

オープンソースソフトウェアを活用した 特許情報の検索・分析システムの構築

—特許情報の普及活用へオープンソースソフトウェアが果たす役割—

会員 小川 延浩

国立情報学研究所 教授 博士（理学） 宇野 毅明

要 約

近年のオープンソースソフトウェア（以下 OSS とする）の発展はめまぐるしく、最新の研究成果や先端技術でさえも、短期間で OSS として公開され、身近に利用可能となる。一例を挙げると、現在の人工知能ブームの火付け役とも言える Deep Learning は、2012 年に学術的なコンテストで圧倒的な性能を示して話題となった⁽¹⁾が、2014 年には Berkeley Vision And Learning Center が OSS のライブラリである CAFFE を公開し、現在では一定条件下⁽²⁾で誰でも利用可能となっている。そこで、特許情報の検索・分析システムについてみると、この情報検索という技術分野においても、OSS によってシステム構築に必要な機能がほぼ網羅されており、無償で利用できる状況となっている。特にエンタープライズ向けの検索プラットフォームである Apache ソフトウェア財団の Solr や、Elastic 社の Elasticsearch では、商用レベルの特許情報検索システムでしか利用できない機能も備えており、概念検索、あいまい検索、SDI⁽³⁾等もそのまま利用可能である。OSS を利用することで、多少のプログラミング知識があれば一人でも高機能な特許情報の検索・分析システムを手軽に作れる時代となったのである。そこで、本稿では、NASA でも利用され⁽⁴⁾、最近話題となっている OSS の検索システム Elasticsearch を取り上げ、具体的にどのような機能を実現できるかについて紹介する。特許情報が年々増加していく中⁽⁵⁾、特許調査業務の負担は増加する一方であり、最新の OSS 利用による知財関係者の業務効率向上の道筋を見つきたい。

目次

1. はじめに
2. 容易になった検索・分析システムの構築
3. オープンソースソフトウェアで実現できる検索・分析機能
 - 1) オープンソース検索エンジン Elasticsearch とは
 - 2) Elasticsearch で実現できる検索・分析機能
 - (1) ブーリアン検索
 - (2) 類似文献検索・概念検索
 - (3) あいまい検索・キーワードサジェスト
 - (4) キーワード抽出
 - (5) SDI (Selective Dissemination Information) 機能
4. 検索・分析システムの構築手順
 - 1) 特許情報の入手
 - 2) Elasticsearch の実行とデータ登録
5. むすび

1. はじめに

特許情報の検索・分析を行うためには独立行政法人工業所有権情報・研修館が提供する J-PlatPat や欧州特許庁が提供する Espacenet のように無償のデータベースも利用できるが、より高度な検索や分析を行うためには有償の商用データベースを利用することになる。商用データベースの利用にかかる費用は各社まちまちであるが、年間ライセンスで数百万円かかるケースもあり、決して安価なものではない。これは、開発費やサーバのメンテナンス費用が積み重なったものであるが、冒頭で述べたようにオープンソースソフトウェア (OSS) の活用が進むと、開発期間の削減やメンテナンス効率の向上によって、今よりも安価で質の高いサービスが提供される可能性⁽⁶⁾がある。そうなればユーザー層も増加し、中小企業等にも特許情報の活用が進むことが考えられる。そして様々なユーザー層が利用することで、また新たなユーザーニーズが発見

され、機能改善が進むというポジティブなサイクルが回る。

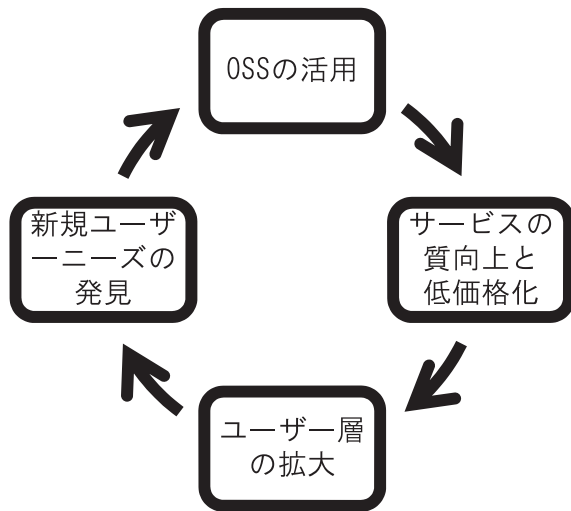


図1 OSS活用を起点とする特許情報検索・分析システムの改善サイクル

このような OSS 活用を起点とするポジティブなサイクルを回すことで、特許情報の普及活用へとつながるのではないかというのが本稿の主張である。それでは、なぜこれまでこのサイクルが十分に回ってこなかったのかというと、従来の OSS では技術的課題等によって簡単には特許情報・分析システムに適用できなかったという点が挙げられる。しかし、近年に起こったいくつかの技術革新によって、そのような障壁は次々と無くなり、現在ではこのサイクルが十分回る環境が整ったと考えられる。そこで、本稿では、まず次章で、これまでどのような点で特許情報検索・分析システムの構築が困難であったか、また、それが現在どのように解消されつつあるかについて説明する。そしてその次の 3 章では、実際に最新の OSS を活用して約 1 週間で構築したシステムの事例を紹介し、どの程度の機能を実現できているのかについて説明する。続く 4 章では、3 章で説明したシステムを構築するステップについて簡単な説明を行い、最後の章では本稿のまとめを述べる。

2. 容易になった検索・分析システムの構築

従来、特許情報の検索・分析システムの構築は、容易ではなかった。特許情報の検索は、キーワード検索が中心のタスクであるが、請求項部分や抄録部分といったように検索対象となる範囲を指定する必要があり、さらに出願番号や出願日といった数値データに関する条件も場合によっては追加される。そのため、通

常の全文検索システムをそのまま利用することができず、リレーショナルデータベースを利用して適切なデータベース設計を行う必要があった。リレーショナルデータベースは、行と列による 2 次元テーブル構造のモデルであるため、ネスト構造（1 件の特許文献に対して発明者や出願人が複数存在するというように 1 件 1 行で表せない構造のこと）を持つ特許情報を格納する場合には、正規化と呼ばれるデータ構造の再設計を行う必要がある。このプロセスに専門的な知識やノウハウが要求された。このような中、2000 年代後半から NoSQL と呼ばれる技術トレンドが起こり、リレーショナルデータベース以外の多様なデータベース⁷⁾の利活用が本格化した。そして、データベース内のデータの扱い方に柔軟性が生まれ、特許情報のようにネスト構造を持つデータでも、そのまま格納できるようなデータベースも出現した。このデータベースそのものの技術進化に加え、データベース周辺の技術進化も追い風となった。従来の OSS は、全文検索エンジンの Namazu にしても、リレーショナルデータベースの MySQL にしても、機能部分は提供されているものの、その機能にアクセスするインターフェース部分は別途開発する必要があった。しかし、サーバのログ解析などのいくつかの分野でデータ活用が活発化し、インターフェース部をカバーする周辺ツールが普及した。それら周辺ツールをあわせて活用することで、特許データを流し込むだけで、すぐに検索環境・分析システムが仕上がるようになり、構築にかかる労力は格段に小さくなった。本稿では、ドキュメント志向型のデータベースを内部に持つ OSS の検索エンジン Elasticsearch と、そのユーザーインターフェースを提供する OSS の Kibana を活用した事例を紹介し、いかに容易にシステム構築ができるか説明する。

3. オープンソースソフトウェアで実現できる検索・分析機能

1) オープンソース検索エンジン Elasticsearch とは

Elasticsearch は、Elastic 社の創設者で現 CTO である Shay Banon によりオープンソースの検索サーバプロジェクトとして開始され、2010 年 2 月に公開された。公開されてからまだ日が浅いにも関わらず、多くの企業での採用実績があり、Facebook, NETFLIX, NewYorkTimes, GoldmanSachs など、名だたる企業

が現在活用を行っている。この採用実績の背景には、基本的な検索機能を押しつつもスケーラビリティのある設計がなされていること（分散処理を前提としており、大規模データにも対応できる設計がなされている）が大きいと考えられるが、本稿では前者の基本的な検索機能としてどのような機能が提供されているかという点についてフォーカスする。

2) Elasticsearch で実現できる検索・分析機能

本節では、Elasticsearch とそのユーザーインターフェースを提供する Kibana で実現できる検索・分析機能の典型例について説明する。

(1) ブーリアン検索

ここで言うブーリアン検索とは、AND や OR といった論理演算で接続した検索条件によって検索式を作成し、その検索式に基づく検索を行う手法のことを指す。情報検索システムの最も基本的な機能であり、この機能自体はリレーショナルデータベースを用いても容易に実現可能である。ただし、リレーショナルデータベースを用いる場合は、先に述べたように正規化された複数のテーブルに対して問い合わせを行うため、Elasticsearch と比較して検索結果の統合に複雑な処理が必要となる。

次に Kibana を使った実際の検索事例を紹介する。Kibana では、Elasticsearch の内部で利用している OSS の全文検索エンジンである Apache Lucene の検索式構文が利用できる。AND, OR, ワイルドカードといった基本的な論理式が使えるだけでなく、Proximity Search（サーチエンジンによっては NEAR 検索と呼ばれることもある）と呼ぶ2つのキーワードの出現間隔を指定した検索も可能である。以下の例は、出願人に toyota と前方一致し、かつ特許分類記号の CPC が b62d6 で前方一致する条件の検索結果を示している。



図2 ブーリアン検索結果の例

また、次のように、検索結果の1レコードをより詳細に表示することもでき、Kibana だけでも最低限の特許検索機能は利用可能である。



図3 検索結果の詳細表示の例

さらに、検索結果は、グラフの種類と軸を指定することで簡単に可視化することができる。次に示す例では、検索結果における特許分類記号の出現比率をパイチャートで表示している。

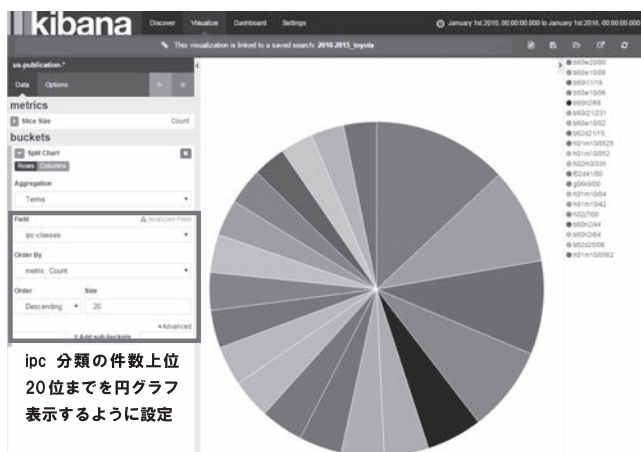


図4 検索結果の可視化の例

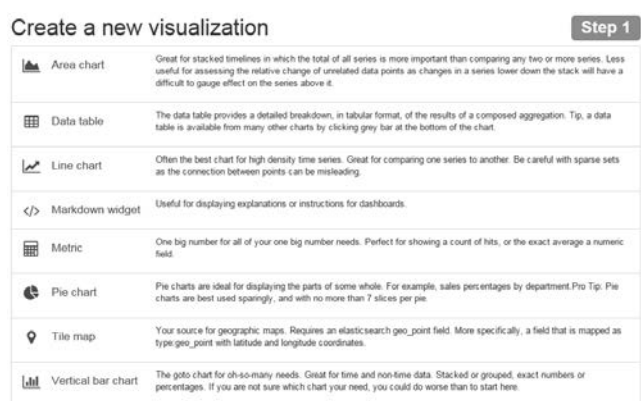


図5 Kibana で利用可能なグラフと図（上図は実際の選択画面）

(2) 類似文献検索・概念検索

Elasticsearch では、内部に自然言語処理を行うアナライザ⁽⁸⁾を保持しており、そのアナライザを使うことで文章を単語のベクトルとして保存できる。そうすることで、文書間の類似度をベクトル間の距離として内部で算出できる。この処理によって、特定の文献に類似する文献の検索や、文章を入力して、その文章と概念的に類似する文献を抽出してくる概念検索も実現することが可能である。リレーショナルデータベースを利用してシステムを構築する場合、最近では拡張機能として文書間の距離を測る演算機能を追加できるが、Elasticsearch のようにデータ登録時に自然言語処理を行っておくわけではないため、検索結果を得るまでに時間がかかってしまう。以下に、実際の検索事例を示す。公開番号 20150266504 の文献に対して、タイトルと IPC 分類が類似する検索を行うと、類似度のスコアと一緒に検索結果が返される。

公開番号	20150266504
出願人・譲渡人	TOYOTA JIDOSHI KABUSHIKI KAISHA
タイトル	STEERING INTENTION DETERMINATION DEVICE, VEHICLE CONTROL DEVICE, STEERING ASSIST DEVICE, AND STEERING ASSIST SYSTEM
IPC 分類	B62D6/04, B62D5/04

表1 検索に用いたターゲットの文献（この文献の類似文献を検索）

公開番号	タイトル	IPC 分類	スコア(*)
20150251692	MOTOR CONTROL DEVICE AND ELECTRIC POWER STEERING DEVICE	B62D5/04, H02P21/00	2.1822925
20150251691	ELECTRIC POWER STEERING DEVICE	B62D5/04	1.8672998
20150266501	STEERING CONTROL DEVICE AND STEERING SYSTEM	B62D5/04	1.6201113

(*)スコアは数値が大きいほど類似度が高い

表2 公報番号 20150266504 と類似度が高いと判断された文献

(3) あいまい検索・キーワードサジェスト

あいまい検索とは、検索式で指定したキーワードとは完全に一致しなくても候補文献を抽出する機能である。Elasticsearch では、編集距離⁽⁹⁾に基づいて類似する語句を抽出することができる。次の例では、title に「autonomos」と誤ったキーワードを指定して検索したケースであるが、検索結果には黄色でハイライトされている通り、「autonomous」と正しい語句が抽出されている。

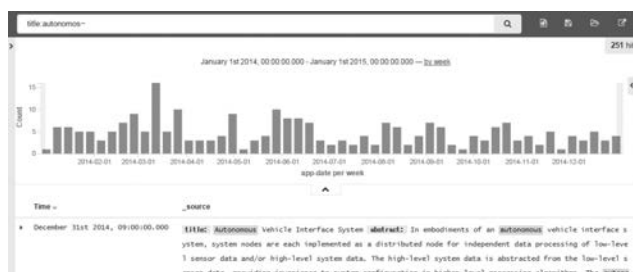


図6 あいまい検索の検索結果例

また、Kibana を通しては利用できないが、Elasticsearch 自体には検索式に用いるキーワードのサジェスト機能が搭載されている。検索エンジンで検索を行った後に表示されることのある「もしかして

～」と出てくる機能である。

(4) キーワード抽出

検索結果の文献集合の中から、特徴的なキーワードをピックアップできれば、検索式の改善に役立てることができる。ここでは、「特徴的」なキーワードであるという点が重要であり、単なる頻出語句をピックアップしてしまうと、英文であれば「and」、「the」のような接続詞や冠詞、「apparatus」、「method」のように分野に関わらず出現する語句が上位にランクインしてしまう。そこで、特徴的なキーワードをピックアップする手法として、TF-IDF⁽¹⁰⁾という手法がある。Elasticsearchでは、例えば Title や Abstract に出現する各 Word の統計情報を即時に取得できるため、それらの情報を利用することで、TF-IDF によるキーワード抽出が簡単に行える。いったん特許情報を格納した後で、Elasticsearch の機能を使って各文献のキーワードを抽出し、既存の特許情報と合わせて保存しておけば、検索結果から頻出キーワードをランキング表示させることなどが可能となる。以下に、2015年1月の公開公報の中から IPC 分類が G06Q30 (Commerce, e.g. shopping or e-commerce) に属する文献を抽出し、キーワードランキングを並べた結果を示す。重みづけなしのキーワードランキングはほぼ文献内容を表す語句がないのに対して、TF-IDF による重みづけキーワードでは、G06Q30 (Commerce, e.g. shopping or e-commerce) に関わる語句が選ばれていることが見て取れる。

Rank	重みづけなしのキーワード	TF-IDF による重みづけキーワード
1	The	User
2	A	Customer
3	And	Advertisement
4	To	Item
5	Of	Product
6	For	Information
7	An	Merchant
8	In	Transaction
9	Method	Content
10	System	Location

表3 TD-IDF による重みづけの有無によるキーワードランク比較

(5) SDI (Selective Dissemination Information) 機能

多くの商用特許情報検索・分析システムでは、予め検索式を登録しておき、該当する特許文献が新規に登録された段階でユーザーに通知される SDI と呼ばれる機能を保持している。リレーショナルデータベースを用いてシステム構築する場合には、この機能は別途自分で開発する必要があるが、Elasticsearch には最初から SDI 機能が準備されている⁽¹¹⁾。

4. 検索・分析システムの構築手順

前の章では、Elasticsearch を用いて構築した特許情報の検索・分析環境で、どのような検索・分析機能が実現できるかについて紹介した。本章では、このようなシステムをどのように構築するかという手順について簡単に説明する。手順は大きく2つのステップに分けることができ、1つめのステップは特許情報の取得である。そして次のステップが Elasticsearch の実行と取得したデータの登録である。これが終われば、Kibana を通して Elasticsearch にアクセスできるようになり、特許情報の検索・分析システムが準備できたことになる。以下、各ステップについて説明する。

1) 特許情報の入手

特許情報は、多くの場合、各国特許庁がデジタルデータで提供を行っている。今回は、入手が容易な米国特許データ (Application Red Book (Bibliographic Text)) を利用して実験を行った。Elasticsearch 自体は、多国語対応しており、日本語文献でも今回と同様の機能を実現できる。

米国では、2012年5月に発表した「デジタル・ガバメント戦略 (Digital Government: Building a 21st Century Platform to Better Serve the American People)」に基づき、数値データだけでなく文書情報等の非構造化データも対象に公開を推進している。⁽¹²⁾

特許情報に関しては、Reedtech 社が USPTO から委託を受けて、WEB サイト⁽¹³⁾にて無償公開⁽¹⁴⁾を行っている。同社の WEB サイトでは、特許公報、出願公開公報をはじめとする多くの特許情報が XML ファイル形式⁽¹⁵⁾や TIFF イメージファイルで提供されている。

Grant Yellow Book (Single-Page TIFF Images)	1790 年以降の特許公報 (Single-Page TIFF イメージファイル版)
Grant (Multi-Page TIFF Images)	1790 年以降の特許公報 (Multi-Page TIFF イメージファイル版)
Grant Red Book (Full Text with Embedded Images)	2001 年以降の特許公報 (テキストと埋め込みイメージファイル)
Grant Red Book (Full Text)	2001 年以降の特許公報 (テキストのみ)
Grant Red Book (Bibliographic Text)	2001 年以降の特許公報 (書誌事項のテキストのみ)
Grant OCR Text	1920~1979 年までの特許公報 (OCR によりテキスト化したデータ)
Official Gazettes	USPTO から公式に発行された特許公報
Application Yellow Book (Single-Page TIFF Images)	2001 年以降の公開公報 (Single-Page TIFF イメージファイル版)
Application Multi-Page TIFF Images	2004 年以降の公開公報 (Multi-Page TIFF イメージファイル版)
Application Red Book (Full Text with Embedded Images)	2001 年以降の公開公報 (テキストと埋め込みイメージファイル)
Application Red Book (Full Text)	2001 年以降の公開公報 (テキストのみ)
Application Red Book (Bibliographic Text)	2001 年以降の公開公報 (書誌事項のテキストのみ)
Assignment XML	1980 年以降の移転に関するデータ
Maintenance Fee Events	1981 年以降の維持年金に関するデータ
Patent Classification Data	最新版の特許分類に関する情報
Patent IFW Petition Decisions	2001 年以降の審決データ
PTAB and PRPS Records	1997 年以降の Patent Trial and Appeal Board と Patent Review Process System のレコード

表 4 Reedtech 社 WEB サイトから入手できる米国特許情報

2) Elasticsearch の実行とデータ登録

Elasticsearch の実行は、Windows, Mac OS, Linux のいずれの環境でも可能であり、1 台の PC 上で動作させるのであれば、Elastic 社からモジュールをダウンロードしてきて実行するだけでよい⁽¹⁶⁾。

Elasticsearch が動作した後は、特許情報の登録である。ここでは若干の手間が発生する。Reedtech 社の WEB サイトから取得した XML ファイルはそのままでは Elasticsearch に入力することはできない。Elasticsearch では JSON⁽¹⁷⁾ 形式のデータを入力するため、その変換を行う必要がある。今回、この処理は自作のプログラムで行ったが、以下のようなデータ

形式の変換を行っているだけであり、それほど複雑な処理は必要ない。

XML 形式(WIPO 標準に準拠)	JSON 形式
<pre><us-patent-application lang="EN" ... <us-bibliographic-data-application lang="EN" ... <publication-reference> <document-id> <country>US</country> <doc-number>2015000001</doc-number> <kind>A1</kind> <date>20150101</date> </document-id> </publication-reference> <application-reference appType="utility"> <document-id> ...</pre>	<pre>{ "pub-num": "2015000001", "pub-date": "20150101", "app-num": "14486383", "app-date": "20140915", "title": "Apron Having Elastic Waist and Zip Closure", "keywords": [], "applicants": [{ "applicant-name": "Jackson, SR.; Monroe", ...], "assignees": [], "usc-classes": [], "ipc-classes": ["A41D13/04", "A41D13/00"], ...</pre>

図 7 取得した XML 形式特許データと Elasticsearch 入力用 JSON 形式データの比較

以上の 2 ステップで、環境構築は完了であり、Kibana を実行すれば WEB 経由で既に紹介したような特許情報の検索・分析環境が利用可能となる。XML 形式から JSON 形式に変換する必要があるが、従来型のデータベースへデータ登録する手間を考えると、ここは格段に簡易化されている。

5. むすび

今回、Elasticsearch を用いることで、多くの検索・分析機能を容易に実現できることを紹介した。各国特許庁が提供する特許・実用新案の無料情報提供サービスの機能比較⁽¹⁸⁾を見ると、現状ではサービスごとの差異があるものの、そのほとんどは今回紹介したような OSS を活用することで実現することができ、将来的には機能差がなくなっていくものと考えられる。OSS の世界では、製品開発時のコスト構造において、Unique Innovation (商品の競争優位性を生み出す価値)、Delivery Support For Innovation (きわめて重要な機能であるが直接的に顧客にとっての Value を生まない機能)、Commodity Platform (ボルト、ナットのような外部調達する部品) の 3 つに分類し、Delivery Support For Innovation を OSS が担うとしている⁽¹⁹⁾。言い換えると、OSS が担っている機能は、既に Unique Innovation 領域ではなく、この領域で競争すべきではないということになる。現在の技術進歩は速く、これまで Unique Innovation と考えられていた領域が、いつのまにか OSS によって Delivery Support For Innovation となっていることも多い。情報検索・分析機能の多くが OSS によって普及してしまっている状況を見ると、今後はユーザビリティや複

数のデータソースの連携（例えば特許情報、企業情報、訴訟記録）など、別の側面で競争が本格化すると考えられる。

以上

注

- (1) 2012 年の ILSVRC (Imagenet Large Scale Visual Recognition Challenge) でトロント大学が開発した画像認識アルゴリズム SuperVision がエラー率 15% から 16% と、それ以下のアルゴリズムのエラー率 26% から 29% に対して大きく差を開けて話題となった。(松尾豊, 人工知能は人間を超えるか, 角川 Epub 選書, p144)
- (2) CAFFE は 2016 年 2 月現在, BSD-2 Clause ライセンスで配布されている。
- (3) SDI とは Selective Dissemination of Information の略で, データベース更新時に予め設定した検索式を実行して検索結果を伝える機能のことである。
- (4) NASA の発表によると, 宇宙から取得したデータの分析を OSS 活用により行っているとある。その OSS の一つとして Elasticsearch も挙げられている。
(<http://www.nasa.gov/content/exploring-space-through-streaming-analytics/>)
- (5) “World Patent Indicators - 2015 Edition”, WIPO, p25 によると 5 大特許庁 (中国, 米国, 日本, 韓国, 欧州) のうち, 日本以外は特許出願数が 1990 年以降は上昇傾向にある。例えば米国では, 2000 年には年間 30 万件程度の出願数であるが, 2014 年には年間 60 万件に届く勢いで増加してきている。
- (6) 特許情報サービスを手掛ける LexisNexis 社では, 既に Elasticsearch の活用を始めている。しかし, 同社では特許情報と訴訟情報の連携など, OSS 活用によって開発費用が抑えられた分, 別の付加価値を追加するというアプローチをとっている。OSS 活用によって低価格化が進むとは必ずしも言えないことを示している。ベンチャーなどの新規参入企業が OSS 活用でサービスインする際には低価格化路線が期待できる。
- (7) Eric Redmond and Jim R.Wilson, 7つのデータベース7つの世界, Ohmsha, p3~p7 によると, リレーショナルデータベース以外にはキーバリュー型, 列志向型, ドキュメント志向型, グラフ, ポリグロットという分類が行われている。特許情報を格納する場合にはネスト構造が可能で柔軟性の高いドキュメント志向型データベースが開発を行う上でのハードルが低いと考えられる。
- (8) 英文に関してはスペースによる Word の分割, 大文字小文字の統一, 活用語尾を統一化するステミングなど, 自然言語処理に必要な一通りの機能がそろっている。また, 日本語文

に関しても, kuromoji という日本語の自然言語処理を行う OSS を利用した外部プラグインが用意されており, 形態素解析 (日本語の文章の場合, 英文のようにスペースで単語が区切られていないため, 区切れを判別する処理が必要となるが, そのような処理を形態素解析と呼ぶ) などを行う事ができる。

- (9) 2つの文字列がどの程度異なっているかを表す距離であり, 文字の挿入・削除・置換によって一方の文字列を他方の文字列に置き換えるために必要な手順の回数として距離が定義されている。レーベンシュタイン距離ともいう。
- (10) TF-IDF は, TF (Term Frequency) と IDF (Inverse Document Frequency) という 2つの尺度の組み合わせである。TF はある文書中に出現頻度の高い語句に重みを加える尺度であり, IDF は多数の文書に出現する語句より少数の文書に出現する語句に重みが加わる尺度であり, これらを合わせて用いることでその文書特有の語句に重みが加わるようにしている。詳細については, 奥村学・藤井敦・谷川英和・岩山真・難波英嗣・山本幹雄・内山将夫, 特許情報処理: 言語処理的アプローチ, コロナ社, p19 に説明されている。
- (11) Elasticsearch ではこの機能は Percolator と呼ばれている。
- (12) 総務省編, 平成 25 年版情報通信白書, p204
- (13) Reedtech 社 WEB サイト (<http://patents.reedtech.com/patent-products.php>)
- (14) Interface2016 年 1 月号, CQ 出版, p35 によると, 「米国の場合, 連邦政府が作成したものはすべてパブリックドメインとなるため, データのライセンスは必要なく, 自由に利用が可能です」とある。
- (15) XML ファイルのフォーマットについては WIPO が定める勧告・ガイドラインに準拠している。(http://www.wipo.int/standards/en/part_03_standards.html)
- (16) Elasticsearch の WEB ページに, モジュールのダウンロード方法, インストールステップが説明されている。https://www.elastic.co/downloads/elasticsearch
- (17) JavaScript Object Notation のことを略して JSON と呼ぶ。JSON は, XML と同様, データを記述する言語であるが, XML と比較してシンプルに記述できるため, WEB API でデータをやり取りする場合などによく使われている。
- (18) 特許庁又は INPIT が運営する公的な特許情報提供サービスのあり方, 第 2 回情報普及活用小委員会配布資料, p5
(http://www.jpo.go.jp/shiryoku/toushin/shingikai/pdf/jouhou_fukyu02_shiryoku/03.pdf)
- (19) 福安德晃, オープンソース経済モデル, オープンソースカンファレンス 2011 Tokyo/Spring 発表資料, p11
(http://www.ospn.jp/osc2011-spring/pdf/osc2011spring_the_linux_foundation.pdf)

(原稿受領 2016. 2. 26)