AI 特許紹介(39)
AI 特許を学ぶ！究める！
～ELECTRA 特許～

2022 年 4 月 8 日
河野特許事務所
所長弁理士　河野英仁

「AI 特許紹介」シリーズは、注目すべき AI 特許のポイントを紹介します。熾烈な競争となっている第４次産業革命下では AI 技術がキーとなり、この AI 技術・ソリューションを特許として適切に権利化しておくことが重要であることは言うまでもありません。

AI 技術は Google, Microsoft, Amazon を始めとした IT プラットフォーマ、研究機関及び大学から毎週のように新たな手法が提案されており、また AI 技術を活用した新たなソリューションも次々とリリースされています。

本稿では米国先進 IT 企業を中心に、これらの企業から出願された AI 特許に記載された AI テクノロジー・ソリューションのポイントをわかりやすく解説致します。

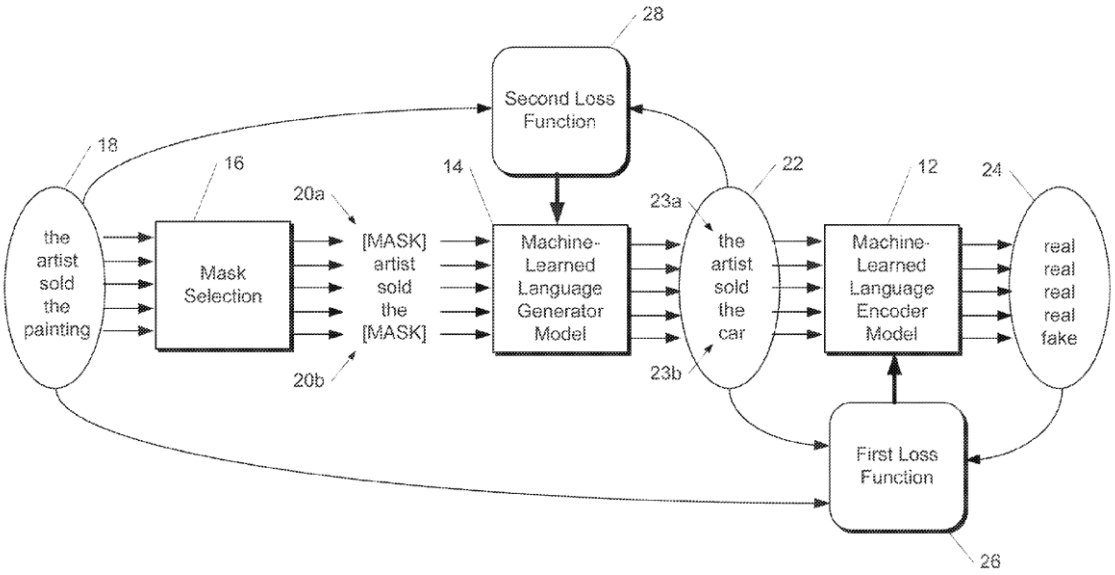
1.概要
特許出願人　Google
出願日　2020 年 9 月 21 日
公開日　2021 年 3 月 25 日
公開番号 US2021/0089724
発明の名称　言語タスクのための対照的な事前トレーニング

724 特許は、言語モデルの事前トレーニングにおけるマスク言語モデリング(MLM: Masked language modeling)に代えて、置換トークンを生成するジェネレータと、置換トークンが置き換えられたものであるか否かを識別するディスクリミネータとを用いることで、計算負荷をかけずに高速で事前トレーニングを可能とする ELECTRA(Efficiently Learning an Encoder that Classifies Token Replacements Accurately)技術に関する。

2.特許内容の説明
現在の最先端の事前トレーニング方法は、主にマスクされた言語モデリング（MLM:

masked language modeling）に依存している。MLM によるアプローチでは、入力の小さなサブセット（通常、約 15%）を選択し、トークン ID またはそれらのトークンへのアテンションをマスクしてから、元の入力を復元するようにモデルをトレーニングする。しかしながら、MLM によるトレーニングではかなりの計算コストがかかる。724 特許はこの課題を解決するものである。

図 1 は、機械学習された言語エンコーダモデル 12 の事前トレーニングプロセスにおけるデータフローを示す。



コンピューティングシステムは、複数の元の入力トークン 18 (‘the’, ‘artist’, ‘sold’, ‘the’, ‘painting’)を含む元の言語入力を取得する。

マスクされたトークンとして機能する複数の元の入力トークン 18 のうちの 1 つまたは複数が選択される。図の例では、20a と 20b に示すように、元のトークン「the」と「painting」がマスクされたトークンとして選択されている。コンピューティングシステムは、 1 つまたは複数の置換トークン 2 23a および 23b を生成する。

元の言語入力の 1 つまたは複数のマスクされたトークン 20a および 20b を、 1 つまたは複数の置換トークン 23a および 23b でそれぞれ置き換えて、複数の更新された入力トークン 22 を含むノイズのある言語入力を形成する。複数の更新された入力トークン 22 は、置換トークン 23a および 23b と、マスクされたトークンとして機能するように選択されなかった複数の元の入力トークン 18 とを含む。

機械学習された言語エンコーダモデル 12 を用いてノイズのある言語入力を処理し、更新された入力トークン 22 についてそれぞれ複数の予測 24 を生成する。更新された入力トークン 22 ごとに機械学習言語エンコーダモデルによって生成された予測 24 は、そのような更新された入力トークンが元の入力トークン 18 の 1 つであるか、または置換入力トークン 23a および 23b の 1 つであるかを予測する。

機械学習言語エンコーダモデル 12 によって生成された複数の予測 24 を評価する損失関数 26 に基づき、機械学習言語エンコーダモデル 12 を訓練する。機械学習言語ジェネレータモデル 14 は、マスクされたトークン 20a および 20b を予測するように訓練された言語モデルを含む。

置換トークン 23a および 23b と、マスクされたトークン（例えば the と painting）として機能するように選択されたトークンとの間の差を評価する第 2 の損失関数 28 に基づき、機械学習言語ジェネレータモデル 14 を訓練する。第 2 の損失関数 28 は、最尤推定関数を含む。

機械学習言語ジェネレータモデル 14 により生成された置換トークン 23a 及び 23b について、機械学習言語エンコーダモデル 12 によって生成された予測 24 を評価する第 2 の目的関数 28 に基づき、強化学習スキームで機械学習言語ジェネレータモデル 14 を訓練しても良い。例えば、ジェネレータモデル 14 は、エンコーダモデル 12 を「だます」ことに対して報酬を得ることができる。

損失関数 26 と第 2 の損失関数 28 の組み合わせを含む複合損失関数に基づいて、機械学習言語ジェネレータモデル 14 および機械学習言語エンコーダモデル 12 を共同でトレーニングする。機械学習言語ジェネレータモデル 14 と機械学習言語エンコーダモデル 12 との間では 1 つ以上の重みが共有される。機械学習言語エンコーダモデル 12 には、Vaswani et al.,2017 で説明されているトランスフォーマーネットワークテキストエンコーダが含まれる。

置換トークンの 1 つ（たとえば、23 a の'the '）が置換する元のトークン（たとえば、18 の' the '）と等しい場合、損失関数 26 は、そのような置換トークン 23a を、あたかもそれが元の入力トークン 18 に含まれているかのように評価する。たとえば、「the」23a の「real」の予測 24 は正しいと見なされる。

図 1 に示されるトレーニングプロセスに続いて、機械学習された言語エンコーダモデル 12 は、言語処理タスクを実行するようにファインチューニングすることができる。

例として、言語処理タスクは、質問応答、次の単語または文の完成または予測、翻訳、エンティティの認識、言語分類、およびその他の言語タスクを含む。

ジェネレータ G14 およびディスクリミネータ D12 に対する具体的なトレーニング方法は以下のとおりである。

入力トークン x=[x1; x2; . . . ; xn] のシーケンスをコンテキスト化されたベクトル表現のシーケンス h(x)=[h1; h2; . . . ; hn]に変換するエンコーダーを用いる。例えばエンコーダーは、トランスフォーマーネットワークまたはセルフアテンションを含むネットワークである。

所定の位置 t （例えば、x_t=[MASK]である位置）について、ジェネレータ 14 は、特定のトークン x_t を生成するための確率を（例えば、ソフトマックス層を使用して）出力する。

$$P_G(x_t \mid x) = \exp(e(x_t)^T h_G(x)_t) \Big/ \sum_{x'} \exp(e(x')^T h_G(x)_t)$$

ここで、e はトークンの埋め込みを示す。

所定の位置 t について、ディスクリミネータ 12 は、トークン x_t が”real”であるかどうか、すなわち、それが生成分布（例えば、ノイズ分布）ではなくデータ分布に由来するかどうかを予測する。ディスクリミネータの一例は以下のとおりである。

$$D(x,t)=\text{sigmoid}(w^T h_D(x)_t)$$

ここで、w はディスクリミネータの学習された重みに対応する。

ジェネレータ 14 は、マスクされた言語モデリングを実行するようにトレーニングする。入力 x=[x_1; x_2; . . . ; x_n]が与えられた場合、マスクされた言語モデリングは、最初にランダムな位置のセット（1 と n の間の整数）を選択して、m=[m_1; . . . ; m_k]をマスクする。

選択した位置のトークンは、[MASK]トークンに置き換えられる。これは、x^{masked} = REPLACE（x; m; [MASK]）として表すことができる。次に、ジェネレータ 14 は、マスクアウトされたトークンの可能性を最大化することを学習する。

ディスクリミネータ 12 は、データ内のトークンをジェネレータ 14 からサンプリングされたトークンから区別するようにトレーニングする。具体的には、「ノイズのある」例 x^{noised} 22 は、マスクされたトークン 20a および 20b をジェネレータサンプルで置き換えることによって生成する。

次に、ディスクリミネータ 12 は、x^{noised} 22 内のどのトークンが元の入力 x18 と一致しないかを予測するためにトレーニングされる。正式には、入力（最初の 3 つの方程式）とジェネレータおよびディスクリミネータの損失（最後の 2 つの方程式）式は以下の通り示すことができる。

$$m_i \sim \text{unif}\{1, n\} \text{ for } i = 1 \text{ to } k$$

$$x^{masked} = \text{REPLACE}(x, m, [\text{MASK}])$$

$$\hat{x}_i \sim p_G(x_i \mid x^{masked}) \text{ for } i \in m$$

$$x^{noised} = \text{REPLACE}(x, m, \hat{x})$$

$$\mathcal{L}_{MLM}(x, \theta_G) = \mathbb{E}\left(\sum_{i \in m} -\log p_G(x_i \mid x^{masked})\right)$$

$$\mathcal{L}_{Disc}(x, \theta_D) =$$

$$\mathbb{E}\left(\sum_{t=1}^{n} 1(x_t^{noised} = x_t)\log D(x^{noised}, t) + 1(x_t^{noised} \neq x_t)\log(1 - D(x^{noised}, t))\right)$$

GAN のトレーニング目標と類似しているが、いくつかの重要な相違点が存在する。まず、ジェネレータ 14 がたまたま正しいトークンを生成した場合、そのトークンは「fake」ではなく「real」と見なされる。この定式化は、ダウンストリームタスクの結果を改善することが知られている。

さらに重要なことに、ジェネレータ 14 は、ディスクリミネータ 12 をだますために敵対的に訓練されるのではなく、最尤法で訓練される。ジェネレータ 14 からのサンプリングを介して逆伝播することは不可能であるため、ジェネレータ 12 を敵対的に訓練することは困難である。最後に、ジェネレータ 14 はコンテキストを入力としてのみ取るが、GAN は通常、GAN ジェネレータにノイズベクトルも供給する。

したがって、学習目標の 1 つの例は、生のテキストの大きなコーパス X 上で、下記

複合損失を最小限に抑えることである。

$$\min_{\theta_C,\theta_D} \sum_{x\in X} \mathcal{L}_{MLM}(x,\theta_G) + \lambda \mathcal{L}_{Disc}(x,\theta_D)$$

損失の予想は、単一のサンプルで概算できる。

トレーニング後、本モデルはさまざまなタスクに使用できる。例として、モデルの出力の上に分類レイヤーを追加することによって、感情分析などの分類タスクを実行することができる。

別のタスク例は、モデルを含むシステムがテキストシーケンスに関する質問を受け取り、シーケンス内の回答をマークする必要がある質問応答である。一例では、Q&A モデルは、回答の開始と終了を示す 2 つの追加のベクトルを学習することによってトレーニングできる。

固有表現抽出（NER：named entity recognition）では、モデルを含むシステムがテキストシーケンスを受信し、テキストに表示されるさまざまなタイプのエンティティ（個人、組織、日付など）にマークを付ける。一例では、NER モデルは、各トークンの出力ベクトルを、NER ラベルを予測する分類層に供給することによって訓練することができる。自然言語生成は、実行できる別のタスク例である（たとえば、提案された検索クエリまたは次の単語の予測を生成するために）。

このように、トレーニングされた言語エンコーダモデルの出力を 1 つまたは複数のニューラルネットワーク層に入力して、分類、質問応答、または自然言語生成などの自然言語処理タスクを実行することができる。

次に、1 つ以上のニューラルネットワーク層が自然言語タスクの結果（分類など）を出力する。特定の自然言語タスクの自然言語モデルは、事前にトレーニングされた言語エンコーダモデルをファインチューニングすることによってトレーニングできる。

事前にトレーニングされた言語エンコーダモデルのパラメータは、初期化時にトレーニングされていない自然言語モデル（分類モデルなど）に入力できる。次に、自然言語モデルは、その特定の（下流の）自然言語処理タスクのために（たとえば、教師あり学習または教師なし学習を使用して）トレーニングできる。

したがって、事前にトレーニングされた言語エンコーダモデルを利用して、自然言語

モデルをより簡単かつ効率的にトレーニングできる（たとえば、トレーニング計算と必要なトレーニングデータの量を減らし、精度を高めることができる）。

3.クレーム

724 特許のクレーム 1 は以下の通りである。

1.機械学習言語エンコーダモデルをトレーニングするためのコンピュータ実装方法において、

1 回以上のトレーニング反復ごとに、

1 つまたは複数のコンピューティングデバイスを含むコンピューティングシステムによって、複数の元の入力トークンを含む元の言語入力を取得し、

コンピューティングシステムによって、1 つまたは複数のマスクされたトークンとして機能させるために、複数の元の入力トークンのうちの 1 つまたは複数を選択し、

コンピューティングシステムによって、1 つまたは複数の置換トークンを生成し、

複数の更新された入力トークンを含むノイズのある言語入力を形成するために、コンピューティングシステムによって、元の言語入力の 1 つまたは複数のマスクされたトークンを 1 つまたは複数の置換トークンでそれぞれ置き換え、

複数の更新された入力トークンに対してそれぞれ複数の予測を生成するために、コンピューティングシステムによって、機械学習言語エンコーダモデルを使用してノイズのある言語入力を処理し、

更新された入力トークンごとに機械学習言語エンコーダモデルによって生成された前記予測は、そのような更新された入力トークンが元の入力トークンの 1 つであるか、置換入力トークンの 1 つであるかを予測し、

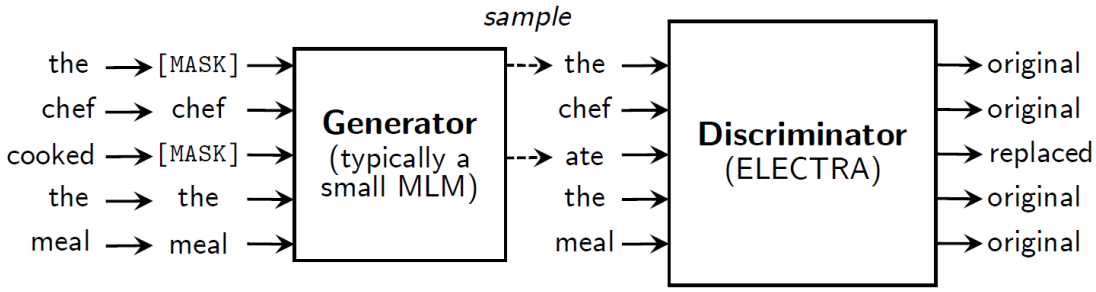
コンピュータシステムによって、機械学習言語エンコーダモデルによって生成された複数の予測を評価する損失関数に少なくとも部分的に基づいて、機械学習言語エンコーダモデルをトレーニングする。

4. 本特許に関する論文

本特許に関する論文“ELECTRA: PRE-TRAINING TEXT ENCODERS AS DISCRIMINATORS RATHER THAN GENERATORS”が、Kevin Clark 氏らにより公表されている[1]。

[1] Kevin Clark, Minh-Thang Luong, Quoc V. Le, Christopher D. Manning “ELECTRA: PRE-TRAINING TEXT ENCODERS AS DISCRIMINATORS RATHER THAN GENERATORS” arXiv:2003.10555v1 [cs.CL] 23 Mar 2020

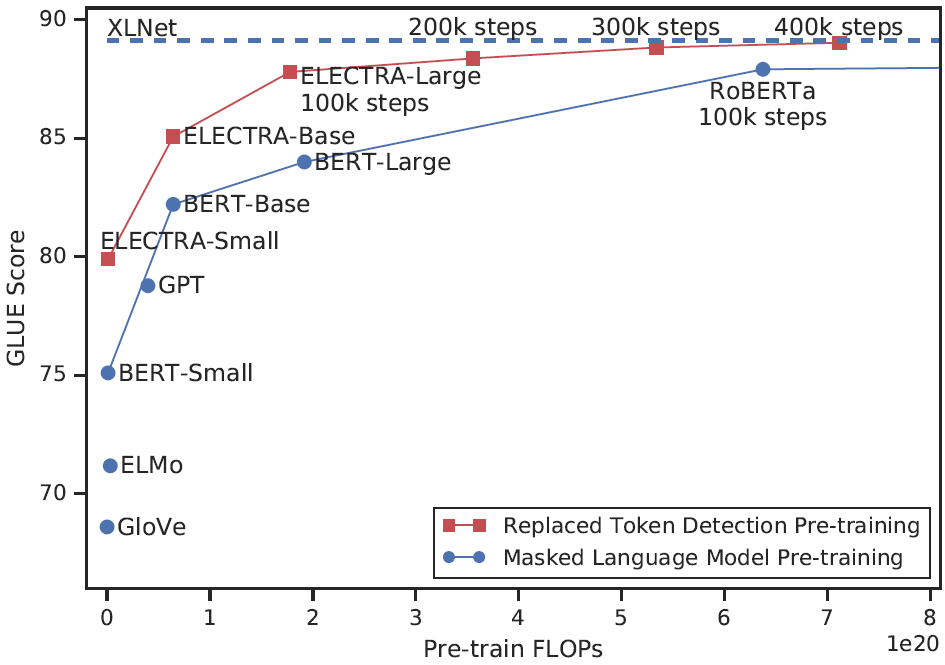
下記図は、置換トークンの生成及び検出を示す説明図である。



ジェネレータは、トークンを介して出力分布を生成する任意のモデルにすることができるが、通常、ディスクリミネータと共同でトレーニングされた小さなマスクされた言語モデルを使用する。モデルは GAN のように構造化されているが、GAN をテキストに適用するのは難しいため、敵対的ではなく最尤法でジェネレータをトレーニングする。

事前トレーニングの後、ジェネレータを破棄し、ダウンストリームタスクでディスクリミネータ（ELECTRA モデル）のみをファインチューニングする。

下記グラフは、ELECTRA 手法による性能と MLM 手法による性能とを対比して示したものである。



置換トークン検出の事前トレーニング(赤色)は、同じ計算バジェットが与えられた場合、マスクされた言語モデルの事前トレーニング（青色）よりも一貫して優れてい

ることが理解できる。

以上

著者紹介

河野英仁

河野特許事務所、所長弁理士。立命館大学情報システム学博士前期課程修了、米国フランクリンピアースローセンター知的財産権法修士修了、中国清華大学法学院知的財産夏季セミナー修了、MIT(マサチューセッツ工科大学)コンピュータ科学・AI 研究所 AI コース修了。

AI 特許コンサルティング、医療 AI 特許コンサルティングの他、米国・中国特許の権利化・侵害訴訟を専門としている。著書に「世界のソフトウエア特許(共著)」、「FinTech 特許入門」、「AI/IoT 特許入門 2.0」、「ブロックチェーン 3.0(共著)」がある。