メタデータの参照関係とスキーマに基づく LOD 間の類似性および併用可能性算出手法

筑波大学

図書館情報メディア研究科

2020 年 3 月

山中 勇樹

目次

図目次

表目次

1. はじめに

近年，政府や地方自治体をはじめとして世界各国の様々な団体が災害情報や施設の情報など様々なデータを誰もが自由に閲覧，利用可能なオープンデータとして公開している[1][2]．オープンデータは Web アプリケーション開発や統計分析など様々な用途で使用されており，オープンデータの利用を推進するコンテストなども開催されている[3]．オープンデータの中でも特に機械可読な共通の形式で記述される Linked Open Data(LOD)が注目されており，様々な団体や大学などで LOD のデータセット(LOD データセット)が公開されている[4][5]. LOD データセットは Resource Description Framework(RDF)という共通の形式で記述される．この RDF 形式で記述されたデータは拡張性が高く，外部のデータを自由に参照することができる．また，専用のクエリ言語を使うことによって環境に依存せず，自由にデータを取り出すことが可能である．そのため，LOD データセットは他のデータセットや WebAPI と組み合わせることにより幅広い用途で使用することができる．

LOD データセットは利用者が独自に開発するもののほか，Web 上で公開されている既存の LOD データセットを再利用し，それらを組み合わせて使われる.新規 LOD データセットの作成にはコストがかかるほか，データのクオリティや信頼性の観点から可能な限り既存のものを再利用することが望ましい. LOD データセットはそれぞれが独自に公開している Web サイトで閲覧できるほか，政府や地方自治体など様々な団体が公開しているデータカタログサイトを用いて検索することができる．データカタログサイトでは各 LOD データセットについての詳細な情報を閲覧することができるが，そのようなサイトにおいて類似するものや組み合わせて使用可能なものといった関連する他の LOD データセットの情報を提示しているものは少ない. LOD データセットは具体的なデータを閲覧するためにファイルのダウンロードや専用の API を用いた問い合わせが必要であるが，RDF の知識がない場合どのようなデータが記述されているのかわかりにくい．また，データカタログサイトではデータセットの検索はできるが，そのデータセットがどのように使われているかといった事例を閲覧できるものは少ない．そのため，具体的にどのように使えば良いかがわかりにくく，複数のものを組み合わせることが容易という LOD データセットの利点が十分に生かされていないといえる．この問題を解決するために，どのような LOD データセット同士が組み合わせて使用できるかといった情報が有用ではないかと考えた．

本研究では，組み合わせて使用可能な LOD データセットの発見を目的として LOD データセット間の類似性や組み合わせて使用できるかを決定する併用可能性の算出手法を提案する. LOD データセットの構造を定義するメタデータスキーマや異なる LOD データセット間の参照関係に着目し，異なる LOD データセット間における類似性と併用可能性を算出

する. LOD データセットの名前や概要, それぞれの LOD データセットに付与されているタグとメタデータスキーマで使用される特徴的な単語を抽出する. 抽出した単語から LOD データセットの特徴を表すベクトルを生成し, それらの類似度を図ることで LOD データセット間の類似性を算出する. また, LOD データセット間の参照関係とトピックの共通性から異なる LOD データセット間の併用可能性を算出する. 本論文ではまず 2 章で LOD に関する研究背景について述べる. 3 章で LOD の利用と探索の問題について述べ, 4 章で問題を解決するための類似性・併用可能性算出手法について述べる. 5 章と 6 章で類似性と併用可能性の評価実験についてそれぞれ述べ, 7 章で考察を述べる.

2. Linked Open Data(LOD)とメタデータ

2.1 Linked Open Data(LOD)

様々な分野のデータを機械可読な共通の形式で記述することにより，それらの相互運用を可能にする取り組みを Linked Data と呼ぶ. Linked Data は Tim Berners-Lee によって提唱された[6]以下の 4 原則を基準としている.

(1) Use URIs as names for things.
(2) Use HTTP URIs so that people can look up those names.
(3) When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL).
(4) Include links to other URIs. so that they can discover more things.

この原則に従っているデータの中で，誰でも利用可能なライセンスで公開されているデータのことを特に Linked Open Data(LOD)と呼ぶ. LOD は分野を問わず Web 上で識別可能な事物についてのメタデータを記載したものである．メタデータとは「データについてのデータ」を意味する言葉であり，対象についての属性(項目)とその値を記述したものである．

LOD として公開されているものの例として，Wikipedia の記事で記述されている事物の情報を機械可読な形式に変換した DBpedia[7]やヨーロッパの芸術作品に関する情報を検索可能にした Europeana[8]などが挙げられる． そのほか，LOD をはじめとした様々な形式のデータセットを公開するためのデータカタログサイトも多数存在しており，世界各国の政府など様々な団体が個別にデータカタログサイトを公開している．また，2013 年に行われた G8 サミットでは政府が所有するデータを LOD の形式で公開することを推奨する「オープンデータ憲章[9]」が合意されるなど，その発展に期待をされている．

図 1 は LOD の中でも主要なデータセットの参照関係を可視化した「LOD cloud[10]」である．それぞれのノードがデータセットを示し，ノードの色はデータセットの分野，エッジはデータセット間の参照関係を表している．この図から様々な分野で LOD が利用されていること，非常に多くのデータセットが分野を越えて直接的，または間接的にリンクしていることが分かる．表 1 は LOD cloud に登録されているデータセットの数の遷移を表したものであり，LOD の取り組みが始まった 2007 年から大きく増加していることが分かる．なお，本論文では LOD に関する取り組みのことを「LOD」，LOD の枠組みに則って記述されたデータセットを LOD データセットと記述する．

表 1 : LOD Cloud に登録されているデータセット数の推移

日付	データセット数
2007/05/01	12
2008/02/28	32
2009/03/05	89
2010/09/22	203
2011/09/19	295
2014/08/30	570
2017/01/26	1146
2018/04/30	1184
2019/03/29	1239

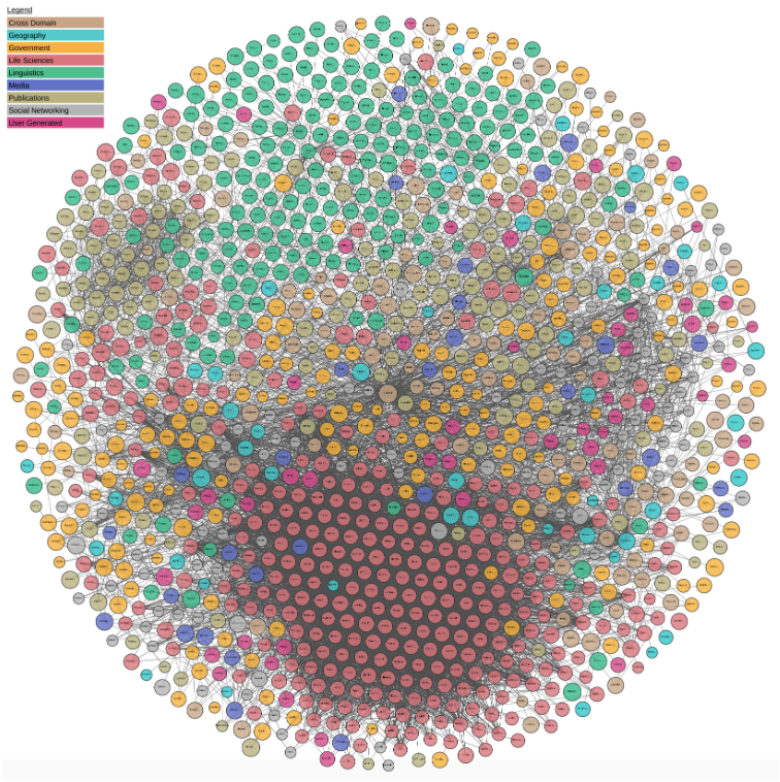

図 1 : 最新版 LOD Cloud(引用元[10])

2.2 Resource Description Framework (RDF)

LOD データセットは様々な分野のデータを機械可読な共通の形式で記述することによって相互運用性を高めている.

この共通の形式として「Resource Description Framework(RDF)[11]」が用いられる.

RDF は Web 上で識別可能なあらゆる事物(リソース)とその属性を表現するためのデータ形式である.RDF では主語・述語・目的語の 3 つからなる文を基本単位としている. 主語は対象とするリソース, 述語はその属性, 目的語は属性に当てはまる値をそれぞれ示す. この文はトリプルと呼ばれる. このトリプルをグラフで表した例が図 2 である. この図は「『学問のすゝめ』の著者は福沢諭吉である」ということを表現している.RDF グラフではリソースを楕円(ノード), リソースではない単なる文字列(リテラル)を矢印で表し, 目的語がリソースではなくリテラルであった場合はノードではなく長方形で表現する. また, RDF では図 3 のようにトリプルの目的語がリソースとなっている場合, それを主語にした別のトリプルを連結することも可能である. このようにトリプルを連結した RDF グラフを構成することにより, リソースに関する複雑なデータを表現することが可能になる.

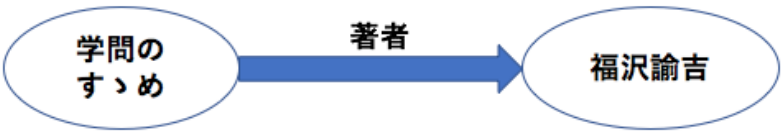

図 2 : RDF トリプルの例

図 3 : RDF トリプルを連結した例

2.3 メタデータスキーマ

メタデータとはデータについてのデータを表した言葉であり，テレビの番組情報やペットボトルのラベルなどあらゆる場所で使用されている．これらの情報では基本的にどのような項目があるか，それらの項目にはどのような値を記述すればよいかといった標準をそれぞれの分野で定め，その基準に従ってメタデータを作成する．このように複数の団体や企業，個人が共通の形式でメタデータを記述するための規則を決定するものをメタデータスキーマと呼ぶ．

RDF 形式のデータにおいてそれらのデータを記述する際の項目や値を決定するメタデータスキーマを定義することにより，人や機械がメタデータの構造や項目の役割を理解することができるようになる．そのため，メタデータスキーマを定義することでメタデータの再利用や他者の利用も容易になる．メタデータスキーマはメタデータを記述するための一連のターム(メタデータターム)を定義する語彙定義とメタデータの構造を定義する記述規則からなる．

2.3.1 語彙定義

RDF は機械可読な表現形式であるため，機械がそれぞれのメタデータにおいて記述された属性の意味を識別しそれを共有するための仕組みが必要となる． 例として図 3 のトリプルで用いられている「著者」や「生年月日」などの述語はそのままでは機械的に処理することはできないため，これらの意味を定義し一意に識別可能にする必要がある．このメタデータ記述の際に使われる用語を定義したものをメタデータ語彙と呼ぶ．メタデータ語彙は RDF におけるトリプルの目的語を表すプロパティターム(以下，プロパティ)とリソースがどのようなものかを分類するクラスターム(以下，クラス)からなる．これらのタームは機械的に処理するために一意な URI が付与されている．例として，図 3 の著者には「http://example.com/author」などの一意な URI を付与することで機械的な処理が可能となる．この URI で表現されたタームの項目名やどのような値を記述すれば良いかといった仕様を定義しているものがメタデータ語彙定義(以下，語彙定義)である．

メタデータ語彙はメタデータの作成者が個人で自由に定義できるほか，様々な団体が定義し，公開しているものを自由に使用することができる．表 2 は分野を問わず，様々な用途のメタデータの記述に使用することができるメタデータ語彙である Dublin Core Metadata Element Set(DCMES)[12]のプロパティの一覧である．

メタデータタームに付与されている URI はそれぞれのタームを識別するためのローカル名とメタデータ語彙それぞれに共通して付与されている名前空間からなる．この URI は名前空間の部分を名前空間接頭辞というもので置き換えた XML 仕様の修飾名(QName)で

表 2 : DCMES のプロパティ一覧

ラベル	説明
Contributor	リソースの内容に貢献している個人，組織，またはサービス
Coverage	リソースが関わる地理的または時間的な範囲
Creator	リソースの作成に責任を持つ個人または団体
Date	リソースのライフサイクル内に関連する日付
Description	リソースの概要
Format	リソースのファイル形式や物理的な形態
Identifier	リソースの識別子
Language	リソースの言語
Publisher	リソースを使用可能にしている個人または団体
Relation	リソースと関連する他のリソース
Rights	リソースの権利関係
Source	リソースの派生元の関連リソース
Subject	リソースの主題
Title	リソースの名前
Type	リソースの性質やジャンル

表すことも可能である．例として「http://purl.org/dc/elements/1.1/contributor」という URI は「http://purl.org/dc/elements/1.1/」の部分を「dc:」に置き換えて「dc:contributor」と記述することが可能である．

2.3.2 記述規則

メタデータを記述する際，記述するための対象とその対象に与える項目，それらの項目に記述する値の制約などを定めたものがメタデータ記述規則(以下，記述規則)である．記述規則を定義し，公開しておくことによって他者がそのメタデータを利用する際に，そのメタデータがどのようなものなのかを理解することが容易になるほか，新たなメタデータを記述する際に他のメタデータの記述規則を参照することによって形式を統一し，組み合わせて使うことが容易になる．記述規則における項目の代表的な制約は以下のようなものが挙げられる．

- 記述項目名
  項目に与える名前. 例としては「title」「creator」など.
- ターム
  その項目を記述する際に用いるメタデータターム. 例として「title」という項目には「dc:title」を用いるなど.
- 省略可能性
  その項目が必須であるか, 記述しなくても良いかを定義する.
- 繰り返し条件
  その項目は繰り返し記述することが可能かどうかを定義する. 例として「title」は繰り返し不可能だが, 「creator」は繰り返し可能など.
- 値制約
  その項目を記述する際に与えるデータ型やメタデータ語彙のクラスを定義する. 例として「title」には文字列を与え, 「creator」には「Person」クラスを与えるなど.

記述規則もまた様々な団体によって提唱・モデル化されたものが使用可能であり, 一例としてシンガポール・フレームワークによってモデル化された「a Dublin Core Application profile[13]」(以下アプリケーションプロファイル)が挙げられる. アプリケーションプロファイルはメタデータの相互運用性を最大限にするための枠組みであり, この規則では複数のメタデータ語彙を選択して使用することが可能である. これによって既存のメタデータ語彙で再利用可能なものと既存のメタデータ語彙では表現できないものを定義した独自語彙を組み合わせることで容易にメタデータを記述することが可能になる. アプリケーションプロファイルは以下の 5 つの要素からなる.

- 機能要件(Functional requirements)[必須]
  アプリケーションプロファイルにおいて対象となる機能と対象外となる機能を定義
- ドメインモデル(Domain model)[必須]
  アプリケーションプロファイルで記述する基礎的な実体とそれらの主要な関係を定義
- 記述セットプロファイル(Description Set Profile: DSP)[必須]
  使用するメタデータ語彙やその値などの記述規則を定義
- 利用ガイドライン(Usage guidelines)[選択]
  アプリケーションプロファイルの適用方法や, 適用時に意図されているプロパティの使用方法などを記述

- 符号化構文ガイドライン(Encoding syntax guidelines)[選択]
  規定するアプリケーションプロファイル固有の構文や構文ガイドラインを記述

2.4 メタデータの参照関係と相互運用性

RDF 形式のメタデータでは主語・述語・目的語からなるトリプルで対象についての情報を記述するが，この際，トリプルの目的語を他のデータセット内のリソースにすることも可能である．このようにトリプルを用いて他のデータセットの情報を参照することを本研究では「RDF リンク」と呼ぶ．図 4 は RDF リンクの例である．この例ではデータセット A で記述している書籍「学問のすゝめ」がプロパティ「dc:creator」を用いることによってデータセット B で記述されている「福沢諭吉」の情報を参照している．この RDF リンクで 2 つのデータセット内のリソースを結びつけることにより，LOD の情報を収集するためのクローラやその他の Web アプリケーションでデータセット A 内の「Book01」の情報を取得する際，Book01 に付与された「dc:creator」のプロパティをたどることでデータセット B に含まれる「PersonA」の情報を合わせて取得することが可能となる．そのため，RDF リンクを用いることによって 2 つのデータセットを同時に利用することが容易になる．LOD ではこの RDF リンクを用いて Web 上に公開されている様々な情報を RDF リンクによって結びつけることによって，異なるデータセットに含まれているメタデータのアクセスが可能になり，メタデータの相互運用性を高めている．

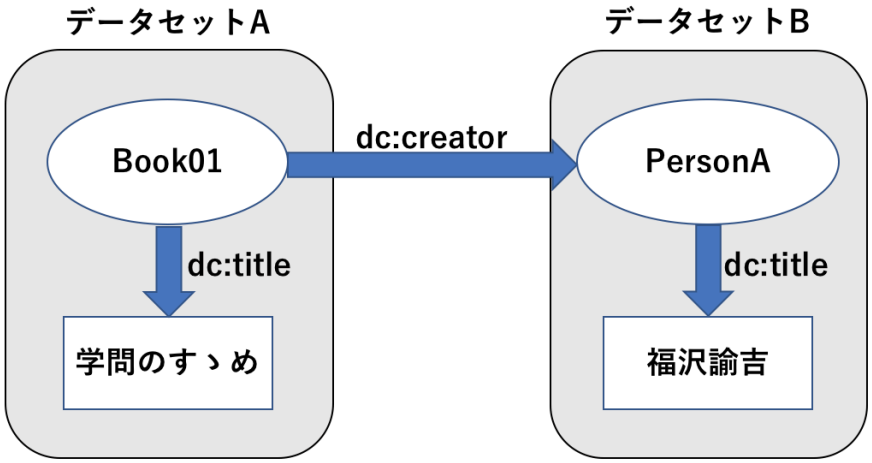

図 4 : RDF リンクの例

3. LOD を利用した Web アプリケーション開発とその問題

3.1 LOD を利用した Web アプリケーション開発

LOD データセットは Web アプリケーション開発やデータ分析など様々な用途で使用することができる. Web アプリケーション開発などに利用する際は CSV や JSON など他の形式のデータを用いることもできる. しかし, LOD データセットは全てが機械可読な共通の形式で記述されており, RDF リンクにより関連する外部のデータセットにも容易にアクセスすることができる. そのため, LOD データセットは他の形式のデータと比較して, 複数のものを容易に組み合わせることができるという利点がある. LOD データセットは分野を問わず共通して機械可読な RDF 形式で記述される. また, RDF リンクによって一つの LOD データセット内のリソースから関連する LOD 複数のデータセットをたどることも可能である. そのため, LOD データセットであれば, 複数のデータセットを結びつけた Web アプリケーション開発を容易に行うことができる. 大規模な LOD データセットであれば, その活用事例を紹介しているケースもある. 例として, 横浜市が提供しているヨコハマ・アート・LOD[14]では実際に他の LOD データセットと組み合わせて Web アプリケーション開発を行なっている事例を紹介している(図 5).

LOD の利用を推進する活動も多数行われている. 日本国内では「Linked Open Data Challenge(LOD チャレンジ)[15]」など LOD やそれを利用した作品のコンテストが複数開催されている. LOD チャレンジではデータセットの作成のほか, それらを利用した分析や Web アプリケーションの開発など 5 つの部門で作品が募集されており, 政府の統計情報を

図 5：ヨコハマ・アート・LOD[14]の事例紹介

LOD 化した「統計 LOD[16]」や日本語版の DBpedia である「DBpedia Japanese[17]」などの大規模なデータセットがパートナーリソースとして提供されている．また日本国外でも Europeana などのデジタルコレクションの活用を図るプロジェクト「TuEuropeana」が展開される[18]など，幅広い活動が行われている．

3.2 既存の LOD の探索とその問題

LOD を用いて Web アプリケーション開発などを行う場合，開発者が独自に作成した LOD データセットのほか，Web 上に公開されている多数の既存の LOD データセットを組み合わせて使用する．LOD データセットを自分で作成する場合，元となるデータを自分で用意しなければならないほか，RDF の知識や他の形式からのデータの変換が必要となるため，時間的なコストがかかる．また，RDF 形式で記述されたデータは専用のクエリ言語である「SPARQL[19]」を用いることにより，環境によらず様々な言語で利用できるため再利用性が高い．それに加え，既に Web 上で公開されている LOD データセットはデータセットのクオリティが高く，データの典拠も信頼性の高いものが多い．これらの理由から LOD を用いて Web アプリケーション開発などを行う際は Web 上に公開されている既存の LOD データセットを用いることが望ましい．

既存の LOD データセットは政府や地方自治体など様々な団体が公開しているデータカタログサイトで検索することができる．データカタログサイトには LOD をはじめとした様々な形式のデータが登録されており，データセットの検索やオープンデータの活用事例の閲覧などを行うことができる．LOD データセットを検索することができるデータカタログサイトの例としては「Data.gov[1]」や「LinkData.org[20]」などが挙げられる．「Data.gov」はアメリカ合衆国の政府が所有するデータを様々な形式で公開しているサイトであり，CSV や JSON のほか RDF 形式のデータも 10,000 件以上公開されている．「LinkData.org」は日本国内のデータカタログサイトであり，誰でも自由にデータセットを作成・公開することができるほか，それらを利用した Web アプリケーションを公開することも可能であり，個人だけではなく様々な地方自治体が所有する公共のデータも LOD データセットとして公開されている．また，Google においてもデータセットの検出や検索ツールの提供が行われており[21]，CSV をはじめとしたテーブルデータやデータをキャプチャした画像など形式やジャンルを問わず，データセットとみなされるものは幅広く検索することができる．

現在，様々な団体が多くのデータカタログサイトを公開しているが，これらのサイトの多くはある特定のデータセットにおいて類似するものや併用して開発に利用できるものといった関連する他のデータセットの情報を探索することは難しい．データカタログサイトを用いることでデータセットの検索自体は可能だが通常の文書検索とは異なり，どのよう

なデータが含まれているかといった具体的なデータセットの中身を検索することはできない．そのため，LOD データセットの具体的な内容を知るためには SPARQL を用いた問い合わせやデータセットそのもののダウンロードをする必要がある．SPARQL を用いることで環境によらず，データを利用することができるが，SPARQL を利用する場合 SPARQL の文法のほか，対象となる LOD データセットのメタデータスキーマを理解している必要がある．しかし，メタデータスキーマを公開していない LOD データセットも多く存在するため，データセットの構造がわからず，目的の情報を得るためにどのようなクエリを書けばよいかといったことがわかりにくい場合も多い．また，SPARQL で検索するための Web API である SPARQL エンドポイントを公開していない LOD データセットも多く存在する．そのような LOD データセットを利用する場合は利用者がその LOD データセットのダンプファイルをダウンロードし，自分で用意した SPARQL エンドポイントにダンプファイルをアップロードする必要がある．そのため，LOD データセットを利用する場合は学習コストと手間がかかり，単一のデータセットを利用する場合 CSV や JSON などほかの形式のものを用いた方が良い場合も多い．

Web アプリケーション開発に LOD を用いる利点は上述した通り，複数のデータセットを容易に組み合わせることができる点である．しかし，関連するほかのデータセットの情報を提示しているサイトは少なく，利用者が自分で関連するデータセットを見つけることは難しい．そのため，LOD をアプリケーション開発に利用する利点は十分に生かされてないのが現状である．これは，LOD データセットの利用事例が未だ少なく，どのような LOD データセットが実際に併用されているかという例があまり公開されていないことが原因として考えられる．そのため，どのような LOD データセットが併用可能かを決定する明確な指針が存在せず，開発者がどのようなデータセットを組み合わせれば良いかが分かりにくい．この問題を解決するために，どのような LOD データセットが併用可能かを算出し，類似しているものや組み合わせて使用可能なものを提示することが有用ではないかと考えた．具体的にどのような LOD データセットが併用可能かを決定し，それを提示することによって開発者が LOD データセットを利用する際の手がかりとなり，開発をより効率化することができると考えられる．

本研究では併用可能な LOD データセットの発見を目的として，異なる LOD データセット間における類似性や併用可能かどうかを決定する併用可能性の算出手法を提案する．これにより，どのような LOD データセットが併用可能かを明確にすることができる．LOD データセットの参照関係とメタデータスキーマに着目し，異なる LOD データセット間の類似度や併用可能性を算出する．より詳細な手法については次章以降で述べる．

3.3 関連研究

3.3.1 LOD の探索を支援する研究

LOD のデータセットの探索に関連する研究としては「LOD4ALL[22]」や「LODStats[23]」が挙げられる.

LOD4ALL は富士通研究所が 2019 年まで公開していた LOD を利用するためのプラットフォームであり, 現在はサービスを停止している. 世界中で公開されている LOD データセットを収集し, それらを一つの SPARQL エンドポイントに格納することにより, 複数の LOD データセットに含まれているデータを横断的に検索することができる. また, LOD データセット内に含まれているインスタンスの検索も可能であるほか, 各 LOD データセットの統計情報なども閲覧することが可能である. LOD に関して多くの情報を検索することが可能なプラットフォームであるが, LOD データセットそのもの関係性には着目していない点が本研究とは異なる.

LODStats は Agile Knowledge Engineering and Semantic Web (AKSW)が公開している LOD データセットの統計情報をまとめたサービスである. 単一の LOD データセットの統計情報のほか, 多数の LOD データセットで用いられているメタデータタームの情報やデータセットに含まれているリソースの名前空間に着目し, それらのリンクの情報なども提示している. LOD に関して多くの視点からの情報を得ることができるサービスであるが, このサイトも LOD データセットそのものの関係性には着目していない.

3.3.2 先行研究

本研究の先行研究としては山中[24]が挙げられる. この研究では LOD データセットの使用例に着目し, LOD データセットの情報に加え, それを用いて開発された Web アプリケーションの情報を収集しそれらの検索を可能にするシステムを開発した. このシステムを用いることで特定の LOD データセットの情報を閲覧した際, 図 6 のようにその LOD データセットを用いて開発された Web アプリケーションの一覧と開発の際に併用された LOD データセットの情報を合わせて閲覧することができる. これにより, LOD データセットがどのように利用されているか, どのような LOD データセットが合わせて使用されているかがわかりやすくなる.

この研究では実際に Web アプリケーションの開発に利用された事例に着目して, 併用可能な LOD データセットの提示を行なっている. しかし, 現状 LOD を利用して開発された Web アプリケーションの事例の数は非常に少ないため, 多くの LOD データセットが検索の対象外となるという問題がある. そのため, 実際の開発にはまだ使用されていない LOD データセットであっても発見することが可能になるように, 開発事例とは異なる視点で

LOD データセットの探索支援を行うことが必要だと考えた．本研究では開発事例の情報は使用せず，LOD データセットに関する他の情報を用いて LOD データセットの探索を支援するための手法を提案する．

図 6：山中[24]のシステムにおける LOD データセットの情報提示の例

4. LOD 間の類似性・併用可能性算出手法の提案

4.1 類似・併用可能な LOD の提示による Web アプリケーション開発支援

併用可能な LOD データセットを発見するためには，具体的にどのような LOD データセットが組み合わせて使用することができるかを整理する必要がある．組み合わせが可能なデータセットとしてはトピックや構造・扱っている時間的・空間的範囲に共通性のあるものやデータセット内のリソースに参照関係があるものが挙げられる．このようなデータセット同士を組み合わせることによって Web アプリケーションで行うサービスの範囲の拡張やより詳細な情報の取得を行うことができる．

例えば，鯖江市内 AED 設置場所[25]」というデータセットと「流山市 AED 設置場所[26]」というデータセットは扱っている地理的範囲は異なるが扱っているトピックやデータの構造は類似している．具体的な RDF の構造はそれぞれ図 7 と図 8 のようになっており，メタデータタームの名前空間などは異なるが AED というトピックと設置されている施設の名前と住所，緯度・経度という項目は一致している．そのため，これらのデータセットや他の類似するデータセットを地図などと併用することで，より広い範囲で AED の設置場所を検索する Web アプリケーションを製作することが可能である．

```
<http://www3.city.sabae.fukui.jp/xml/aed#1>
      <http://linkdata.org/property/rdf1s284i#name> "鯖江高等学校"@ja ;
      <http://linkdata.org/property/rdf1s284i#address> "鯖江市舟津町２丁目５－４２"@ja ;
      <http://linkdata.org/property/rdf1s284i#count> "1"^^xsd:int ;
      <http://www.w3.org/2003/01/geo/wgs84_pos#lat> "35.938162"^^xsd:float ;
      <http://www.w3.org/2003/01/geo/wgs84_pos#long> "136.183918"^^xsd:float .

<http://www3.city.sabae.fukui.jp/xml/aed#2>
      <http://linkdata.org/property/rdf1s284i#name> "丹南高等学校"@ja ;
      <http://linkdata.org/property/rdf1s284i#address> "鯖江市熊田町１０－７"@ja ;
      <http://linkdata.org/property/rdf1s284i#count> "1"^^xsd:int ;
      <http://www.w3.org/2003/01/geo/wgs84_pos#lat> "35.953576"^^xsd:float ;
      <http://www.w3.org/2003/01/geo/wgs84_pos#long> "136.16883"^^xsd:float .
```

図 7：鯖江市内 AED 設置場所[25]のデータセットの構造

```
<http://linkdata.org/resource/rdf1s648i#1>
      <http://www.w3.org/2000/01/rdf-schema#label> "1"@ja, "流山市消防本部・中央消防署"@ja ;
      <http://linkdata.org/property/rdf1s648i#address> "流山市三輪野山1-994"@ja ;
      <http://www.w3.org/2003/01/geo/wgs84_pos#lat> "35.8660916"^^xsd:float ;
      <http://www.w3.org/2003/01/geo/wgs84_pos#long> "139.9030043"^^xsd:float ;
      <http://linkdata.org/property/rdf1s648i#telephone> "04-7158-0119"@ja .

<http://linkdata.org/resource/rdf1s648i#2>
      <http://www.w3.org/2000/01/rdf-schema#label> "2"@ja, "東消防署"@ja ;
      <http://linkdata.org/property/rdf1s648i#address> "流山市前ヶ崎449-1"@ja ;
      <http://www.w3.org/2003/01/geo/wgs84_pos#lat> "35.8466701"^^xsd:float ;
      <http://www.w3.org/2003/01/geo/wgs84_pos#long> "139.9339132"^^xsd:float ;
      <http://linkdata.org/property/rdf1s648i#telephone> "04-7146-0119"@ja .
```

図 8：流山市 AED 設置場所[26]のデータの構造

また，「温泉宿・適応症 LOD[1]」は日本各地の温泉や宿，その泉質や適応症などを記述したデータセットだが，同時にそれらを擬人化したキャラクターについても記述されており，図 9 のようにそのキャラクターの声優の情報を「声優 LOD[2]」から参照している．この 2 つのデータセットを併用することにより，温泉を擬人化したキャラクターについてより多角的な情報を得ることができる．

このように LOD では異なる複数のデータセットを併用することにより，扱える情報の量やサービスの範囲を増やすことが可能になり，様々な Web アプリケーションの開発に応用することができる．しかし，異なる複数のデータセットを併用する際はそれらの間に何らかの共通項が必要となる．上述した 2 つの例において AED を検索するアプリケーションでは地理的な範囲が異なるが，AED というトピックが共通しているほか，構造が類似している．また，温泉を擬人化したキャラクターに関する Web アプリケーションでは主として扱っているトピックは異なるがキャラクターと声優というトピックは近似しており，この 2 つのデータセットは参照関係がある．このように併用可能な LOD データセットには扱っているトピックや構造の類似性やデータセット内のリソース間における参照関係が見られる．そのため，トピックや構造の類似性と参照関係に着目することによって併用可能な LOD データセットを発見することができると考えた．

本研究では，主として扱っているトピックや構造が類似しているかどうかの指数を LOD データセット間の類似性とする．また，参照関係に着目し Web アプリケーションを開発する際に併用できるかどうかを決定する指標を LOD データセット間の併用可能性とする．

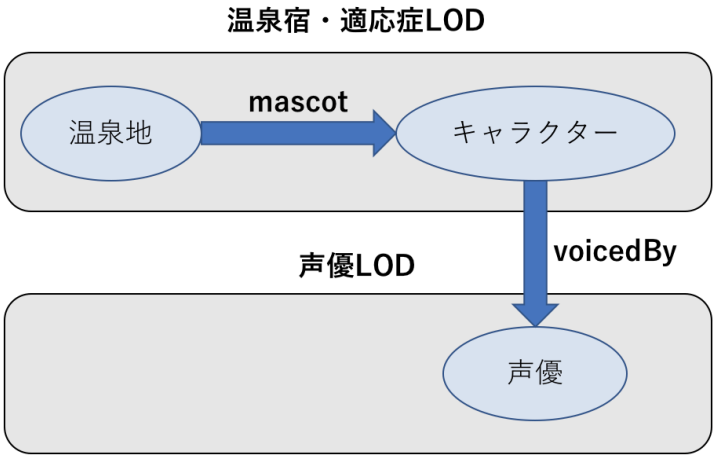

図 9：異なる LOD データセット間の参照関係

[1]http://mdlab.slis.tsukuba.ac.jp/lodc2019/onsen/

[2] http://idea.linkdata.org/idea/idea1s2433i

次節以降ではこの類似性と併用可能性の算出手法を提案する.

4.2 LOD 間の類似性算出手法

LOD データセット間の類似性はその LOD データセットのトピック及び構造がどれほど類似しているかによって決定する. この LOD データセット間の類似性を表す式を以下のように定義する.

$$\mathrm{Sim}(dsA, dsB) = \alpha * \mathrm{SimTopic}(dsA, dsB) + (1-\alpha) * \mathrm{SimTerms}(dsA, dsB) \quad \text{数式 1}$$

Sim(dsA, dsB)は 2 つの LOD データセット dsA, dsB の類似性, SimTopic(dsA, dsB)は LOD データセット dsA, dsB が扱っているトピックの類似性, SimTerms(dsA, dsB)は LOD データセット dsA, dsB の構造の類似性をそれぞれ示す. また, α は 0 から 1 のいずれかの値をとる.

Sim(dsA, dsB)を算出するために, SimTopic(dsA, dsB)と SimTerms(dsA, dsB)をそれぞれ求める必要がある. SimTopic(dsA, dsB)と SimTerms(dsA, dsB)は扱っているトピックや使用されているメタデータタームがどれほど共通しているかによって変化する. 例として Data.gov で公開されている 2 つのデータセット「Jurisdictions By Election Year[3]」と「Candidate Surplus Funds Reports[4]」は同じ政治に関するトピックを扱っているデータセットであるため, 概要やタグなどで使用される用語の意味には類似性が見られ, SimTopic(dsA, dsB)の値は高くなると考えられる. しかし, 同じ政治分野であっても一方は区域に関するデータセット, もう一方は資金に関するデータセットであるため, メタデータの構造は大きく異なり, SimTerms(dsA, dsB)の値は低くなると考えられる. また, 分野が異なるデータセット同士であってもメタデータスキーマが住所や名前といった分野を問わずに使用されるような項目のみで構成されていた場合 SimTopic(dsA, dsB)の値は低いが SimTerms(dsA, dsB)の値は高くなると考えられる. このように SimTopic(dsA, dsB)と SimTerms(dsA, dsB)のどちらか一方の値が高くてももう一方の値が低い場合はそれらのデータセットを併用することは難しいと考えられる. そのため, トピックの類似性を表す SimTopic(dsA, dsB)と構造の類似性を表す SimTerms(dsA, dsB)の双方の値を見て 2 つの LOD データセットの類似性 Sim(dsA, dsB)を求める必要がある.

SimTopic(dsA, dsB)と SimTerms(dsA, dsB)を算出するために LOD データセットのトピ

[3] https://catalog.data.gov/dataset/draft-jurisdictions-by-election-year

[4] https://catalog.data.gov/dataset/surplus

ックや構造をそれぞれ一つの文書として扱い，それらの類似度を求める．文書の類似度を算出する手法としては，編集距離や単語の一致率などを測る方法が挙げられるが，本研究ではより詳細に単語の意味から類似度を測るために，LOD データセットにおけるトピックや使用されているメタデータタームの特徴を表す分散表現を作成し，それらの類似度を元にして LOD データセットの類似性を算出する．

分散表現とは単語や文書を数百次元のベクトルに変換したものである．意味が類似している単語を類似するベクトルに表現することで，単語間の演算を可能にすることができる．一つの文書に登場する全ての単語の分散表現を加算することで文書の分散表現を算出することができる．単語の分散表現を生成するためのツールとしては「Word2Vec[27]」を用いる．また，特に LOD データセットの構造を表現するために用いられるメタデータタームにおいては同じタームであっても，LOD データセットごとに重要なタームかどうかは異なると考えられる．そのため，本研究では情報検索の分野で用いられることが多い TF-IDF 法を用いて，LOD データセットのトピックや使用タームにおける単語の重みを決定する．Word2Vec で生成した単語の分散表現に TF-IDF 法で算出した単語の重みを掛け，それらを加算したものを LOD データセットのトピックやタームの分散表現とする．

分散表現を生成するためには LOD データセットのトピックや構造を表す単語の集合を作成する必要がある．本研究では，LOD データセットで用いられる単語の中でも LOD データセットのタイトルや概要などの項目で記載されているからなるトピックを表す単語と構造を表す単語それぞれの集合を作成する．

作成の方法として，トピックを表す単語の集合はその LOD データセットが登録されているデータカタログサイトで得られる LOD データセットの名前・概要・タグの 3 つの情報から作成する．その他に LOD データセットの特徴を表すものとしては，LOD データセットの分野や作成者などの項目が考えられるが，分野などの情報はデータカタログサイトごとに付与のルールが異なる．また，作成者や組織などの単語は固有名詞が多く，LOD データセットの特徴とは結びつかないことも多く考えられる．そのため，多くのデータカタログサイトで共通して得ることができるタイトル・概要・タグを情報源として単語の集合を作成する．ただし，これらの情報も文章であるため，「a」や「the」など単独では意味をなさない単語も多く含まれる．そのため，ステミングと形態素解析を行い，動詞・名詞・形容詞のみを抽出したものをそのトピックを表す単語の集合とする．ステミングと形態素解析は Python のライブラリ「Tree-Tagger[5]」を用いて行う．

また，メタデータタームの場合は上述したようにその単語を定義する一意な URI で表さ

[5] https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/

れるため，そのままでは単語としてベクトルに変換することはできない．そのため，そのLOD データセット内の全てのプロパティとクラス及びそれらの使用回数を SPARQL で取得した後，それらのローカル名のみを取得し，それらの集合を作成する．なお，プロパティとクラスはそれぞれ，図 10 と図 11 に示したクエリで取得可能である．図 10 のクエリにおける S は主語，P はプロパティ，O は目的語をそれぞれ表しており，DISTINCT で重複を取り除いた後 COUNT でそれらの数を数えている．また図 11 のクエリにおける a は「http://www.w3.org/1999/02/22-rdf-syntax-ns#type」を省略して表現したものであり，SPARQL 式ではクラスをこのように表現する．

このようにして LOD データセットそれぞれのトピックと構造を表す文書を作成し，それらを元にして LOD データセット A, B における Sim(dsA, dsB)を決定する．

```
SELECT DISTINCT ?p (COUNT(?p) as ?num)
WHERE{
  ?s ?p ?o.
}GROUP BY ?p
```

図 10 : LOD データセット内のすべてのプロパティと使用回数を取得する SPARQL 式

```
SELECT DISTINCT ?class (COUNT(?class) as ?num)
WHERE{
  ?s a ?class.
}GROUP BY ?class
```

図 11 : LOD データセット内のすべてのクラスと使用回数を取得する SPARQL 式

4.3 LOD 間の併用可能性算出手法

併用可能な LOD データセットとしては，構造的に類似するもの以外を挙げるとトピックが共通しているものや参照関係があるものが考えられる．4.1 節で述べた AED を扱う LOD データセットではそれらの間に直接の参照関係はないものの，AED という主として扱っているトピックが共通しているため，そのトピックに関する Web アプリケーション開発に利用することができる．参照関係にあるものでは扱っているトピックに共通性がない場合でも RDF リンクをたどることにより，その主語となっているリソースに関する情報を増やすことができる．そのため，本研究では LOD データセットにおけるトピックの共通性と参照関係に着目し，LOD データセットの併用可能性を算出する．

LOD データセットで扱っているトピックにどれほど類似性があるかを示す指数をトピックの共通度，LOD データセット間の参照関係がどれほど強いかを表す参照関係値とする．このトピックの共通度と参照関係値から LOD データセット間の併用可能性を算出する．2 つの LOD データセット ds1 と ds2 があった場合，それらの併用可能性 ComUse(ds1, ds2)を以下の式で表す．

$$\mathrm{ComUse}(ds1, ds2) = a * \mathrm{ComTopic}(ds1, ds2) + (1-a) * \mathrm{ComLinks}(ds1, ds2) \quad \text{数式 2}$$

ComTopic(ds1, ds2)はデータセット ds1 と ds2 のトピックの共通度を ComLinks(ds1, ds2)は A と B の参照関係値を表す．a は 0 から 1 までの値を取り，ComUse(ds1, ds2)は最小値 0，最大値 1 となる．

ComTopic(ds1, ds2)は 4.2 節で定義した SimTopic(A, B)と同様にトピックの共通性を表すため，概要やタグなどの項目に共通する単語が多く含まれているものは高い傾向になる．一方で，ComLinks(ds1, ds2)は ds1 と ds2 の間における参照関係によって決定し，特に 2 つの LOD データセットの間に直接の参照関係があれば高くなる．例えば，図 1 の LODCloud においてノードの色が赤で示されている生命科学分野の LOD データセットはそれぞれが密接に参照し合っており，双方向に参照している LOD データセットも多く見られる．そのため，このような LOD データセットは ComLinks(ds1, ds2)が高くなると考えられる．一方で，直接の参照関係はなく ds1 から ds2 にたどり着くまでに多くの LOD データセットを経由する必要があるような 2 つの LOD データセットは ComLinks(ds1, ds2)の値も低くなると考えられる．また，DBpedia のように多くの LOD データセットが参照しているような大規模な LOD データセットの場合，ds1 から ds2 への参照関係はあるが逆方向の参照関係はないケースが多くそのような場合も ComLinks(ds1, ds2)はあまり高い数値にはならない傾向にある．

ComTopic(ds1, ds2)を求める手法としてはタグやカテゴリの一致率などを見る方法も挙げられるが，これらの情報はデータカタログサイトや作成者によって付与のルールが異なるほか，一致率だけでは単語の意味も考慮できないため，併用可能性算出手法においても上述の類似性算出手法と同じようにタイトル・概要・タグから分散表現を作成し，それらのコサイン類似度をトピックの共通度とする．なおコサイン類似度は最大値 1，最小値-1 だがこの手法では 0 以上の値を取ったもののみ加算し，コサイン類似度が 0 未満の場合 ComTopic(ds1, ds2)は 0 とする．

次に ComLinks(ds1, ds2)を定義する．Web 上に存在する LOD データセットにおいて全ての LOD データセット間に直接の参照関係があるわけではない．しかし，図 12 のように他の LOD データセットを経由することによって，間接的に参照関係にある場合も考えられる．そのため，2 つの LOD データセットを見たときに直接の参照関係があるか，間接的に参照関係にあるかによって ComLinks(ds1, ds2)の算出手法を分けて考える．

2 つの LOD データセット ds1 と ds2 の間に直接の参照関係があった場合 ComLinks(ds1, ds2)は以下のようになる．

$$ComLinks(ds1, ds2) = Links(ds1, ds2) + Links(ds2, ds1) \quad 数式\ 3$$

Links(ds1, ds2)は ds1 から ds2 への参照関係を評価する関数であり，0~0.5 の値を取る．ここで，2 つのデータセットにおいては参照関係にある RDF リンクの数が多ければ多いほど，併用して得られる情報は多くなると考えられる．しかし，単純に RDF リンクの数を評価すると規模の大きなデータセットは全体的に RDF リンクの数が多いため，併用可能性が高くなる一方で規模の小さいデータセットは全体的に併用可能性が低くなってしまう．そのため，本研究では RDF リンクの数に閾値を設け，RDF リンクの数が閾値以上の場合は強リンクとして Links(ds1, ds2)の値は 0.5, RDF リンクの数が閾値未満だった場合は弱リンクとして Links(ds1, ds2)の値は 0.25 とする．よって，ds1 と ds2 が双方向で強リンクしている場合，それらの間には多くの参照関係があると考えられるため，参照関係値 ComLinks(ds1, ds2)の値は 1 となる．

次に，間接的に参照関係にある LOD データセット ds1, dsN があった場合の参照関係値を定義する．それらの参照関係値 ComLinks(ds1, dsN)は以下のようになる．

$$ComLinks(ds1, dsN) = Com\ Links(ds1, ds2) * ComLinks(ds2, ds3) \cdots * ComLinks(ds(N-1), dsN) \quad 数式\ 4$$

まず, ds1 から dsN にたどり着くまでに最短でいくつの LOD データセットを経由すれば良いかを求める. その後, 数式 3 を用いて経由するそれぞれの LOD データセット間における参照関係値を求める. それら全ての参照関係値をかけたものを ds1 と dsN の参照関係値とする. 例として, 図 10 の場合は ds1 から ds2 へたどり着くまでに ds3 のみを経由すれば良いので ComLinks(ds1, ds2)は

$$\mathrm{ComLinks}(ds1, ds2) = \mathrm{ComLinks}(ds1, ds3) * \mathrm{ComLinks}(ds3, ds2) \quad \text{数式 5}$$

となる.

このようにして求めたトピックの共通性を表す ComTopic(ds1, ds2)と参照関係値 ComLinks(ds1, ds2)にそれぞれ重みづけを行い, 合算したものを 2 つの LOD データセット間における併用可能性 ComUse(ds1, ds2)として定義する.

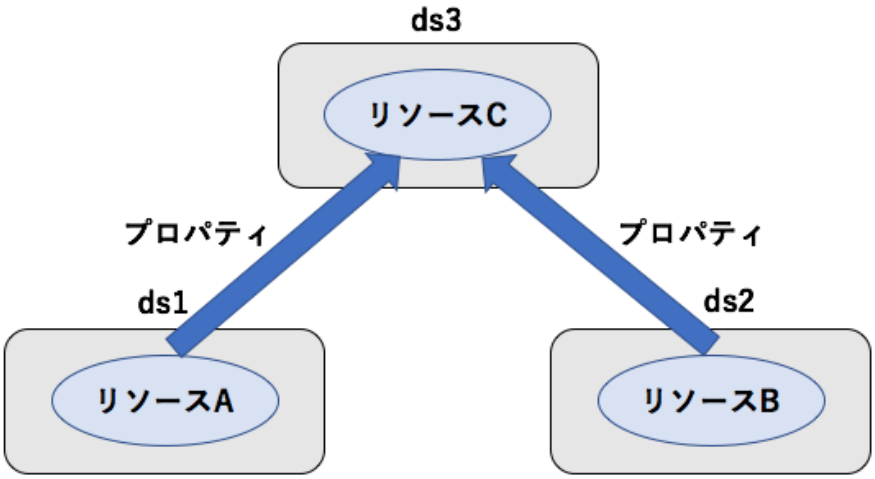

図 12 : 間接的にリンクしている LOD データセット

5. LOD 間の類似性算出手法の評価実験

5.1 評価手法

類似性算出手法の評価実験は実際に複数の LOD データセットの分散表現を生成した後，生成した分散表現同士のすべての組み合わせで類似度を算出し，それらの分布と平均値を算出する．また，生成した分散表現すべてでの平均値のほか，あらかじめ類似度が高いと予想される LOD データセットの組の集合 A と類似度が低いと予想される LOD データセットの組の集合 B を用意し，それらの類似度の平均値を算出する．また，類似度が高いと予想される組と類似度が低いと予想される組の割合が 1:1 で構成される LOD データセットの組の集合 C を用意する．そしてそれぞれの集合の平均値 A, B, C が

$$\text{平均値 A} > \text{平均値 C} > \text{平均値 B}$$

となった場合，この手法で算出した LOD データセット間の類似性は妥当性があると考えられる．また，データセットの組全体での平均値が平均値 C に最も近づけば，全体での類似度はより妥当性があると言える．

その後，分散表現を生成する式のパラメータを変更し，どのパラメータが類似度に与える影響が大きいかを考察する．

5.2 実験

実験は以下の手順で行う．

1. 実験用データセットの取得
   Data.gov で RDF 形式のファイルを公開しているデータセットのうち最新 1,000 件を取得する．そのうち，文法的に問題のなかった 814 件のデータセットを実験に使用した．
2. 取得したデータセットから分散表現を生成
   取得した 814 件のデータセットそれぞれのタイトル・概要・タグから抽出した単語の集合と，メタデータタームのローカル名を抽出した単語の集合を作成し，それぞれの分散表現を生成した．
3. 生成した分散表現それぞれにおけるコサイン類似度を算出
   生成した分散表現を用いて，全ての LOD データセットの組み合わせにおける SimTopic(dsA, dsB)と SimTerms(dsA, dsB)を算出する．

4. SimTopic(dsA, dsB)と SimTerms(dsA, dsB)の重みを設定
   SimTopic(dsA, dsB)と SimTerms(dsA, dsB)それぞれの平均と分散を算出し，その結果を元に 2 つの値の重みづけを行う．
5. Sim(dsA, dsB)を算出

最初に手順 1 で実験に用いる LOD データセットの選別と取得を行う．実験に使う LOD データセットの集合は類似性が高いと思われる組と類似性が低いと思われる組を明確な基準でグループ分けを行えることが望ましい．そのため，同一のデータカタログサイトにアップロードされている LOD データセットを対象とする．また，この手法では対象とする LOD データセットで使用されているメタデータタームのすべてを算出する必要があるため，何らかの方法で SPARQL 式での問い合わせを行う必要がある．よって，SPARQL エンドポイントがあらかじめ提供されているもの，もしくはダンプファイルをダウンロードできるものである必要がある．また，DBpedia のように大規模なデータセットであった場合，使用されているメタデータタームすべてを取得するようなクエリを実行した場合，問い合わせがタイムアウトを起こしてしまい，必要なメタデータタームが取得できない可能性がある．

これらの理由から本研究では実験用データセットとして Data.gov にアップロードされているものを使用した．Data.gov には 2019 年 12 月時点で約 26 万件のデータセットが登録されており，そのうち約 12,000 件が RDF 形式のファイルをダウンロードすることができる．これらのファイルはそれほど容量が大きくないものが多いため，SPARQL 式での問い合わせがタイムアウトを起こす可能性は低い．また，概要やタグといった情報に加え，Data.gov のサイト側で設定されたテーマが与えられているため，このテーマが同じものであれば類似度が高いもの，テーマが異なるものであれば類似度が低いものとして明確な基準でグループ分けを行うことができる．

実験用データセットを取得する手順として，まず各 LOD データセットに割り当てられている ID を取得する必要がある．そのため，Data.gov のデータセット検索システムでファイル形式が RDF となっているものを絞り込む．その後 Ruby ライブラリ「Nokogiri[6]」を用いて Web スクレイピングを行い，検索結果の画面から各データセットの詳細情報画面の URL を取得する．取得した URL の末尾が各データセットの ID である．

ID を取得した後，各データセットのタイトル・概要・タグ・RDF ファイルのダウンロード URL・テーマを取得する．Data.gov では図 13 のように各データセットの情報を JSON 形式で取得できる API[26]を提供しており，各データセットの ID をキーとして与えることで

[6] https://nokogiri.org

その詳細情報を取得することができる．そして取得したRDFファイルのダウンロードURLからファイルのダウンロードが成功したものをローカル環境で立ち上げたSPARQLエンドポイントにアップロードし，文法などのエラーが起こらず正常にアップロードできたものを実験用データセットとして使用する．

実験用データセットとして，RDF 形式のファイルがアップロードされておりメタデータの更新日時が新しい順に 1,000 件のデータセットをダウンロードした．それらのデータセットのうち，正常にSPARQLエンドポイントにアップロードできたデータセットは814件であった．よってこれらのデータセットを実験用データセットとして実験を行なった．

手順2では取得したLODデータセットそれぞれの分散表現の生成を行う．分散表現の生成をするにあたり，Word2Vecで単語のベクトル化を行うために単語の意味を学習したモデルが必要となる．学習用のデータとしては英語版の Wikipedia ダンプをダウンロードし，300次元反復回数5で学習を行った．このモデルを用いてデータセットのトピックを表す単語の集合と各メタデータタームのローカル名を抽出した単語の集合それぞれのベクトル化を行い，トピックを表す分散表現と構造を表す分散表現をそれぞれ生成する．

手順3では生成した2つの分散表現を用いてSimTopic(dsA, dsB)とSimTerms(dsA, dsB)を求める．実験用データセットから 2 つを選んだ場合のすべての組み合わせにおいてトピックを表す分散表現と構造を表す分散表現それぞれのコサイン類似度を算出する．これらの値がSimTopic(dsA, dsB)とSimTerms(dsA, dsB)となる．

手順 4 では SimTopic(dsA, dsB)と SimTerms(dsA, dsB)の重みを設定する．算出したSimTopic(dsA, dsB)とSimTerms(dsA, dsB)の平均と分散を算出し，どちらの値がより

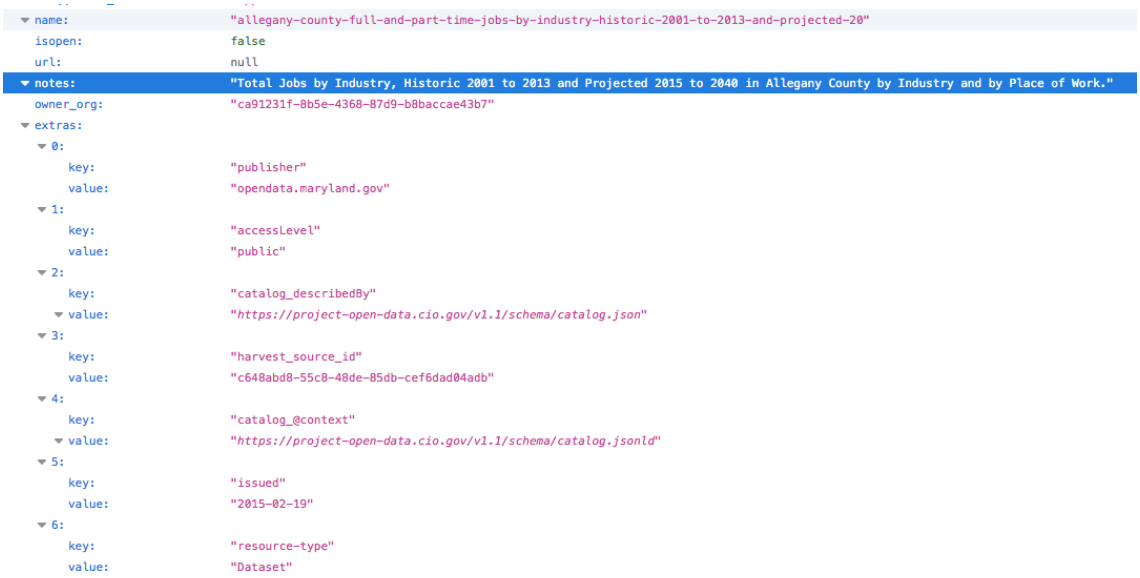

図 13 : Data.gov の各データセット情報を提供する API

表 3 : SimTopic(dsA, dsB)と SimTerms(dsA, dsB)の平均と分散

	平均	分散
SimTopic(dsA, dsB)	0.367	0.0242
SimTerms(dsA,dsB)	0.321	0.0342

Sim(dsA, dsB)に与える影響が大きいかを確認する．その結果を元に 2 つの値の重みづけを行う．

SimTopic(dsA, dsB)と SimTerms(dsA, dsB)それぞれの平均値は表 3 のようになった．SimTopic(dsA, dsB)と SimTerms(dsA, dsB)それぞれの平均と分散には大きな差は見られないことがわかる．そのため，2 つの値が Sim(dsA, dsB)に与える影響は同程度と考え，それぞれの重みを 0.5 に設定する．

最後に手順 5 で重みづけした SimTopic(dsA, dsB)と SimTerms(dsA, dsB)を合算して，Sim(dsA, dsB)を求める．

5.3 実験結果

取得した 814 件の LOD データセットでのすべての組み合わせにおける類似性の分布は図 14 のようになった．横軸は各 LOD データセット間の類似性，縦軸は組み合わせの数を示す．この分布は中央値に近づくにつれ数が多くなる正規分布に近いが，類似度が「0.7~0.8」の組み合わせの数と「0.8~0.9」の組み合わせの数と比べ，「0.9~1.0」の組み合わせの数の

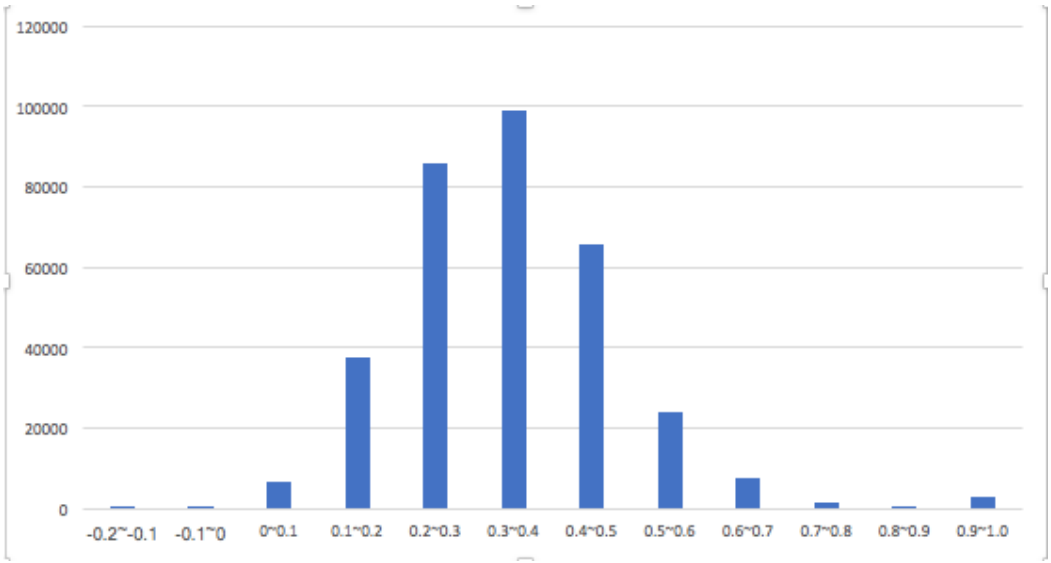

図 14 : 全データセットの組の類似性の分布

表 4 : LOD データセット全体と集合 A・B・C の類似性の平均値の比較

全体の平均値	集合 A の平均値	集合 B の平均値	集合 C の平均値
0.3434	0.5630	0.3276	0.4343

表 5 : パラメータを変えた状態での LOD データセット間の類似度の平均値の比較

通常の手法	パターン 1	パターン 2	パターン 3
0.3440	0.3544	0.3001	0.3594

方がやや多いことがわかる.

また, コサイン類似度は最大値 1, 最小値-1 を取るが, 実験用の LOD データセットの組み合わせでは類似度が-0.2 以下となった組み合わせは 1 組も見られなかった.

次に LOD データセットの類似性の平均値を算出する. 実験用のデータセット 814 件から LOD データセット 2 件を選んだ組み合わせの数は 330,891 組. その中から類似性が高いと予想される組み合わせと類似性が低いと予想される組み合わせを 1,000 組ずつサンプリングし, 類似度が高い LOD データセットの組の集合 A と類似度が低い LOD データセットの組み合わせの集合 B を作成し, それらの平均値を算出した. また, 類似性が高いと予想される組み合わせと類似性が低いと予想される組み合わせを 500 件ずつ取得して一つの集合 C とし, その平均値を算出した. これらの LOD データセットの組み合わせの類似性の平均値は表 4 のようになった. 結果として集合 A・B・C の平均値は「平均値 A > 平均値 C > 平均値 B」となったが, 一方で全体での平均値は集合 B の平均値に最も近いという結果となった.

次に LOD データセットの分散表現を生成するにあたり, どのパラメータが最も影響を与えるかを検証する. 収集した LOD データセット 814 件のうち 100 件をランダムに選び, それらの分散表現を通常の手法の分散表現とは別に以下の 3 パターンの分散表現を生成した.

パターン 1 : タイトルで出現する単語のベクトルを分散表現に加算しない
パターン 2 : 概要で出現する単語のベクトルを分散表現に加算しない
パターン 3 : タグで出現する単語のベクトルを分散表現に加算しない

これら 3 つのパターンと通常の手法の分散表現における類似性の平均値を比較すると表 5 のようになった. これを見ると概要に出現する単語を使用しないパターン 2 の手法が最も変化が大きい. また, パターン 1 やパターン 3 の場合は平均値が増加しているのに比べ, パ

ターン 2 は平均値が減少していることがわかる.この結果から概要には意味が類似する単語が多く含まれると考えられる.

6. LOD 間の併用可能性算出手法の評価実験

6.1 評価手法

併用可能性算出手法の評価は，Web 上に公開されている既存の LOD データセット複数の情報を取得し，それらの LOD データセット間における併用可能性を実際に算出することで行う．算出した LOD データセット間全体における併用可能性の分布と平均値と同一分野の LOD データセット間における併用可能性の平均値，異なる分野の LOD データセット間における平均値を比較する．

また，LOD データセット全体の組とは別に間接的にリンクしている 3 つの LOD データセットに着目して，併用可能性の平均値を算出する．3 つの LOD データセット 1・2・3 があった際，それらの関係性は以下の 3 つが予想される．なお，ここでは特定のデータセット 1 内のリソースが他のデータセット 2 のリソースを参照している場合，データセット 1 はデータセット 2 を参照していると表現する．

ケース 1：LOD データセット 1 が LOD データセット 3 を参照し，LOD データセット 3 が LOD データセット 2 を参照している（図 15）

ケース 2：LOD データセット 1 と LOD データセット 2 が共通して LOD データセット 3 を参照している（図 16）

ケース 3：単一の LOD データセット 3 が LOD データセット 1 と LOD データセット 2 をそれぞれ参照している（図 17）

これらの 3 つのケースにおけるデータセット 1 と 2 の併用可能性の平均値を比較して評価を行う．この 3 つのケースにおいて特にケース 2 の場合，1 と 2 は同じデータセットから情報を得ているため，扱っているトピックにもある程度共通性があり，他の 2 つのケースより併用可能性が高くなると考えられる．一方で，単一のデータセットが異なる複数のデータセットを参照しているケース 3 の場合，それらは異なるトピックを扱っており，併用可能性は他の 2 つのケースより低くなると予想される．

その後，併用可能性を算出する際のパラメータを変更し，どのパラメータが併用可能性に

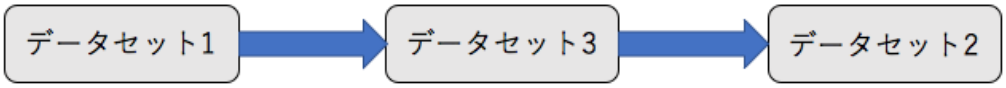

図 15：間接的に LOD データセットがリンクしているケース 1

与える影響が大きいかを考察する.

6.2 実験

実験は以下の手順で行う.

1. 実験用データセットの取得
   LODCloud に登録されているデータセットのうち参照関係が確認できた 1,232 件のデータセットのタイトル・概要・タグと参照関係を取得
2. 取得したデータセットから ComTopic(ds1, ds2)の算出
   取得した 1,232 件のデータセットそれぞれのタイトル・概要・タグから抽出した単語の集合を作成し, その分散表現を生成する. その後, 生成した分散表現のコサイン類似度を算出する.

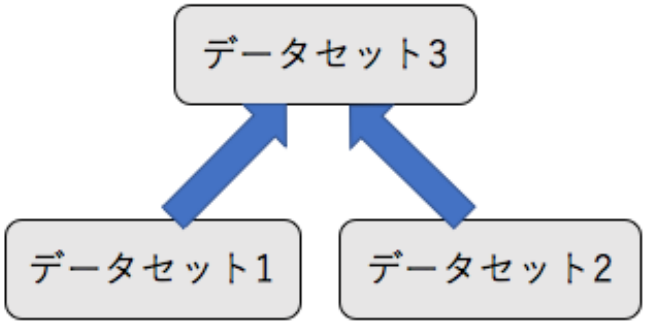

図 16 : 間接的に LOD データセットがリンクしているケース 2

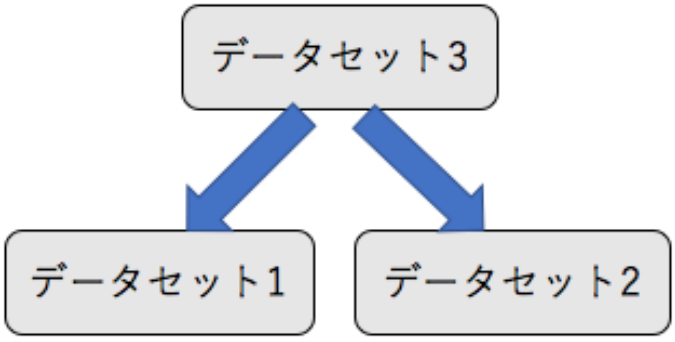

図 17 : 間接的に LOD データセットがリンクしているケース 3

3. 実験用データセット間における最短経路の算出
   グラフにおける 2 つのノードの最短経路を求めるワーシャル・フロイド法を用いてデータセット間における参照関係の最短経路を算出する．
4. 算出した最短経路を用いて ComLinks(ds1, ds2)を算出
   数式 3, 4 を用いて実験用データセットにおけるすべての組み合わせの ComLinks(ds1, ds2)を算出
5. ComTopic(ds1, ds2)と ComLinks(ds1, ds2)の重みを設定
   手順 2 と手順 4 で算出した ComTopic(ds1, ds2)と ComLinks(ds1, ds2)の平均と分散を算出し，その結果を元にそれぞれの重みを設定する．
6. ComUse(ds1, ds2)を算出

まず手順 1 で実験用データセットを取得する．この実験では各 LOD データセットのトピックを表すタイトル・概要・タグのほか，特定の LOD データセットが他のどの LOD データセットを参照しているかが明確になっていることが望ましい．そのため，実験用 LOD データセットには「LODCloud」に登録されているものを使用する．これらの LOD データセットは参照関係が可視化されているほか，LOD データセットの各種情報や参照関係を一つの JSON ファイルにまとめて提供されている．そのため，これらの LOD データセットの情報を取得し，実際に併用可能性を算出する．

LOD データセットの情報をまとめた JSON ファイルは図 18 のようになっている．最も上の階層のキーがそれぞれの LOD データセットの ID となっており，下の階層に各 LOD データセットの詳細情報が格納されている．そのため，それぞれの詳細情報にアクセスするためには各 LOD データセットに割り当てられた ID をあらかじめ取得する必要がある．

図 18 : LODCloud の情報をまとめた JSON ファイル

各 LOD データセットの ID は LODCloud の参照関係を可視化している SVG ファイルから Web スクレイピングを用いて取得する．取得には Ruby ライブラリ「Nokogiri」を用いた．ID の取得後はそれらの ID を使って，図 16 の JSON ファイルから各 LOD データセットの詳細情報を取得する．

手順 2 では取得したデータセットの ComTopic(ds1, ds2)を算出する．図 16 の JSON ファイルから各 LOD データセットのタイトル・概要・タグを取得し，それらから特徴的な単語の集合を作成する．作成した単語の集合をベクトル化し，各データセットのトピックを表す分散表現を生成する．その後，生成したすべての分散表現の間におけるコサイン類似度を算出し，その値を 2 つの LOD データセット ds1, ds2 のトピックの共通度 ComTopic(ds1, ds2)とする．

手順 3 では各 LOD データセットの参照関係値を算出するための準備を行う．各 LOD データセットの参照関係を示す構造は図 19 のようになっている．このデータは参照元となる LOD データセットの詳細情報を格納している構造の中にあり，「target」は参照先となる LOD データセットの ID，「value」はその LOD データセット間の RDF リンクの数を表している．これにより，直接リンクしている LOD データセット間の参照関係を取得することができる．しかし，この情報のみでは間接的にリンクしている LOD データセット間の参照関係を取得することはできない．間接的に LOD データセット 1 と 2 において，参照関係を辿って 1 から 2 にたどり着くための経路は一つとは限らない．しかし，経由する LOD データセットが多くなるとそれだけ 1 と 2 を併用するために仲介しなければならない LOD データセットの数も多くなり，使いにくくなると考えられる．そのため，LOD データセット 1 から LOD データセット 2 までの最短経路を求める．

LODCloud には明確な始点と終点が存在しないほか，すべての LOD データセット間における最短経路を求めることが望ましい．そのため，各 LOD データセット間の最短経路を求める手法としてワーシャル・フロイド法を用いる．ワーシャル・フロイド法はグラフにおける全てのノード間の最短経路を求めるための手法であり，アルゴリズムは図 20 のように

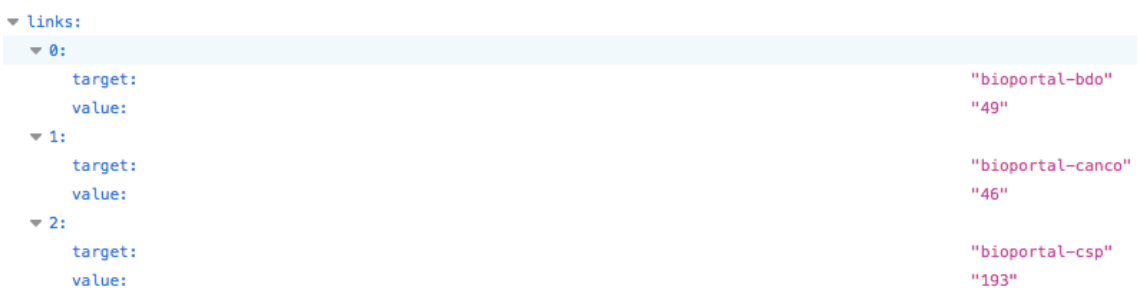

図 19：各 LOD データセットの参照関係

なる. 図 20 における num はグラフにおけるノードの数, cost[i][j]はノード i からノード j までの経路のコスト, k は経由するノードの ID, i は始点となるノードの ID, j は終点となるノードの ID をそれぞれ表す. 事前の準備としてノード i からノード j に向かってパスが伸びていた場合, cost[i][j]にそのパスのコストを格納しておく. さらに直接パスが伸びていないノード間においては到達不能な値 INF を格納しておく. そして, 始点 i から終点 j までの経路においてノード k を中継した方が現在のコストよりも低い場合はコストを書き換える. この手法を LODCloud のグラフに適用し, 各 LOD データセット間の最短経路を算出する.

事前準備として各 LOD データセットの参照関係を取得し, LOD データセット 1 が LOD データセット 2 を参照していた場合, それらの番号をそれぞれ m, n とすると cost[m][n]の値を 1 にする. 同時に cost[n][m]の値も 1 にする. これは LOD データセットを併用するという観点で見た場合, どちらが参照元であっても, 併用自体はできると考えられるためである. そのため, cost[m][n]と cost[n][m]の値は等しくなる.

手順 4 では手順 3 で算出した LOD データセット間の最短経路を元にして 2 つの LOD データセット ds1 と ds2 の間における参照関係値 ComLinks(ds1, ds2)の値を決定する. この値の決定には図 19 の value を使用する. LODCloud に LOD データセットを登録するための条件はすでに登録されている LOD データセットの少なくとも一つに対し 50 以上の RDF リンクがあることが条件とされている[10]. 実際に LODCloud 内の LOD データセット間における参照関係から value を確認したところ, 直接の参照関係が確認できた LOD データセットの組み合わせ 17,632 組のうち value が 50 を超えたのは 8,121 組であった. およそ 46%の割合であり半分以上の組み合わせが RDF リンク数 50 を下回っている. 強リンクと弱リンクを設定する尺度としては半分ずつに分けられることが理想的であり, この割合は比較的 50%に近いため尺度としては妥当性があると言える. そのため, LOD データセット 1 から LOD データセット 2 の参照関係を確認し, value が 50 以上であれば Links(ds1, ds2)の値は 0.5, value が 50 未満であれば Links(ds1, ds2)の値は 0.25, 参照関係がなければ

```
for k in 0..num-1
    for i in 0..num-1
        for j in 0..num-1
            if cost[i][j] > cost[i][k] + cost[k][j]
                cost[i][j] = cost[i][k] + cost[k][j]
            end
        end
    end
end
```

図 20：ワーシャル・フロイド法のアルゴリズム

Links(ds1, ds2)の値は 0 と設定し Links(ds1, ds2)と Links(ds2, ds1)の和を ComLinks(ds1, ds2)の値として二次元配列 link[m][n]に格納する．そして図 20 のアルゴリズムを適用し，cost[i][j]が更新された場合

$$link[i][j] = link[i][k] * link[k][j] \quad 数式\ 6$$

を実行する．これにより，間接的にリンクしている LOD データセット A と B の間の最短経路と ComLinks(ds1, ds2)の値を求めることができる．

手順 5 では ComTopic(ds1, ds2)と ComLinks(ds1, ds2)の重みを設定する．それぞれの値の平均と分散を算出し，どちらの値が ComUse(ds1, ds2)に与える影響が大きいかを確認する．その結果を元に ComTopic(ds1, ds2)と ComLinks(ds1, ds2)それぞれの重みを決定する．

ComTopic(ds1, ds2)と ComLinks(ds1, ds2)それぞれの平均と分散は表 6 のようになった．これを見ると平均・分散共に ComTopic(ds1, ds2)の方が大きく上回っている．そのため，ComUse(ds1, ds2)を考慮した時，ComTopic(ds1, ds2)の数値に大きく影響を受けると考えられる．よって，この実験では ComTopic(ds1, ds2)の数値を重視し，こちらの重みを 0.6, ComLinks(ds1, ds2)の重みを 0.4 に設定する．

最後に手順 6 では重みづけした ComTopic(ds1, ds2)と ComLinks(ds1, ds2)を合算して，ComUse(ds1, ds2)を算出する．

6.3 実験結果

LODCloud に登録されている 1232 件の LOD データセットでのすべての組み合わせにおける併用可能性を求めると分布は図 21 のようになった． 数値が 0.7 以上になっているもののの数が極端に少ないのが特徴的である．また，正規分布とは大きく異なり，中央値にあたる「0.4~0.5」の数もその前後と比べて少ないことがわかる．

また，LOD データセット全ての組み合わせでの ComUse(ds1, ds2)の平均値・同一分野の LOD データセット間の ComUse(ds1, ds2)の平均値・異なる分野の LOD データセット間の ComUse(ds1, ds2)の平均値とそれらの組み合わせの数は表 7 のようになった．同一分野における平均値は異なる分野における平均値と比較して 0.17 ほど大きいが，組み合わせ数は異なる分野のものが全体の約 9 割を占めているため，全体の平均値は異なる分野のものに近い結果となった．

次に間接的にリンクしている LOD データセットのパターンごとの平均値を算出する．パターン I, II, III に従う LOD データセット 1 と 2 のリストを作り，そのリストからランダム

表 6 : ComLinks(ds1, ds2)と ComTopic(ds1, ds2)の平均と分散

	平均	分散
ComLinks(ds1, ds2)	0.0887	0.0173
ComTopic(ds1, ds2)	0.6517	0.1254

表 7 : 分野による併用可能性の平均値の比較

	全体	同一分野	異なる分野
平均値	0.4265	0.5743	0.4051
組み合わせ数	758,296	95,934	662,362

に 5,000 組ずつサンプリングする．そしてサンプリングした 5,000 組の LOD データセット間の併用可能性の平均値は表 8 のようになった．予想とは異なりケース 2 に当てはまる LOD データセットの組が最も併用可能性の平均値が低いという結果となった．一方で，ケース 1 とケース 3 の LOD データセットの組は併用可能性の平均値に大きな差は見られなかった．

最後にパラメータを変更しての併用可能性を比較する．通常の併用可能性とは別に取得した LOD データセット 1,232 件において以下の 3 パターンの条件下で併用可能性を算出し，それぞれの平均値を比較したものを表 9 に示す．

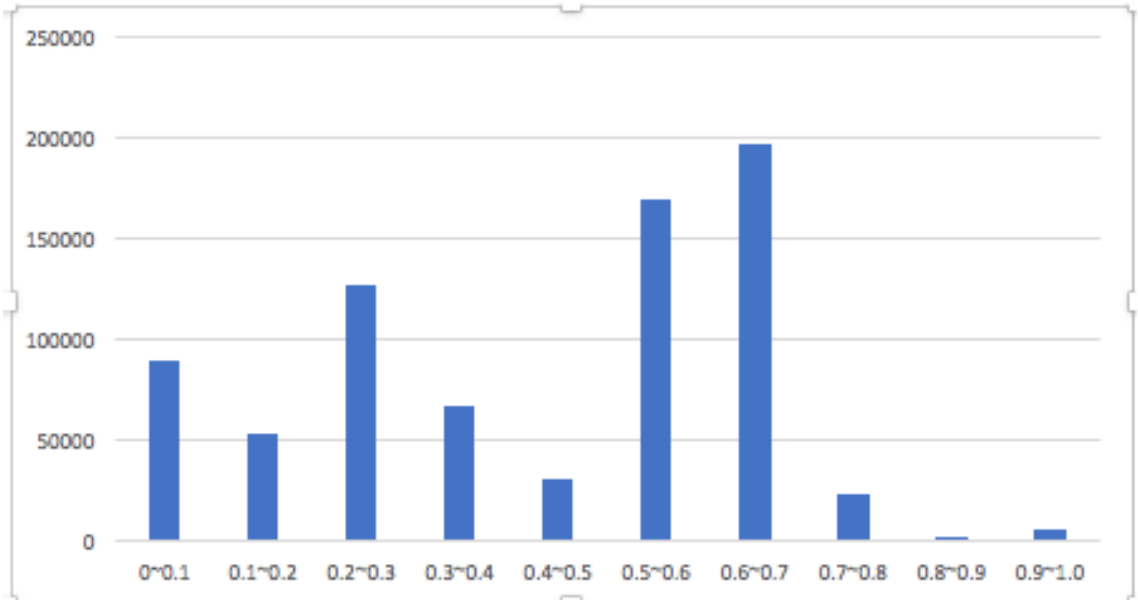

図 21 : LOD データセットの併用可能性の分布

表 8：間接的にリンクしている LOD データセット間の併用可能性平均値比較

ケース 1	ケース 2	ケース 3
0.5933	0.5242	0.6050

表 9：パラメータを変えた状態での LOD データセット間の併用可能性の平均値の比較

通常の手法	パターン 1	パターン 2	パターン 3
0.4265	0.4270	0.4414	0.3025

パターン 1：タイトルで出現する単語のベクトルを分散表現に加算しない

パターン 2：概要で出現する単語のベクトルを分散表現に加算しない

パターン 3：タグで出現する単語のベクトルを分散表現に加算しない

パターン 1 とパターン 2 は通常の手法と大きく変化がない一方で，パターン 3 は通常の手法と比べ，平均値が 0.1 ほど減少しているのが特徴的である．

7. 考察

7.1 LOD 間の類似性に関する考察

5 章で行った実験により，本研究で算出した類似性では類似度が高いと想定した集合 A，類似度が低いと想定した集合 B，類似度が高い組み合わせと類似度が低い組み合わせを同数用意した集合 C で「平均値 A＞ 平均値 C＞ 平均値 B」となったため，おおよそ予想通りに類似度を算出することができたと言える．一方で，全体の類似度はこちらの予想とは異なり平均値 B に最も近い結果となった．これは，類似度が高いか低いかを決定する基準をテーマが同じかそうでないかに設定したことが原因であると考えられる．今回の実験では 814 件のデータセットの分散表現を生成したが，この 814 件のデータセットから 2 件を選んでできる組み合わせ 330,891 件のうちテーマが同じ組み合わせは 20,603 件であり 10% にも満たない．そのため，類似度が高いか低いかを決定するための基準はまた別のものを設定して再び実験を行う必要がある．

分布においては正規分布に近いが，類似度が 0.7~0.8 の組み合わせと 0.8~0.9 の組み合わせより 0.9~1.0 の組み合わせの方が多く見られることが特徴である．これは「Charles County Full and Part Time Jobs By Industry[7]」と「Frederick County Full and Part Time Jobs By Industry[8]」のように固有名詞を除けばタイトル・概要・タグで使用している単語が全て同じという組み合わせが多数存在することが原因であると考えられる．このようなデータセットは製作者が同じであるため，タイトルや概要，タグに加え，実際のデータセットのファイルを確認すると使用されているメタデータタームが完全に一致するという特徴が挙げられる．そのため，固有名詞が正しくベクトル化できなかった場合，類似度は限りなく 1 に近づいてしまう．そのため，固有名詞が出現した場合も想定し，ベクトルでの分散表現とは別に単純な単語の一致率も考慮する必要がある．ただし，構造が全く同じデータセットであれば，Web アプリケーションでの併用は容易なため，本研究での目的を考慮すればこの結果は妥当であるとも言える．

パラメータにおいてはタイトル・概要・タグのいずれも出現する単語のベクトルを加算しなかった場合，タイトルやタグの場合は若干ではあるが平均値が増加したのに対し，概要の場合は平均値が減少し，差も他の 2 つより大きいことが特徴的である．これは他の項目と

[7] https://catalog.data.gov/dataset/charles-county-full-and-part-time-jobs-by-industry-historic-2001-to-2013-and-projected-201

[8] https://catalog.data.gov/dataset/frederick-county-full-and-part-time-jobs-by-industry-historic-2001-to-2013-and-projected-2

比較して，概要には類似する単語が多く含まれることを意味する．そのため，概要における単語の使用傾向を分析し，多くのデータセットで共通して使用されるような単語を取り除くことによってより類似性の精度を高めることができるのではないかと考えられる．

また，本研究で提案した手法で対象となるLODデータセットで使用されているメタデータタームすべての情報を必要とするが，大規模なデータセットではその取得が難しいほか，分散表現の生成にもかなりの長時間を要するという問題がある．そのため，すべてのタームを利用するのではなくある程度 LOD データセットの中身をサンプリングするなどの改善が必要となる．

7.2 LOD 間の併用可能性に関する考察

6章で行った実験により，本研究で算出したLODデータセット間の併用可能性の分布は併用可能性が0.7以上になるものが極端に少ないという結果になった．これは，2つのLODデータセット間における参照関係値が全体的に低いことが原因であると考えられる．LODCloudに登録されているLODデータセットは1,232件あるが，それらが他のLODデータセットを参照していることを示すエッジはおよそ16,000件程度しか存在しない．LODデータセット 1,232 件から 2 件のデータセットを選んだ場合，その組み合わせの数はおよそ 76 万組あることを考慮するとエッジの数が非常に少ない．そのため，多くの LOD データセットが到達不能になっているものだと考えられる．また，多くのデータセットがDBpedia などの著名なデータセットを参照しているため，参照先になっていないデータセットも多数見られる．このことからデータセット間のリンクを辿った先の終点となるデータセットに偏りが生まれ，参照先となっていないデータセットの場合，他のデータセットとの併用可能性が低くなってしまうと考えられる．一方で，生命科学分野のようにその分野内のデータセット同士で密接に参照し合っている場合は，参照関係値が平均的に高くなり併用可能性の値も大きくなる傾向にある．

次に間接的にリンクしている LOD データセットのパターンごとによる併用可能性の違いについて述べる．予想に反して，2つのLODデータセットが同一のLODデータセットを参照しているパターンが最も併用可能性の平均値が低いという結果となった．これは 2 件以上の LOD データセットから参照されている LOD データセットの数が半分にも満たず，さらにDBpediaをはじめとしてFreebase[9]やVIAF[10]など著名なLODデータセットに多くのリンクが集まっていることが原因として挙げられる．そのような著名なLODデータセッ

[9] https://developers.google.com/freebase/

[10] https://viaf.org

トは他の LOD データセットから参照される数と比べ，自らが参照している LOD データセットの数は少ない傾向にある．そのため，同一の LOD データセットにリンクしていた場合でもその間の参照関係性は低くなるほか，多くの LOD データセットが共通して参照するもののため，それらのトピックの共通度も平均して低くなると考えられる．

最後に併用可能性算出のために用いたパラメータについて考察する．他のパラメータと比較して，タグを用いずに併用可能性を算出した場合，平均値には大きな変化が見られた．そのため，LODCloud に登録されている LOD データセットに付与されているタグは Data.gov に登録されている LOD データセットに付与されているものと比較して，特徴が現れやすいものと考えられる．しかし，多くの LOD データセットに共通して付与されているタグも存在すると考えられるので使用傾向を分析し，ストップワードを設定した上で再び実験を行う必要がある．

7.3 研究全体における考察

本研究では Web アプリケーション開発などの際に組み合わせて使用できる LOD データセットの発見を目的として，類似性や参照関係に着目した手法を提案した．提案した 2 つの手法において異なる 2 つのデータセットにおけるトピックの類似性は同一の手法で算出している．しかし，5 章で行った類似性算出実験におけるトピックの類似性 SimTopic(dsA, dsB)と 6 章で行った併用可能性算出実験におけるトピックの類似性 ComTopic(ds1, ds2)の平均値は大きく異なる結果となった．そのため，データカタログサイトごとに名前や概要，タグなどで使用される単語の傾向を分析し，より正確にトピックの類似性を算出する手法を検討する必要がある．

また，本研究では併用可能なデータセットとして類似性や参照関係のあるものを挙げて，それらを数値化する手法を提案した．しかし，この手法で高い数値を記録した 2 つのデータセットが実際に Web アプリケーションの開発の際に組み合わせて使用できるかの定性的な評価は不十分である．そのため，これらの手法で高い数値を記録した組と低い数値を記録した組を比較し，どちらの方が組み合わせて使用しやすいかを比較するなどしてこの手法の妥当性を検証する必要がある．

8. おわりに

本論文では, LOD データセットを用いて Web アプリケーションを開発する際に，併用可能な他の LOD データセットの情報を提示することが有用であると考え，併用可能なものの中でも特に類似している LOD データセット，さらに参照関係に着目して，併用可能な LOD データセットを発見するための LOD データセット間の類似性と併用可能性の算出手法を提案した.

類似性算出手法では LOD データセットの構造に基づき，使用されているメタデータタームの傾向が近ければ，それらの LOD データセットの類似性は高くなると仮定して, LOD データセットの分散表現を作成した．それらの類似性を実際に算出したところ，事前に類似度が高いと思われるものと低いと思われるものの数値はおおよそ予想した通りの結果となった．しかし，全体的な LOD データセットにおける類似性は低く，またパラメータを変更したところ，概要で出現する単語を取り除いた場合，他の項目と比べ，類似度の平均が大きく下がった．そのため，概要には全体的に意味が類似する単語が多く含まれていると考えられるため，単語の使用傾向の分析が必要となる.

併用可能性の算出手法では LOD データセット間の参照関係に着目して実験を行なった．ワーシャル・フロイド法を用いて LOD データセット間の最短経路を算出，そこから LOD データセットの参照関係の度合いを表す参照関係値を算出して，併用可能性を決定した．結果としては予想以上に到達可能な LOD データセットの組み合わせが少なく，併用可能性は全体的に低い数値となった．また，他のパラメータと比べ, LODCloud に登録されている LOD データセットに付与されているタグはそのデータセットの特徴が現れやすいという知見を得た．しかし，この手法ではあくまでも他の LOD データセットを参照しているという事実に基づいているだけなので，具体的にどのようなプロパティを用いて参照しているかといった観点での分析が必要となる.

また，これらの手法は一つの LOD データセット内で使用されている全てのメタデータタームや参照関係を用いる必要がある．しかし，大規模な LOD データセットではそういった情報を全て取得するのは難しいほか，類似性・併用可能性の算出にかなりの時間を要するという問題がある．そのため, LOD データセットの中身の一部をサンプリングするなどして手法を効率化する必要がある．また，これらの手法で高い数値を記録したデータセットの組が実際に組み合わせて使用できるかの分析はまだ不十分であるため，今後の課題として挙げられる.

謝辞

これまでの研究を進めるにあたり，テーマの決定やゼミでの議論において様々な場面でご指導やご意見をいただいた永森光晴先生，森嶋厚行先生，三原鉄也先生，阪口哲男先生，杉本重雄先生に感謝申し上げます．併せて本研究に多くの助言をくださったメタデータ研究室の皆様に感謝いたします．ここに心より感謝の意を表します．

参考文献

[1] Data.gov. https://www.data.gov, (accessed 2020-01-08)

[2] Data.go.jp. https://www.data.go.jp, (accessed 2020-01-08)

[3] “Open Gov Data Hack 2019 Terms and Comditions”. Event Data.Gov.in. https://event.data.gov.in/open-gov-data-hack-2019-terms-and-conditions/, (accessed 2020-01-08).

[4] ジャパンサーチ(BETA). https://jpsearch.go.jp, (accessed 2020-01-08)

[5] 東京大学学術資産等アーカイブズポータル. https://da.dl.itc.u-tokyo.ac.jp/portal/, (accessed 2020-01-08)

[6] Tim Berners-Lee. Linked Data – Design Issues. https://www.w3.org/DesignIssues/LinkedData.html, (accessed 2020-01-08)

[7] DBpedia. https://wiki.dbpedia.org, (accessed 2020-01-08)

[8] Europeana. https://www.europeana.eu/portal/, (accessed 2020-01-08)

[9] “オープンデータ憲章(概要)”. 外務省. https://www.mofa.go.jp/mofaj/files/000006820.pdf, (accessed 2020-01-08)

[10] The Linked Open Data Cloud. https://lod-cloud.net, (accessed 2020-01-08)

[11] RDF – Semantic Web Standards. https://www.w3.org/RDF/, (accessed 2020-01-08)

[12] Dublin Core Metadata Initiative. DCMI: Dublin Core Metadata Element Set, Version 1.1: Reference Description. https://www.dublincore.org/specifications/dublin-core/dces/, (accessed 2020-01-08)

[13] Dublin Core Metadata Initiative. The Singapore Framework for Dublin Core Application Profiles. https://www.dublincore.org/specifications/dublin-core/singapore-framework/, (accessed 2020-01-08)

[14] “ヨコハマ・アート・LOD　利活用事例の紹介” 横浜のアート・イベント検索サイト ヨコハマ・アートナビ. https://artnavi.yokohama/lod-case/, (accessed 2020-01-08)

[15] Linked Open Data Challenge 2019(LOD チャレンジ 2019). https://2019.lodc.jp, (accessed 2020-01-08)

[16] 統計 LOD. http://data.e-stat.go.jp/lodw/, (accessed 2020-01-08)

[17] DBpedia Japanese. http://ja.dbpedia.org, (accessed 2020-01-08)

[18] “Europeana などのデジタルコレクション活用を図るプロジェクト“TuEuropeana”（ポーランド）”. カレントアウェアネス・ポータル. https://current.ndl.go.jp/node/39806, (accessed 2020-01-08)

[19] SPARQL 1.1 Query Language. https://www.w3.org/TR/sparql11-query/, (accessed 2020-01-08)

[20] LinkData.org. http://linkdata.org, (accessed 2020-01-08)

[21] “Dataset | Search.” Google Devlopers. https://developers.google.com/search/docs/data-types/dataset, (accessed 2020-01-08)

[22] 岡嶋 成司, 山根 昇平, 粂 照宣. (2015). “Linked Data 活用を促進するプラットフォーム”. 人工知能 30 巻 5 号.

[23] Ivan Ermilov, Jens Lehmann, Michael Martin, Soren Auer. (2016). “LODStats: The Data Web Census Dataset”. Proceedings of 15th International Semantic Web Conference(ISWC2016).

[24] 山中勇樹. (2018). “アプリケーション開発事例を用いた LOD データセットの探索支援”. 筑波大学, 卒業研究論文.

[25] “鯖江市内ＡＥＤ設置場所”. LinkData. http://linkdata.org/work/rdf1s284i, (accessed 2020-01-08)

[26] “流山市 AED 設置場所”. LinkData. http://linkdata.org/work/rdf1s648i, (accessed 2020-01-08)

[27] Tomas Mikolov, kai Chen, Greg Corrado, Jeffrey Dean. (2013). “Efficient Estimation of Word Representations in Vector Space”. Proceedings of the International Conference on Learning Representations 2013.

[28] “API guide”. CKAN 2.7.3 documentation. https://docs.ckan.org/en/ckan-2.7.3/api/, (accessed 2020-01-08)