

主要国・地域の特許関連APIの現状と その利用方法

—自作プログラムを使用した特許 API アクセスの手引き—

Status of patent-related APIs in major countries/regions and how to use them

アジア特許情報研究会

西尾 潤

現所属 株式会社ユポコーポレーション 主任研究員

2015年～アジア特許情報研究会にて活動、2018～日本知的財産協会セミナー講師

✉ info@sasiapi.org

1 はじめに

日本特許庁は2022年（令和4年）1月から特許情報取得 API の施行提供を実施しており、本年5月からは意匠・商標情報およびワン・ポータル・ドシエ（OPD）情報の施行提供に拡大した¹。加えて、主要国・地域の各知財庁においても API が提供されている²。

API（Application Programming Interface）とは、プログラム同士をつなぐ仕組みを指す。

本稿では、API の仕組み、各知財庁における API の提供状況、および API を利用して特許情報を取得する方法を紹介する。

なお、各知財庁の名称は台湾特許庁のように、正式な組織の名称ではなく、通称を用いる。

2 特許関連 API を用いてできること

国内外の特許関連 API は、以下に示す様々な活用方法が考えられる。しかし、後述のように国・地域によって API サービスの内容が異なるため、必ずしも以下に示す活用ができるとは限らない。

自社特許管理：出願番号から PCT 出願の日本国内移行後の出願番号を取得する、拒絶理由通知書を取得する、引用文献情報を取得する、年金等の経費管理、など。

他社特許監視：出願番号から審査状況を確認する、審判情報を取得する、あらかじめ設定した検索式で定期的に検索を行い、前回までの差分を出力する SDI、など。

特許情報解析：バルクデータの取得、解析用テキスト

の取得、1st OA までの期間の取得、出願から公開までの期間の取得、など。

検索サポート：IPC と CPC の変換、同義語・類義語の取得、国を跨いだ技術用語の取得（事実上の翻訳）、など。

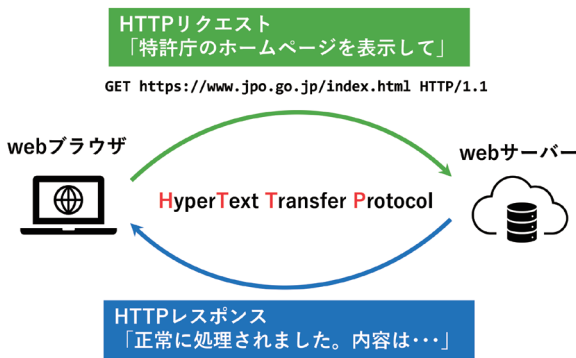
機械学習データ：技術用語の対訳コーパス、がんなどの特定分野出願データベース、など。

これらの API サービスを利用して、小規模特許管理システムを運用したり、パテントマップや特許ポートフォリオを作成したりすることができる。

3 API に関する用語について

API の通信方法は、web ブラウザで web ページを表示させる仕組みとよく似ている。web ページを表示させるとき、最初に web ブラウザはサーバーに HTTP（HyperText Transfer Protocol）リクエストを送る。リクエストを受けたサーバーは HTTP レスポンスを返す。HTTP とは通信の仕様（プロトコル）であり、実際にやり取りされるのは html（Hyper Text Markup Language）という、文字で書かれたページ記述データである。レスポンスのデータの中に html が含まれる。ブラウザは html を解析し、記述された通りにページを表示させる、というやり取りが行われている。一方、API 通信とは、web ブラウザを介することなく通信を行い、web ページを表示させるのとは異なる形式のデータを送受信する通信方法を指すことが多い。API とはデータの送受信に人間が関与することなく行われるサー

バーとの通信ということができる。



```

HTTP/1.0 200 Document follows
Content-type: text/html
.....
<!DOCTYPE HTML ..... >
<HTML>
<HEAD><TITLE>ホーム | 経済産業省 特許庁</TITLE></HEAD>
<BODY>
  ページの内容.....
</BODY>
</HTML>

```

図1 HTTP リクエストの様子

Web ブラウザを介しない API 通信としては、REST と SOAP という規格がよく用いられる。REST (REpresentational State Transfer) API は html だけでなく、さまざまな形式のデータを送信することができる。そのうち XML (Extensible Markup Language)、json (JavaScript Object Notation) は、図 2 に示すように項目と値のセットが階層構造で表現されている、文字列 (テキスト) で記述したデータである。API ではこの 2 種類の形式でデータが送られることが多い。

さらに、REST API では PDF・画像ファイル・圧縮ファイルなどのバイナリデータを送ることができる。リクエスト (送信) やレスポンス (受信) のコマンド中に、どのような種類のデータを送るか、API 仕様書に記載されているので、仕様書に基づいて API を用いて通信す

ば、自作プログラムでデータを取得できる。

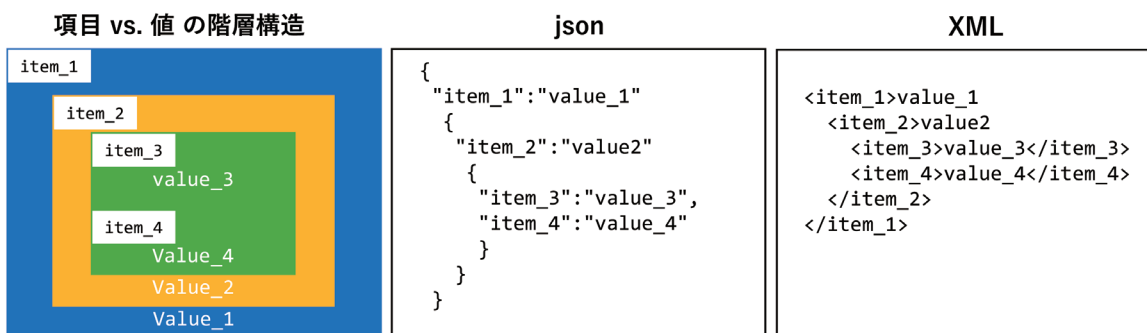


図3 日本特許庁の API 仕様書ページ³

API による通信のもうひとつの代表的なプロトコルである、SOAP (Simple Object Access Protocol) は、データの種類として XML に限定される代わりに HTTP 以外のいくつかの送受信方法が使える。SOAP の実装はやや難しいので、本稿では REST API を使った通信について紹介する。

HTTP リクエストを送る際に、問い合わせ内容 (たとえば出願番号) を URL に仕込んでデータを取得する方法を GET といい、問い合わせ内容を包含したリクエストを送りデータを登録する方法を POST という。やや乱暴な例えをすると、一般の web ページを要求するのが GET、会員登録サイトで氏名・住所・電話番号等を送信するときに用いるのが POST に相当する。

本稿では web 閲覧を前提としたサービス (たとえば Espacenet) は REST API でアクセスできるにもかかわらず取り上げない。また、一括で大量のデータをダウンロードできるバルクデータサービスも一般にブラウザからのアクセスを前提としているため取り上げない。



注：API を介した通信では改行は含まれないが、構造を見やすくするために改行とインデントを加えている。

図2 json と XML の記述方法



4 ロボットアクセス制限の仕組み

プログラムから HTTP リクエストを送ったとしてもその送信頻度があまりにも高すぎるとサーバーが処理できなくなり、最悪の場合サーバーが停止することになる。そこで web ブラウザを用いたアクセスでは、ユーザーに数字を入力させたり、自動車や信号など特定の種類の画像をクリックさせたり、さらにはジグソーパズルのピースを合わせさせたりしてロボットアクセスを制限している。

しかし、API の場合は人を介さない自動プログラムからのリクエストが前提となるため、画像を使わないいくつかのアクセス制限が設けられている。特に API で用いられるのはアクセスキーというもので、ユーザーごとに一意の文字列であり、管理者に申請すると数日後に送られてくることが多い。

アクセスキーは漏えいしないよう厳重に管理する。例えば、アクセスキーをプログラムコードに直接書かず、暗号化ファイルとして PC に保存する方法がとられる。

アクセスキーの利用方法は、日本特許庁の API サービスにおいてはまず POST でアクセスキーを送り、一時トークンを取得する。次にそのトークンをリクエスト内のヘッダーに組み込んで REST で送る。また韓国特許庁や台湾特許庁の REST API サービスでは、URL の中にアクセスキーを組み込んで送信する。このようにアクセスキーひとつとっても様々な使用方法がある。

また、サービスによっては時間内のアクセス回数制限を設けていたり、一定量を超える通信がある場合は有

料制にしたりして制限している。

5 主要国・地域の各知財庁が提供する API

以下、各知財庁の API について解説する。また、表 2 に概要を一覧にまとめる。API の数はサービスごとに集約してカウントしているため、仕様書に書かれているバリエーションの数より少なくなっている。

5.1 日本特許庁 (JPO)

2023 年 6 月現在、特許情報取得 API 14 種類、意匠情報取得 API 11 種類、商標情報取得 API 11 種類、OPD の API 6 種類が運用されている。いずれも REST API、リクエスト方式は GET で、すべてのサービスが無料である。特許情報 API の概要を表 1 に示す。詳細な情報は API 仕様書³を参照されたい。

一例として特許経過情報取得 API の場合、出願番号の入力に対して発明の名称、代理人、出願日、公開番号等のデータが返却される。

しかし、日本特許庁の API ではキーワードを用いたリクエストができない。言い換えると、検索機能を有する API は提供されていない。

API の利用申し込み手続きは特許庁の web ページ¹から利用申込書 (microsoft word 形式) をダウンロードし、必要事項を記載してメールに添付して送信する。数日後、メールで ID、パスワード、トークン取得パスが記載された添付ファイルが送信されてくる。

リクエストの手続きは、POST でアクセスキーを送

表 1 JPO 特許情報 API の概要

	リクエスト形式 (https://ip-data.jpo.go.jp/api に続けて)	名称	アクセス上限 (回/日)
GET	/patent/v1/app_progress/{出願番号}	特許経過情報取得 API	400
GET	/patent/v1/app_progress_simple/{出願番号}	シンプル版特許経過情報取得 API	400
GET	/patent/v1/divisional_app_info/{出願番号}	特許分割出願情報取得 API	30
GET	/patent/v1/priority_right_app_info/{出願番号}	特許優先基礎出願情報取得 API	30
GET	/patent/v1/applicant_attorney_cd/{申請人コード}	特許申請人氏名・名称取得 API	200
GET	/patent/v1/applicant_attorney/{申請人氏名・名称}	特許申請人コード取得 API	200
GET	/patent/v1/case_number_reference/{種別}/{案件番号}	特許番号参照 API	50
GET	/patent/v1/app_doc_cont_opinion_amendment/{出願番号}	特許申請書類取得 API	100
GET	/patent/v1/app_doc_cont_refusal_reason_decision/{出願番号}	特許発送書類取得 API	100
GET	/patent/v1/app_doc_cont_refusal_reason/{出願番号}	特許拒絶理由通知書取得 API	100
GET	/patent/v1/cite_doc_info/{出願番号}	特許引用文献情報取得 API	50
GET	/patent/v1/registration_info/{出願番号}	特許登録情報取得 API	200
GET	/patent/v1/jpp_fixed_address/{出願番号}	特許 J-PlatPat 固定アドレス取得 API	200
GET	/patent/v1/pct_national_phase_application_number/{種別}/{案件番号}	特許 PCT 出願の日本国内移行後の出願番号取得 API	200

り、アクセストークンを取得する。次にアクセストークンをリクエスト内のヘッダーに組み込んでGETで送る。また、アクセストークンと同時にリフレッシュトークンが提供されるので、アクセストークンが失効する（例えば3600秒）以内にリフレッシュトークンでPOSTすると、新たなアクセストークンが返却される。また、レスポンスにアクセス可能な残数（回/日）が返却される。

特許・意匠・商標情報 API ではテキストが json 形式で返却され、OPD の API ではテキストが XML 形式で返却される。

5.2 米国特許商標庁 (USPTO)

2023年6月現在、特許・意匠情報取得 API 18種類、商標情報取得 API 2種、Cancer Moonshot Patent Data Set API（がんの治療に関する特許データ）を運用している。

バルクデータに対する大量検索・大量ダウンロードができる BDSS (Bulk Data Storage System) や、1週間分のオフィスアクションをダウンロードできる Office Action Weekly Zips API などユニークなサービスを持つ。

特許の発明者、企業、地理的位置等の情報を提供する PatentsView と商標 TSDR は API キーを要する。しかし、それ以外のサービスは API キーが不要である。また、すべてのサービスが無料である。

それではどのようにアクセス制限するかというと、過剰なアクセスがあったときに IP ブロックされるが、どの程度のリクエスト頻度で制限がかかるかは示されていない。

アクセスキーが不要である代わりに、API 仕様書はサインインが必要な開発者のページで公開されている。

5.3 欧州特許庁 (EPO)

EPO では 2023年6月現在、Linked open EP data 6種類、Open Patent Services (OPS) 20種類、European Publication Server の API サービスを提供している。このうち OPS を除く 2 サービスは html をサポートしており、ブラウザを通じてのリクエストが可能である。OPS は公報検索、ファミリーデータ、番号変換（日本公報番号→DOCDB形式など）、登録特許検索、法律状態検索、CPC 検索、特許分類変換の合計

20種類のサービスがあり、いずれも REST API である。リクエスト方式は POST および GET であり、4GB/week まで無料である。

OPS を利用するには、ユーザー登録が必要で、その時アプリケーション（名前は任意）を登録する。OPS の web ページで My Apps > アプリケーション名と辿っていくと、Consumer Key と Consumer Secret Key が得られる。これを用いて POST するとトークンが返却される。次にトークンをヘッダーに組み込んでリクエストすると、検索条件に対応するレスポンスが返却される。なお、EPO OPS v3.2 API Documentation のページ⁴では、web ブラウザ上で各 API のテストを実行することができる。

5.4 韓国特許庁 (KIPO)

韓国特許庁は web ページベースの検索サイト KIPRIS に加え、API を提供する KIPRIS-plus を運営している。

KIPRIS-plus では 2023年6月現在、特許 10種類、意匠 4種類、商標 5種類を含む 120種類の API を運用している⁵。特許情報以外でもチャットボット用の機械学習データセットや、商標画像の機械学習データセット、日韓特許技術用語の対訳データセットなど多彩なデータを提供している。

API は REST、SOAP として提供している。API は KIPRIS-plus にユーザー登録して、ログイン後に API サービスごとに購入する。1か月 1000 回のリクエスト以内であれば無料で利用できる。

API キー (access key) はマイページで SOAP と REST 別々に発行される。REST API では URL に API キーを付加して送信する。1秒当たり 50 回のリクエストを超えると制限がかかる。

バルクデータはログイン後に Data Service 以下の各ページで web ページとして提供されている。たとえば、特許・実案の検索ページは商用データベースが提供する高度な検索並みの検索フィールドを有している。

詳細は KIPRISPlus Open API 利用ガイドのページ⁶に記載されている。また、このページから Java と Python のサンプルプログラムをダウンロードできる。

5.5 世界知的所有権機関 (WIPO)

WIPOは特許・意匠・商標に関するAPIサービスを提供していないが、APIについての議論がなされている⁷ので、近い将来APIサービスが提供されると思われる。

2023年6月現在、WIPOではWIPO PEARL APIサービスの1種類を提供している。WIPO PEARLは特許文献から抽出された科学技術用語を提供するサービスで、webページでも提供されており、WIPOのアカウントにログインせずに利用することができる。

そのうち用語検索は、入力言語と検索対象言語、オプションで技術分野を指定し、技術用語をキーワード検索すると、入力言語を機械翻訳して、対象の言語で技術用語を検索する仕組みである。この機能はREST APIでも2種類提供されており、APIキーを申請して取得することで、無料で利用できる。

WIPO PEARLのもう1つの機能はコンセプトマップ検索で、起点と終点にそれぞれ技術用語を入力すると、技術用語のネットワーク図を出力するものであるが、APIサービスとしては提供されていない。

5.6 台湾特許庁 (TIPO)

台湾特許庁は、台湾地域内の専利商標と、グローバル専利検索システム (GPSS) の2つのデータベースを有しており、それぞれ個別にAPIサービスを提供している。

台湾地域内のデータベースでは専利 (特許・実用新案・意匠) 8種類、商標5種類のAPIサービスを提供している。API利用を申請するには、専利商標開放資料ダウンロードのページ⁸の最下部からAPI申請書をダウンロードし、必要事項を記載してメール添付して、台湾特許庁に送付する。

APIサービスはREST APIであり、取得したAPIキー

をURLに組み込んで送信する。リクエストURLの中で出力形式 (json/XML) を指定する。

GPSSでは専利 (特許・実用新案・意匠) 1種類のAPIサービスを提供している。APIの説明ページ⁹内に「API 検証コード線上申請」のボタンがあり、ここからAPIキーをオンライン申請することができる。

6 API へのリクエスト方法

6.1 自作プログラムによる方法

各種プログラム言語にはHTTP通信を行う仕組みがオブジェクトとして搭載されているため、自作プログラムでREST API実装することは困難ではない¹⁰。

図4にExcel VBA (Visual Basic for Applications、一般にはマクロともいう) の例を示す。①XMLHTTPというオブジェクトを呼び出し、②GET,POSTなどのメソッドとURLを指定して通信ポートを開き、③ヘッダーを設定し、④送信 (POSTのときはパラメーターを付ける) する、という手順で完了する。

```
Sub POST_access_token()
    Dim Id As String
    Dim Pass As String

    Id = "*****"
    Pass = "*****"

    Set xml = CreateObject("MSXML2.XMLHTTP") '①

    Url = "https://*****" 'トークン取得パス
    param = "grant_type=password&username=" + Id + "&password=" + Pass

    'API接続の実行
    xml.Open "POST", Url, False '②
    xml.setRequestHeader "Content-Type", "application/x-www-form-urlencoded" '③
    xml.send (param) '④

    '通信完了まで待機
    Do While xml.readyState < 4
        DoEvents
    Loop

    'レスポンス文字列の取り出し
    Debug.Print xml.Status
    Debug.Print xml.responseText

    'GET リクエストのとき
    Url = "https://ip-data.jpo.go.jp/api/trademark/v1/case_number_reference/application/2018009480"
    xml.Open "GET", Url, False
    xml.setRequestHeader "Authorization", "Bearer " + access_token
    xml.send

    Set xml = Nothing
End Sub

200
{"access_token":"アクセストークン文字列","expires_in":3600,
"refresh_expires_in":28800,"refresh_token":"リフレッシュトークン文字列",
"token_type":"Bearer","id_token":"IDトークン文字列","not-before-policy":0,"session_state":"セッションステート文字列","scope":"openid profile"}
```

図4 Excel VBA を使用した日本特許庁 API のアクセス例

表2 各知財庁 API の概要

API サービス	JPO	USPTO	EPO	KIPO	WIPO	TIPO
特許 (実案)	14 種類	18 種類	20 種類 (OPS)	10 種類	なし	8 種類 (TIPO) 1 種類 (GPSS)
意匠	11 種類			4 種類	なし	
商標	11 種類	2 種類	なし	5 種類	なし	5 種類 (TIPO)
ドシエ	6 種類	なし	なし	なし	なし	なし
その他	なし	バルクデータなど	CPC 検索など	対訳データなど	技術用語検索	なし
アクセスキー	トークン取得用	不使用 (一部除く)	OAuth 2.0 認証用	REST : url に付加	—	REST : url に付加
アクセス制限	30 ~ 400 回/日	あり (数量非明示)	4GB/week まで無料	1000 回 / 月まで無料	—	あり (数量非明示)
キーワード検索機能	なし	あり	あり	あり	—	なし

図5にpythonの例を示す。requestsというモジュールをインポートすれば、上記①～④の動作を requests.post () または requests.get () の1回だけで行うことができる。

```
import requests
import urllib.parse

Id = r'*****'
Pass = r'*****'
Tokenpass = r'https://*****' # トークン取得パス

# APIに送信する情報
headers = {'Content-Type': 'application/x-www-form-urlencoded'}
payload = 'grant_type=password&username=' + urllib.parse.quote(Id) + '&password=' + urllib.parse.quote(Pass)

# API接続の実行
res = requests.post(Tokenpass, data=payload, headers=headers)

# レスポンス文字列
data = res.text
print(data)

# GETの送信例
Authorization = 'Bearer ' + access_token
headers = {'Authorization': Authorization}
url = r'https://ip-data.jpo.go.jp/api/trademark/v1/case_number_reference/application/2018009480'
response = requests.get(url, headers=headers)

{"access_token": "アクセストークン文字列", "expires_in": 3600,
 "refresh_expires_in": 28800, "refresh_token": "リフレッシュトークン文字列",
 "token_type": "Bearer", "id_token": "IDトークン文字列", "not-before-policy": 0, "session_state": "セッションステート文字列", "scope": "openid profile"}
```

図5 Pythonを使用した日本特許庁APIのアクセス例

6.2 公開されているプログラムを利用する方法

日本特許庁APIサービスを利用するプログラムは、特許情報取得API+（プログラム言語名）でネット検索すると得ることができる。代表的なものは次の2つである。

JpoApiClass¹¹ (VBA)

easy_patents¹² (Python)

外国のAPIサービスを利用するプログラムとしては以下のものがあげられる。

uspto-opendata-python¹³ (USPTO, Python)

patent_client¹⁴ (USPTO/EPO, Python)

python-epo-ops-client¹⁵ (EPO, Python)

patent-api¹⁶ (EPO, Python)

Rops¹⁷ (EPO, R)

JpoApiClassはマクロ付きExcelファイルで提供されており、ダウンロードすればすぐに使用できる。各プログラムの使い方は参考文献に記載のページ（特にREADME）を参照されたい。

6.3 APIのテストができるGUIツールを利用する方法

PC画面上でAPIのテストができるツールを用いれば、プログラムを書くことなくリクエストを送信することができるため、導入が簡単である。

プログラムとして提供されているものとしては、Postman¹⁸がある。また、ブラウザ（Google Chrome）の拡張機能としては、Talend API Tester¹⁹がある。これらは基本的に画面上でメソッド（GET/POST）、ホスト（URL）、ヘッダー（図5におけるheaders）、POSTの場合はBodyにパラメーター（図5におけるpayload）を設定して送信するだけで、レスポンスを受け取ることができる。また、各知財庁のAPIページでテストできる場合がある。

7 レスポンスの解析方法

サーバー側でリクエストが処理されると、レスポンスが返却される。正常であれば[200]のHTTPレスポンスコードが付加されて返ってくるので、レスポンスコードを判定して解析するプログラムを組むと、エラーを回避しやすい。

レスポンスに含まれるテキストは図2に示すように、jsonかXMLであることが多い。いずれも特許APIの場合、出願日、発明の名称等の各項目がその値と対になって返却される（図5～7）。

ユーザー側のプログラムはその文字列を解析して項目と値を取り出す。プログラム言語の種類によっては解析器（パーサーと称する）が準備されている。Pythonではjson、beautifulsoup（XML解析）というパッケージが使いやすい。パーサーがない場合は正規表現を用いて特定文字列（jsonの場合はキー、XMLの場合はタグ）を含むテキストを抜き出すとよい。

```
# data
{"access_token": "アクセストークン文字列", "expires_in": 3600,
 "refresh_expires_in": 28800, "refresh_token": "リフレッシュトークン文字列",
 "token_type": "Bearer", "id_token": "IDトークン文字列", "not-before-policy": 0, "session_state": "セッションステート文字列", "scope": "openid profile"}
```

```
import json
access_token = json.loads(data)['access_token']
refresh_token = json.loads(data)['refresh_token']
```

```
# data
{'result': {'statusCode': '100', 'errorMessage': '', 'remainAccessCount': '48', 'data': {'applicationNumber': '2018009480', 'registrationNumber': '6036291'}}}
```

```
import json
result = json.loads(response.text)['result']
print(result)
print(result['statusCode'])
print(result['data']['registrationNumber'])
```

```
{'statusCode': '100', 'errorMessage': '', 'remainAccessCount': '44', 'data': {'applicationNumber': '2018009480', 'registrationNumber': '6036291'}}
100
6036291
```

図6 Pythonを使用した日本特許庁APIのjson解析例



```
import requests
from bs4 import BeautifulSoup

Authorization = 'Bearer ' + access_token
headers = {'Authorization': Authorization}
url = r'https://ip-data.jpo.go.jp/opdapi/patent/v1/global_doc_list/JP.2007550210.A'

response = requests.get(url, headers=headers)

soup = BeautifulSoup(response.text, "html.parser")
documents = soup.find_all('document-list')
[[document.find('legal-date').text,
 document.find('document-description').text] for document in documents]

[['2007-09-12', '国際調査報告 (日本語)'],
 ['2007-09-12', '添付書類'],
 ['2007-09-12', '優先書類'],
 ['2007-09-12', '優先権書類'],
 ['2007-09-12', '日本語国際公開要約 (省略)'],
 ['2012-03-13', '拒絶理由通知書'],
 ['2012-05-09', '対応記録'],
 ['2012-05-11', '意見書'],
 ['2012-05-11', '手続補正書'],
 ['2012-07-30', '意見書'],
 ['2012-07-30', '手続補正書'],
 ['2012-08-28', '特許査定'],
 ['2012-11-27', '職権訂正通知書 (書類修正)']]

# soup
<?xml version="1.0" encoding="utf-8"(省略)>/bibliographic>
<document-lists><document-list
group="4,101,102"><legal-date>2007-09-12</legal-date><original
id="International_Search_Report%2528in_Japanese%2529_60712710794_JP"><document-description>国際調査報告 (日本語)</document-description></translated> (省略)</document-list></api-data>
```

図7 Pythonを使用した日本特許庁 OPD API の XML 解析例

バイナリデータを返すレスポンスの場合、図8に示すようにそのままファイルに書き出すことで、ファイルのダウンロードができる。

```
import io
import zipfile

Authorization = 'Bearer ' + access_token
headers = {'Authorization': Authorization}
url = r'https://ip-data.jpo.go.jp/api/patent/v1/app_doc_cont_refusal_reason/2007035937'
response = requests.get(url, headers=headers)

# レスポンスの種類がzipファイルだったとき
if response.headers['content-type'] == 'application/zip':
    contentDisposition = response.headers['Content-Disposition']
    ATTRIBUTE = 'filename='
    fileName = contentDisposition[contentDisposition.find(ATTRIBUTE) + len(ATTRIBUTE):]

# ファイルのダウンロード
with open(fileName, 'wb') as f:
    f.write(response.content)
```

図8 Pythonを使用した日本特許庁 API のファイルダウンロード例

8 おわりに

これまで五大庁のうち中国を除く4庁と、WIPO、TIPOが提供するAPIサービスについてきわめて簡単ながら紹介した。各庁が提供するAPIのサービス一覧とpythonコード例をアジア特許情報研究会のnote²⁰に投稿しているので参考になれば幸いである。

今回紹介したAPIサービス以外にも各国知財庁がAPIを提供している。その状況は参考文献2に詳しい。

また、オープンソースの特許データベースである、Lens.orgは検索可能なAPIサービスを提供している。試用したところ、webブラウザで提供している検索機能より、APIを使った検索の方が高度な検索ができた²¹。

巷で話題のChat GPTは、webブラウザ版とAPI版のサービスを提供している。webブラウザ版ではオプトアウトしない限り質問文（プロンプト）を新たな大規模言語モデルの学習に用いるが、API版では質問文を学習に利用しないとされている²²。

これらのようにAPIのサービスを上手に使うことによって、web版にない機能を引き出すことができる。

本報告は2023年の「アジア特許情報研究会」におけるワーキングの一環として報告するものである。

研究会のメンバーの皆様には様々な協力をしていただきました。ここに感謝申し上げます。

参考文献

1. 特許庁、APIを利用した特許情報の試行提供
<https://www.jpo.go.jp/system/laws/sesaku/data/api-provision.html>
2. 特許庁、令和元年度 特許情報提供サービスの現状と今後に関する調査報告書
https://www.jpo.go.jp/resources/report/sonota/service/document/index/2022hokokusho_2.pdf
3. 特許庁、特許情報取得 API
https://ip-data.jpo.go.jp/api_guide/api_reference.html#/
4. 欧州特許庁、EPO OPS v3.2 API Documentation (アクセスにはOPSのログインが必要)
<https://developers.epo.org/ops-v3-2/apis>
5. 韓国特許庁、API Data List
<https://plus.kipris.or.kr/eng/data/service/List.do?subTab=SC001&menuNo=300100>
6. 韓国特許庁、KIPRISPlus Open API 利用ガイド
<https://plus.kipris.or.kr/portal/bbs/view.do?nttId=638&bbsId=B0000001&searchCnd=&searchWrd=§ion=&sdate=&edate=&useAt=&replyAt=&menuNo=200015&viewType=&delCode=0&pageIndex=1>
7. 世界知的所有権機関、WIPO API Day 2023
https://www.wipo.int/meetings/en/details.jsp?meeting_id=76392
8. 台湾經濟部智慧財産局、専利商標開放資料下載
<https://opdata.tipo.gov.tw/>
9. 台湾經濟部智慧財産局、API 使用説明

<https://gpss1.tipo.gov.tw/gpsskmc/gpssapi>

10. 綾木健一郎、特許情報取得 API のトークン取得
https://qiita.com/kenichiro_ayaki/items/dde737d0d5561c5e4b4c
11. 綾木健一郎、JpoApiClient : 特許情報取得 API 用 VBA クラス
https://qiita.com/kenichiro_ayaki/items/12e6ef9ab1db0e513588
12. 特許情報の簡易取得パッケージ easy_patents
<https://qiita.com/easypatents39/items/d069d003b52e8c87873f>
13. Andreas Motl, uspto-opendata-python
<https://github.com/ip-tools/uspto-opendata-python>
14. Parker Hancock, patent_client
https://github.com/parkerhancock/patent_client
15. George Song, python-epo-ops-client
<https://github.com/gsong/python-epo-ops-client>
16. Martin Galese, patent-api
<https://github.com/frosly/patent-api>
17. Alessio Nardin, Rops
<https://github.com/AlessioNar/Rops>
18. Postman, Inc., Postman
<https://www.postman.com/>
19. Talend, Inc., Talend API Tester
<https://chrome.google.com/webstore/detail/talend-api-tester-free-ed/aejoelaoggembcahagimdiliamlcdmfm?hl=ja>
20. アジア特許情報研究会 note
https://note.com/sasiapi_/
21. Lens.org
<https://www.lens.org/>
たとえば、web 版では優先日と公開日の AND 検索ができないが、API ではこれが可能なため、出願日から公開日までの経過期間を指定した絞り込みができる。
22. New ways to manage your data in ChatGPT (April 25, 2023)

<https://openai.com/blog/new-ways-to-manage-your-data-in-chatgpt>

