータに 6:2:2 で分割する。

表 1 事前学習別の精度比較

| 事前学習手法 | precision | recall | F1 | PR-AUC |
|--------------|---------------|---------------|---------------|---------------|
| 事前学習なし | 0.6691 | 0.6162 | 0.6416 | 0.6682 |
| 二値分類 | 0.6972 | 0.6198 | 0.6562 | 0.6891 |
| cos 類似度 | 0.6661 | 0.6245 | 0.6446 | 0.6805 |
| triplet loss | 0.6502 | 0.6097 | 0.6293 | 0.6652 |

各事前学習手法ごとの結果が表 1 となる。事前学習なしの状態と比較して二値分類による事前学習は大きく精度向上に繋がると分かった。その一方、cos 類似度による事前学習は精度向上が小さく、triplet loss による事前学習は精度が向上していない。このように特許文書の埋め込みと IPC の埋め込みを用いた埋め込みベースの事前学習手法は精度向上に結びつきにくいことがわかった。

4.2 Encoder による精度比較

3.1 節で提案した Encoder と既存の Encoder による精度比較を 4.1 節で用いた特許データで行う。事

表2 事前学習別の精度比較

| BERT | Patent Asymmetric Attention | Node2vec | precision | recall | F1 | PR-AUC |
|------|-----------------------------|----------|---------------|---------------|---------------|---------------|
| ○ | × | × | 0.6295 | 0.6248 | 0.6271 | 0.6213 |
| ○ | ○ | × | 0.6972 | 0.6198 | 0.6562 | 0.6891 |
| ○ | × | ○ | 0.6616 | 0.6509 | 0.6562 | 0.6871 |
| ○ | ○ | ○ | 0.6762 | 0.6598 | 0.6679 | 0.7037 |

前学習方法については二値分類を用いる。結果が表2である。要約文単体を入力するBERTのみのモデルと比較して、要約文とIPCを入力に用いるBERT+Node2vecやBERT+Asymmetric Patent Encoderは精度が大きく伸びていることがわかる。このことから特許分類においてIPCの利用は有効であることといえる。また、本研究で提案したAsymmetric Patent Encoderは、[8]で提案された共起グラフを用いた手法と同等の効果があることが示唆される。そして、提案Encoder(BERT+Asymmetric Patent Encoder+Node2vec)は、全てのモデルの中で最も精度が高い。つまり共起グラフとAsymmetric Patent Encoderが捉えているIPCの特徴は別の要素であり、2つを組み合わせることでより高い精度を達成していると考えられる。

4.3 教師なし分類手法の精度検証

3.3節で提案した教師なし学習を395,470件の特許データで行う。ここでは、推論時にクラス名を変換せずにそのまま利用、zero-shot変換、few-shot変換の3つ変換について精度がどのように変化するかを検証する。推論については4.1節で用いた教師あり学習のテストデータを使用し、評価指標は類似度の高い上位K個のクラスを真と予測したときのprecision, recall, F1, accuracyとする。結果が表3である。まずzero-shot変換によって精度が大きく伸びていることからGPT3.5を用いたクラス名の変換は効果的であることがわかる。次にfew-shot変換によってzero-shot変換よりも精度をさらに向上させられることがわかる。そのためGPT3.5によるクラス名の変換は効果的であることがわかった。

4.4 実用性に対する考察

教師あり学習における提案Encoder(BERT+Asymmetric Patent Encoder+Node2vec)の性能は、表2よりprecisionが約68%、recallが約66%であることがわかる。よって、機械学習単体での運用は難しいが、

表3 教師なし分類:変換比較

| | | 変換なし | zero-shot | few-shot |
|--------|-----------|--------|-----------|---------------|
| K = 1 | precision | 0.1597 | 0.2700 | 0.3650 |
| | recall | 0.0702 | 0.1187 | 0.1604 |
| | F1 | 0.0975 | 0.1649 | 0.2228 |
| | accuracy | 0.1597 | 0.2700 | 0.3650 |
| K = 5 | precision | 0.1059 | 0.1503 | 0.1700 |
| | recall | 0.2327 | 0.3303 | 0.3736 |
| | F1 | 0.1456 | 0.2066 | 0.2337 |
| | accuracy | 0.4501 | 0.5715 | 0.6640 |
| K = 10 | precision | 0.0801 | 0.1083 | 0.1192 |
| | recall | 0.3522 | 0.4760 | 0.5237 |
| | F1 | 0.1306 | 0.1765 | 0.1942 |
| | accuracy | 0.6073 | 0.7189 | 0.8101 |

機械学習による分類結果を人間が適宜修正するといった使い方が現実的である。続いて教師なし分類手法については、表3より上位10個を取ってくることで約81%の特許について有効なクラスを提供できることがわかる。そのため分類クラスの候補を10個程度まで絞り、そこから専門家が有効なクラスに紐づけるといった利用により、専門家への負担を軽減することが可能であると考えられる。

5 おわりに

本研究では、企業独自の特許分類を自動で行うための特許Encoderの提案と、特許の大規模データを用いた教師なし分類手法の提案を行った。その結果、提案Encoderは既存研究を上回る精度となり、教師なし分類手法についてはクラスの候補を絞ることで専門家の負担を軽減させられるツールになりうることがわかった。今後の課題として、Encoderの改良による特許分類精度の向上とGPT3.5に入力するより良いプロンプトの開発が挙げられる。

参考文献

- [1] 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.
- [2] Jieh-Sheng Lee and Jieh Hsiang. Patent classification by fine-tuning bert language model. **World Patent Information**, Vol. 61, p. 101965, 2020.
- [3] Jacob Devlin, Kenton Lee Ming-Wei Chang, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, Vol. 1, pp. 4171–4186, 2019.
- [4] Yuki Hoshino, Yoshimasa Utsumi, Yoshiro Matsuda, Yoshitoshi Tanaka, and Kazuhide Nakata. Ipc prediction of patent documents using neural network with attention for hierarchical structure. **PLOS ONE**, 2023.
- [5] Seokkyu Choi, Hyeonju Lee, Eunjeong Lucy 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] Benedek Rozemberczki and Rik Sarkar. Fast sequence-based embedding with diffusion graphs. In **International Workshop on Complex Networks**, pp. 99–107, 2018.
- [7] Ashish Vaswani, Niki Parmar Noam Shazeer, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, /Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In **Advances in Neural Information Processing Systems**, Vol. 30, 2017.
- [8] Masaki Higashi, Yoshimasa Utsumi, and Kazuhide Nakata. Patent classification for business strategy with bert. In **Intelligent Computing and Optimization**, Vol. 5, pp. 84–94, 2023.
- [9] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In **Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining**, pp. 855–864, 2016.
- [10] Lisa Anne Hendricks, John Mellor, Rosalia Schneider, Jean-Baptiste Alayrac, and Aida Nematzadeh. Decoupling the role of data, attention, and losses in multimodal transformers. **Transactions of the Association for Computational Linguistics**, Vol. 9, pp. 570–585, 2021.
- [11] 東北大学 乾研究室. Pretrained Japanese BERT models. <https://github.com/cl-tohoku/bert-japanese> (参照 2024-01-12).
- [12] Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. In **Similarity-Based Pattern Recognition**, pp. 84–92, 2015.
- [13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In **Proceedings of the 37th International Conference on Machine Learning**, Vol. 119, pp. 1597–1607, 2020.
- [14] Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing**, pp. 6894–6910, 2021.
- [15] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In **Proceedings of the 38th International Conference on Machine Learning**, Vol. 139, pp. 8748–8763, 2021.
- [16] Open AI. GPT-3.5-turbo. <https://platform.openai.com/docs/models/gpt-3-5> (参照 2024-01-12).