2022 年度

修　士　論　文

異種ネットワークのグラフ埋め込みによるサービスクラスタリング

指導教員: 村上 陽平

立命館大学大学院 情報理工学研究科
博士課程前期課程 情報理工学専攻
計算機科学コース

学生証番号: 6611210017-3

氏名: 大井 也史

異種ネットワークのグラフ埋め込みによる サービスクラスタリング

大井 也史

内容梗概

複数の Web サービスを組み合わせて作成されたサービスのことを複合サービスという.複合サービスにより，個々に提供される原子サービスよりも拡張した機能を提供することができる．現在，多種多様な Web サービスが増えており，それらを組み合わせて新たな複合サービスが構築されている．多くのサービスからユーザが必要に応じて適切なサービスを発見できるように，機能に応じて Web サービスをクラスタリングする必要がある．この問題に対して，WEB サービス記述ファイル(以下，WSDL ドキュメントと呼ぶ)やサービス説明文をテキストマイニングし，サービス機能ごとにクラスタリングする研究がある.しかしながら，これらの手法ではテキストの記述内容に大きく依存するため，サービス提供者の命名規則による影響を受けやすい.

そこで，本研究では，サービス連携関係に加え,利用者とサービスの利用関係を用いてサービス機能ごとのクラスタリングを行う．利用者とサービスの利用関係を用いることで，同じような利用者に利用されているサービス同士は同種のサービスであるという特徴を埋め込むことが可能である．具体的には，同一の複合サービスに組み合わされたことのある原子サービスの連携関係，サービスを提供している提供者とそのサービスを利用している利用者の利用関係をそれぞれグラフで表し，グラフ埋め込みを用いてノードの特徴ベクトルを生成し，各ベクトル間の類似度を計算してサービス機能ごとのクラスタリングを行う． 本手法の実現にあたり，取り組むべき課題は以下の 2 点である.

サービス提供者利用者の情報を用いたネットワークの構築とサンプリング

本研究ではサービス連携関係に加え,利用者とサービスの利用関係を用いたグラフを表す必要がある．また，新たに構築したネットワークのサンプリング方法を考案する必要がある.

異種グラフの埋め込みの統合方法

サンプリングした利用者とサービスの利用関係の特徴をどのようにサービス連携関係のサンプリング結果とあわせて統合するのか，手法を考案する必要がある.

一つ目の課題に対しては,サービスの提供者利用者利用関係から提供者共利

用ネットワーク・サービス提供利用ネットワークを構築し,サンプリングを行う.具体的には提供者共利用ネットワークは提供者をノード，エッジをサービスの利用者とし，ネットワークを構築する．サービスサービス提供利用ネットワークは提供者・利用者をノード，エッジをサービスの利用とした．これらのネットワークを構築したのちに探索を行い，サンプリングを行う．従来手法では，サービス連携ネットワークは同一複合サービス内を優先して探索し，サンプリングを行っていたが，新たに構築したネットワークは node2vec に搭載されている幅優先探索に似た探索手法でネットワーク探索を行い，サンプリングする．また，利用者とサービスの利用関係を表すグラフを構築する際に，提供者を用いる．この提供者の中に一つののサービスしか提供していない Anonymous 提供者があり，ネットワークにおける影響を調査する．

二つ目の課題に対しては,サービス連携関係ネットワークから作成したベクトルと新たに構築したサービスの提供利用利用関係に基づくネットワークから作成したベクトルを Concat し，そのベクトルを用いてサービスのクラスタリングを行なった.

提案手法によってグラフ埋め込みを用いて作成したサービスベクトルのクラスターと人手で付与したカテゴリに基づく正解クラスターと比較することで，生成したクラスターがサービスのカテゴリごとにクラスタリングされていたかを評価した．本研究の貢献は以下のとおりである.

サービス提供者利用者の情報を用いたネットワークの構築とサンプリング

サービス提供者利用者利用関係からネットワークを構築し,サンプリングした．この結果を用い，後述の異種グラフの埋め込みの統合方法により，サービスのカテゴリごとのクラスタリング精度が向上した.

異種グラフの埋め込みの統合方法

サービスの利用関係のサンプリング結果から得られたベクトルに,提供者利用者利用関係のサンプリング結果から得られたベクトルを Concat することで,サービスのカテゴリごとのクラスタリングの精度が従来手法と比較し,6.6%向上した.

Service clustering by graph embedding of heterogeneous networks

Narifumi Oi

Abstract

A service created by combining multiple Web services is called a composite service, which can provide extended functions beyond the atomic services provided individually. At present, a wide variety of Web services are increasing, and new composite services are being constructed by combining them. From many services, Web services need to be clustered according to function so that users can discover the appropriate service as needed. In response to this problem, there have been studies in which Web service description files (Hereafter referred to as a WSDL document) and service descriptions are text-mined and clustered for each service function, but since these methods rely heavily on text descriptions, they are susceptible to the naming rules of service providers.

Therefore, in this study, clustering is performed for each service function by using user and service usage relationships in addition to service linkage relationships. By using the relationship between users and services, it is possible to embed the feature that services used by similar users are the same kind of services. Concretely, the linkage relationship of atomic services that have been combined into the same composite service, the usage relationship between the provider providing the service and the user using the service are represented by graphs, the feature vectors of nodes are generated using graph embedding, and the degree of similarity between each vector is calculated to perform clustering for each service function. In the implementation of this method, the following two issues need to be addressed.

Network construction and sampling using provider, user information

In this study, it is necessary to represent a graph using user and service usage relationships in addition to service linkage relationships. It is also necessary to devise a sampling method for the newly constructed network.

How to integrate embedding of heterogeneous graphs

It is necessary to devise a method for integrating the characteristics of the sampled user and service usage relationship with the sampling results of the

service linkage relationship.

For the first problem, we construct a provider co-use network and a service provider user network from the service provider user use relationship, and conduct sampling. Concretely, the provider co-use network constructs the network with the provider as the node and the edge as the user of the service. The service provider user network consists of provider/user as node and edge as service use. After constructing these networks, search is carried out and sampling is performed. In the conventional method, the service linkage network preferentially searches and samples in the same composite service, but the newly constructed network performs network search and samples in a search method similar to the broad-priority search installed in node2vec. The provider is also used when constructing a graph showing the relationship between users and service usage. Among these providers is an Anonymous provider that provides only one service, and we investigate the impact in the network.

For the second task, we concat vectors created from the service linkage relationship network and those created from the newly constructed network based on the service provision/use relationship, and use these vectors to cluster services.

By comparing a cluster of service vectors generated by the proposed method using graph embedding with a correct cluster based on manually assigned categories, we evaluated whether the generated clusters were clustered for each category of service.

Network construction and sampling using provider, user information

The network was constructed and sampled from the usage relationship between service providers and users. Using this result, the clustering accuracy for each category of service was improved by the integration method of embedding heterogeneous graphs described later.

How to integrate embedding of heterogeneous graphs

By concatting the vectors obtained from the sampling results of service usage relationships with the vectors obtained from the sampling results of provider user usage relationships, the accuracy of clustering for each category of service was improved by 6.6% compared with the conventional method.

目次

第1章 はじめに

複数の Web サービスを組み合わせて作成されたサービスのことを複合サービスという.複合サービスにより，個々に提供される原子サービスよりも拡張した機能を提供することができる．現在，多種多様な Web サービスが増えており，それらを組み合わせて新たな複合サービスが構築されている．多くのサービスからユーザが必要に応じて適切なサービスを発見できるように，機能に応じて Web サービスをクラスタリングする必要がある．この問題に対して，WEB サービス記述ファイル(以下，WSDL ドキュメントと呼ぶ)やサービス説明文をテキストマイニングし，サービス機能ごとにクラスタリングする研究がある.しかしながら，これらの手法ではテキストの記述内容に大きく依存するため，サービス提供者の命名規則による影響を受けやすい.

そこで，本研究では，サービス連携関係に加え,利用者とサービスの利用関係を用いてサービス機能ごとのクラスタリングを行う．利用者とサービスの利用関係を用いることで，同じような利用者に利用されているサービス同士は同種のサービスであるという特徴を埋め込むことが可能である．具体的には，同一の複合サービスに組み合わされたことのある原子サービスの連携関係，サービスを提供している提供者とそのサービスを利用している利用者の利用関係をそれぞれグラフで表し，グラフ埋め込みを用いてノードの特徴ベクトルを生成し，各ベクトル間の類似度を計算してサービス機能ごとのクラスタリングを行う． 本手法の実現にあたり，取り組むべき課題は以下の 2 点である.

サービス提供者利用者の情報を用いたネットワークの構築とサンプリング

本研究ではサービス連携関係に加え,利用者とサービスの利用関係を用いたグラフを表す必要がある．また，新たに構築したネットワークのサンプリング方法を考案する必要がある.

異種グラフの埋め込みの統合方法

サンプリングした利用者とサービスの利用関係の特徴をどのようにサービス連携関係のサンプリング結果とあわせて統合するのか，手法を考案する必要がある.

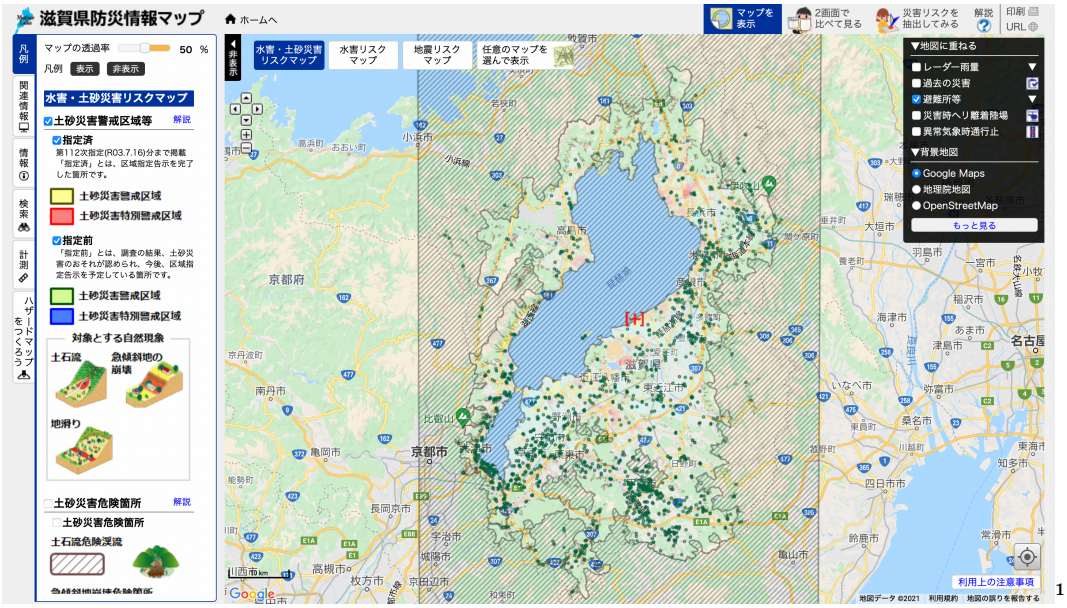
以下，本論文では，2 章において従来の研究のクラスタリング手法として，関連研究１，関連研究２，関連研究３について説明する．次に，3 章では本研究で用いた各ネットワークの解説や，コサイン類似度，ジャカード類似度の計算方法

について説明する．続いて，4 章ではノードの分散表現に基づくクラスタリングであるグラフ埋め込み技術について説明を行い，ネットワークのサンプリング手法やグラフ埋め込みで用いる skip-gram model について説明する．5 章では 3 章と 4 章で説明を行なったクラスタリング手法に対する評価を行い, 6 章にて考察を行い，7 章にて今後の展望や課題について述べて結論とする．

第2章 サービスクラスタリング

本章では，既存の類似機能サービスのクラスタリング手法について説明する．サービスクラスタリングは，複合サービスを作成する際に，その構成要素の候補となるサービスを探し出したいときに有用である．

複合サービスのサンプルケースとして，図 1 の滋賀県防災情報マップを用いる．滋賀県防災情報マップでは，防災情報と地図の API である”google maps”，または国土地理院の地図を表示している．滋賀県防災情報マップでは地図情報である”googlemaps”と国土地理院の地図を切り替えることができる．このように，複合サービスを作成する際，必要なジャンルのサービスを発見する必要がある．しかしながら，オンライン上に存在する膨大な数の Web サービスの中から類似サービスを発見するのは非常に困難である.このような場合，Web サービスを自動的に分類することが必要になり，クラスタリングは Web サービスを分類するための非常に効率的なアプローチの 1 つである．本章では，このアプローチに関する研究として，サービス説明文に基づく手法とオントロジーに基づく手法を説明する．



[1]

図 1：複合サービスの例

[1] https://shiga-bousai.jp/dmap/top/index

2.1 サービス説明文に基づくクラスタリング

LDA モデルとは文書のトピックを推定するモデルで，文書の分類や検索などに利用される手法である．Min らは，Web サービスを word2vec によって拡張された LDA モデルを用いてクラスタリングする手法を提案した．WSDL ドキュメントから特徴を抽出し，サービスを機能ごとに抽出する手法では，WSDL ドキュメントに記述されている単語が限られているためクラスタリング品質が低くなるという問題がある．そこで，サービス説明文と LDA モデルを用いてサービスの機能ごとのクラスタリングを行う．具体的には LDA モデルを word2vec に拡張することで，精度の高い単語ベクトルを取得する．そして，取得した単語ベクトルを類似性に基づいてクラスタリングを行う．クラスタリングによって構築されたクラスタの情報はサービスのベクトルを構築する際に使用する．最後に，k-means などのクラスタリングアルゴリズムに基づき，全ての Web サービスを異なる機能ドメインにクラスタリングするか，潜在的なトピックに従って，同じトピックに属する Web サービスを一緒にクラスタリングする．

2.2 インタフェースに基づくクラスタリング

インタフェース情報に基づくクラスタリング手法とは，ユーザが推薦されるサービスはユーザが指定したサービスの機能を表す単語に類似するものである．Elgazzar らは，WSDL ドキュメントからサービスの機能を表す用語を収集し，それらを機能的に類似したサービスクラスタリングを行う方法を提案した．この提案手法を行うには，前もって以下の 2 工程が必要である．

1. WSDL ドキュメントからクラスターを作成する
2. クラスターからユーザの要求する Web サービスを発見する

1 つ目の WSDL ドキュメントからクラスターを作成するステップについて説明する．はじめに検索エンジンのクローラーを使い，インターネット上から WSDL ドキュメントをクロールする．次に WSDL ドキュメントから Web サービスの意味や動作などを表す用語を抽出するために,WSDL タイプ，WSDL メッセージ，WSDL ポート，および Web サービス名から単語を抽出する.これらの単語を結合して，Web サービスを機能的に類似したグループにクラスター化する．これは検索エンジンが Web サービスを識別し，ユーザ要求を満たす Web サービスを発見するための処理である．

2 つ目のクラスターからユーザの要求する Web サービスを見つけるステップ

について説明する．はじめにユーザからサービス検索エンジンに目的の単語でクエリを実行する. 次に 1 つ目のステップで作成されたクラスターと入力されたクエリを意味的に一致させる．これは例えば，「bike」と「scooter」のような意味は同じだが単語が違うもの同士を一致させるための処理である. 最後に，要求された目的を満たす最も関連性の高い Web サービスをユーザに返す.

第3章 近傍に基づくクラスタリング

本章では，コサイン類似度，ジャカード類似度の計算で用いるネットワークの定義とその計算方法について説明する．

3.1 全体のネットワークモデル

本研究で用いる複合サービスのモデルと提供者・利用者のモデルを紹介する．本研究で用いるデータモデルは図 2 のようになっている．原子サービスは，1 つの提供者があり，複数の利用者がいる．また，複合サービスに組み込まれている場合がある．提供者は 1 つ以上のサービスを提供しており，複数の原子サービスを提供している場合がある．複合サービスは複数の原子サービスを複合している．このデータモデルから複合サービスのモデルと提供者・利用者のモデルを抜き出し，各ネットワークを構築する．

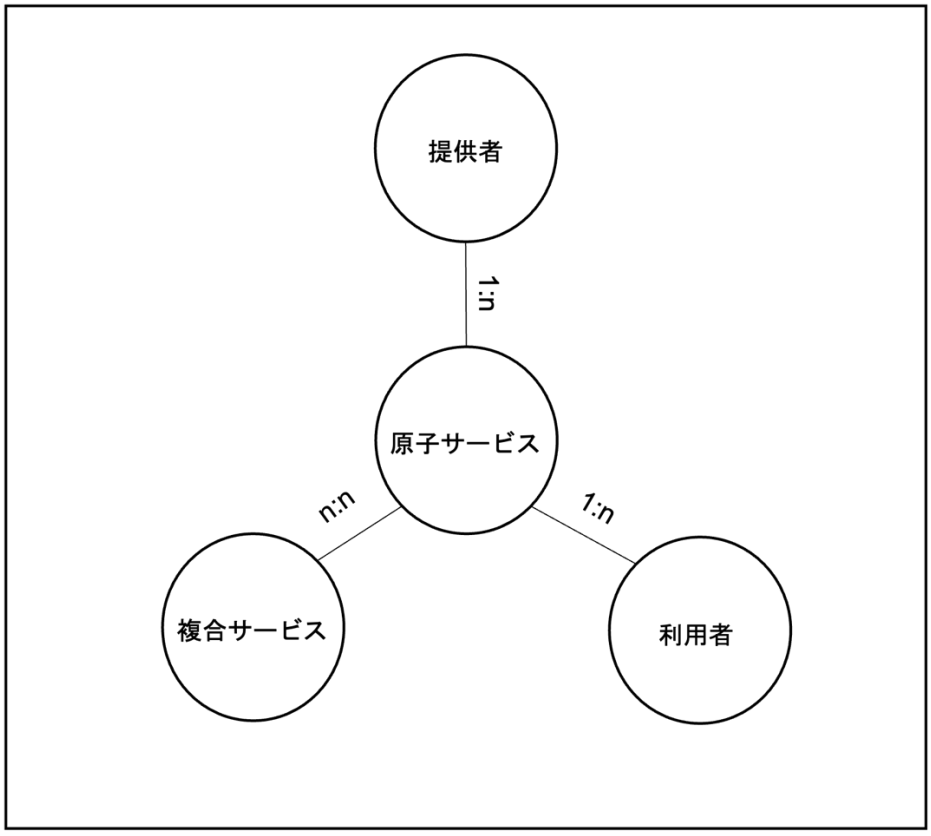


図 2：全体のデータモデル

3.1.1 複合サービスのモデル

本研究では複合サービスのモデルとして，以下の図 3 のような複合サービスと原子サービスの関係を用いる．複合サービスは複数の原子サービスが複合されて構築されている．よって，複合サービスと，その複合サービスに複合されている原子サービスのノードを複合関係によって繋げることで，図 3 のようなモデルとなる．このデータモデルを用いて，後述のネットワークを構築する．

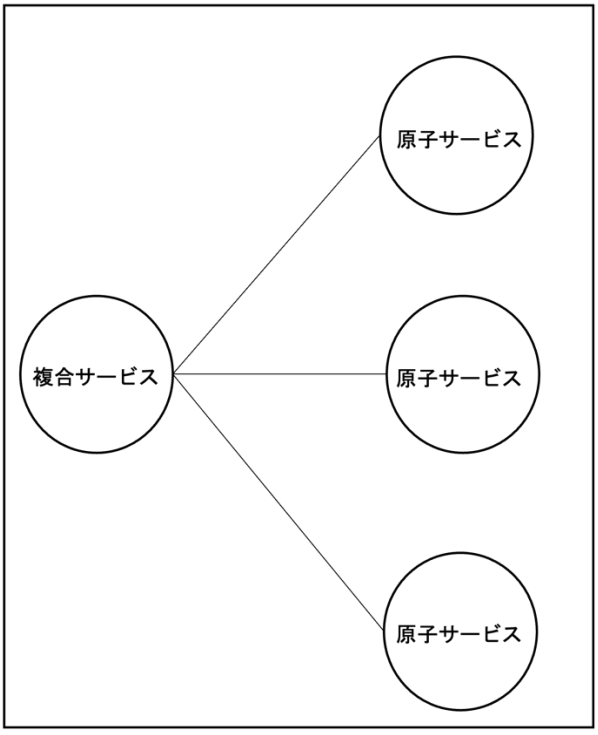


図 3：複合サービスと原子サービスデータモデル

3.1.2 提供者・利用者のモデル

本研究では提供者・利用者のモデルとして，以下の図 4 のような提供者と利用者の関係を用いる．提供者とは，原子サービスをリリースしている提供者である．また，利用者とは，提供者がリリースしている原子サービスを利用している利用者である．提供者ノードと原子サービスノードを提供関係，原子サービスノードと利用者ノードを利用関係で結ぶことによって，以下の図 4 のようなモデルとなる．このデータモデルを用いて，後述のネットワークを構築する．

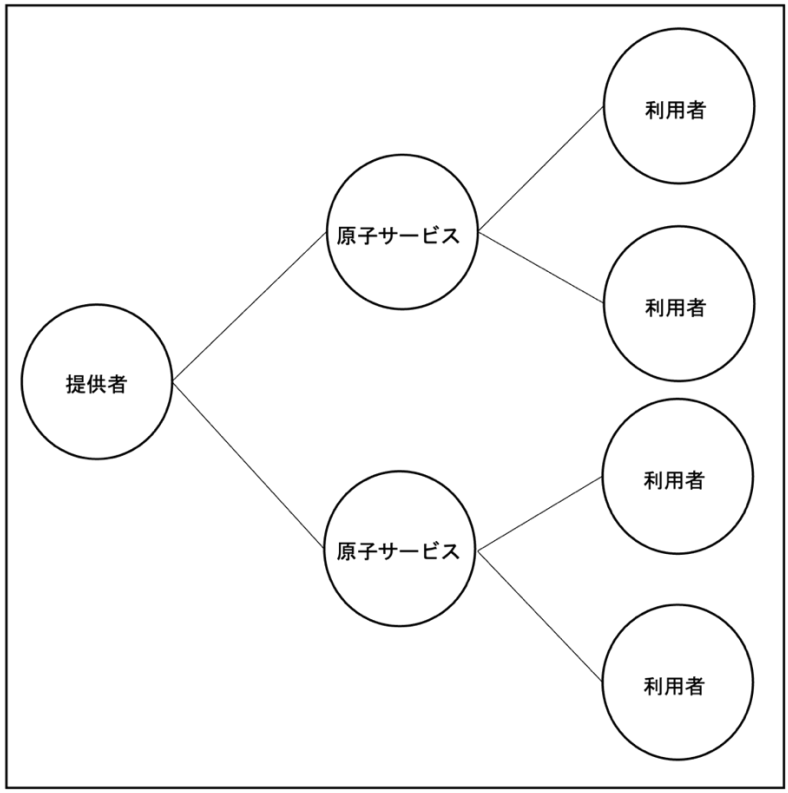


図 4：提供者と利用者のデータモデル

3.2 サービス連携ネットワーク

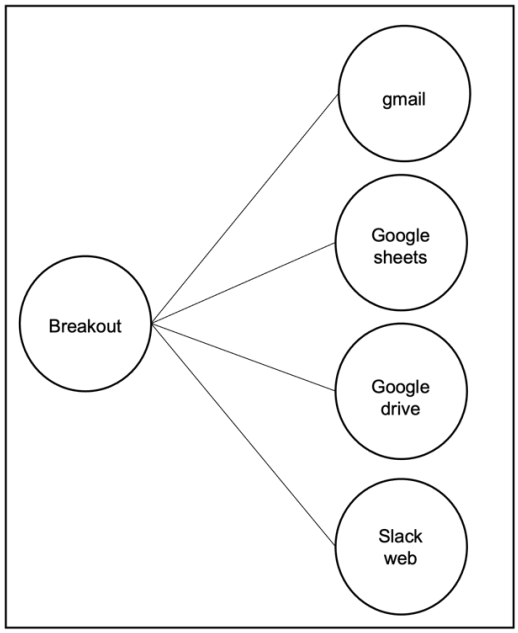


図 5：複合サービスの例

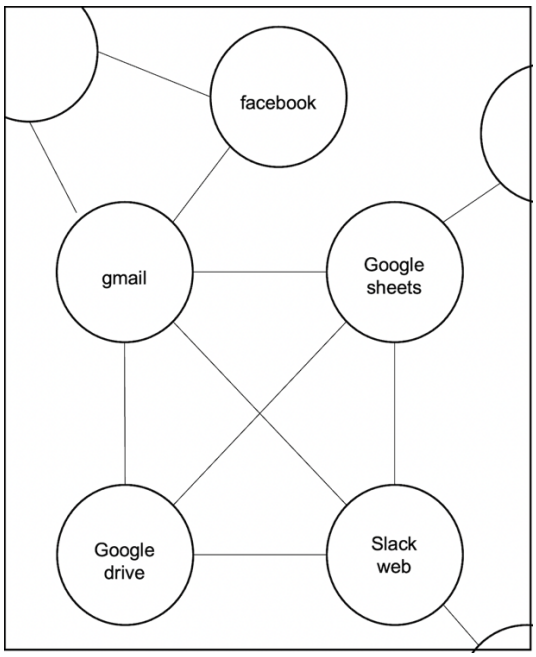


図 6：サービス連携ネットワークの例

複合サービスに組み合わされている原子サービスデータを用いて，複合サービスによるサービス連携の依存グラフを構築する．複合サービスの例として，図 5 の”Breakout”を示す．複合サービス”Breakout”は原子サービスである”gmail”，”Google sheets”，”Google drive”，”Slack web”が組み込まれている．このような複合サービスから，図 6 のようなグラフを作成する．このグラフでは，ある複合サービスで”gmail”，”Google sheets”，”Google drive”，”Slack web”が組み合わされていたことから，この 4 つのノードを生成し，エッジを繋いでいる.また，他の複合サービスで”gmail”と”facebook”が組み合わされていた場合，新しく”facebook”ノードを作成し，”gmail”ノードとエッジを繋ぐ.

3.3 サービス共利用ネットワーク

サービスを提供している提供者と，サービスを利用している利用者の情報を用い, 提供者･利用者による提供者の依存グラフを構築する. 利用者の例として，利用者 a，利用者 b を表 1 にしめす．このような提供者・利用者の情報から，

表 1：利用者の例

	利用している提供者	繋ぐエッジ
利用者 a	ABC	AB,AC,BC
利用者 b	BCD	BC,BD,CD

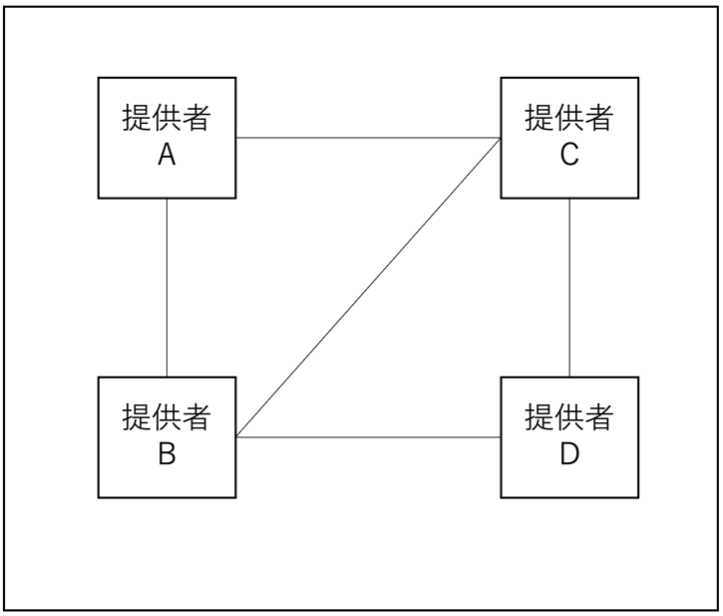


図 7：提供者共利用ネットワークの例

図 7 のようなグラフを生成する．具体的には，利用者 a は提供者 A,B,C が提供している原子サービスを利用しているので，ノード A,B,C を生成し，エッジを繋ぐ．

また，利用者 b は提供者 B,C,D が提供している原子サービスを利用しているので，ノード D を新しく生成し，ノード B,C,D の間でエッジを繋ぐ．

3.4 サービス提供利用ネットワーク

サービスを提供している提供者と，サービスを利用している利用者の情報を用い，提供者・利用者による提供者の依存グラフを構築する利用者の例として，利用者 a，利用者 b を表 2 にしめす．このような提供者・利用者の情報から，図 8 のようなグラフを生成する．具体的には，利用者 a は提供者 A,B,C が提供している原子サービスを利用しているので，利用者ノード a と提供者ノード A,B,C を

表 2：利用者の例

	利用している 提供者	サービス提供利用 ネットワークで繋ぐエッジ
利用者 a	A,B,C	aA,aB,aC
利用者 b	B,C,D	bB,bC,bD

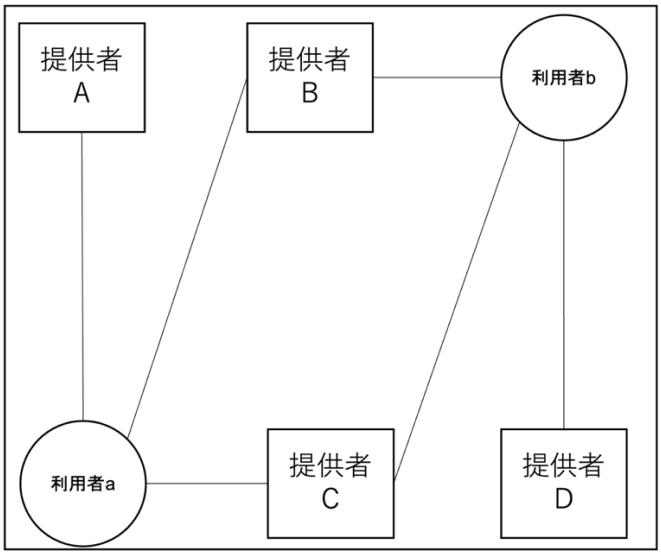


図 8：サービス提供利用ネットワークの例

生成し，利用者ノード a と提供者ノード A,B,C の間にエッジを繋ぐ．同様に，利用者 b は提供者 B,C,D が提供している原子サービスを利用しているので，利用者ノード b と提供者ノード D を生成し，利用者ノード b と提供者ノード B,C,D の間にエッジを繋ぐ．

3.5 コサイン，ジャカード類似度を用いた類似度計算

サービス間の類似度の計算にはコサイン類似度とジャカード類似度を用いる．コサイン類似度では，２つのベクトルの角度の近さで類似度を算出する．具体的には，２つのベクトルを A，B とすると，以下の式によって算出できる．

$$\cos(A, B)\frac{A \cdot B}{|A||B|}$$

またジャカード類似度では，2 つの集合同士の和集合の割合で類似度を算出する.具体的には，2 つの集合を U,V とすると，以下の式によって算出できる.

$$jaccard(U,V)\frac{|U \cap V|}{|U \cup V|}$$

それらを用いて類似度計算を行うアルゴリズムを図 9 に示す.コサイン類似度，ジャカード類似度を用いた類似度計算では，まず 3.1 で述べたネットワークを構築し，全サービスの隣接ノード群を取得する．例として図 6 を用いると ”gmail” の隣接ノード群は ”Google sheets”，”Google drive”，”Slack web”，”facebook”となる．次に，全サービスの総当たりの類似度を計算し，その結果を用いて類似機能サービス群にクラスタリングする.

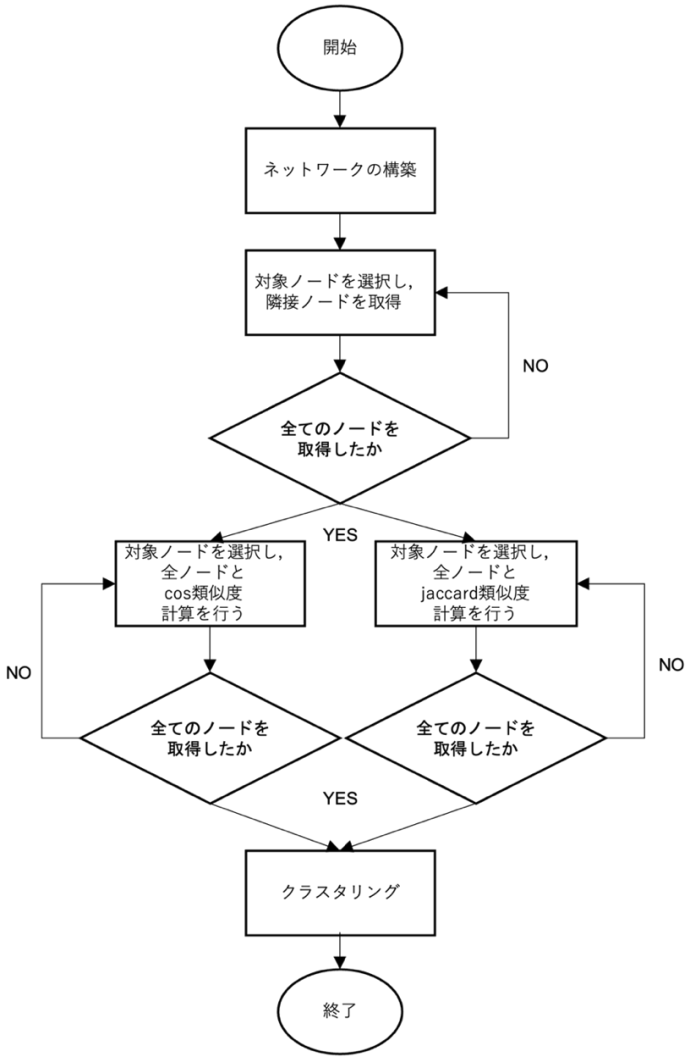


図 9：類似度計算のアルゴリズム

第4章 グラフ埋め込み

グラフ埋め込みとはネットワークデータのノードやエッジ，サブグラフなどの分散表現を得る手法のことであり，特に skip-gram model をネットワーク構造に対して拡張した手法がいくつか提案されている. 本節では, skip-gram model のネットワーク構造への拡張手法を説明した後，ネットワークデータのサンプリング手法として同一複合サービス優先のサンプリングや共利用優先のサンプリングについて説明する．また，最後に異種グラフの統合方法として，Concat，クラスタの統合について説明する.

4.1 Skip-gram model

skip-gram model とはニューラルネットワークモデルの 1 つである. 具体的には，単語分散表現を得る学習モデルである．単語の分散表現の獲得には，意味が近い単語は周辺単語も似ているというという学習により実現する．以下の例文を用いて説明する.

・私はりんごを切って食べる

・私はなしを切って食べる

これらの文があるとき,「りんご」と「なし」は,「私は」「切って」「食べる」に囲まれている．つまり,「りんご」と「なし」は似た使われ方をしていることがわかる．この時の「りんご」や「なし」のような単語を入力データ,「食べる」のような周辺単語を教師データとし，単語に対するその周辺単語の重みを学習させることで, 各単語の分散表現を得ることができる．この手法で得た単語ベクトルは，以下のように単語の足し算や引き算が可能となる.

・王 + 女性 = 女王

王ベクトルと女性ベクトルを足し合わせることで，女王ベクトルに近いベクトルを得ることができる. また, skip-gram model と似た手法として cbow model が存在する. skip- gram model が中心語から周辺単語を予測するのに対し, cbow model は周辺単語から中心単語を予測する．また，一般的には cbow model よりも skip-gram model の方が予測精度は高い．skip-gram model をネットワーク構造に対応させると，入力データは各ノード, 教師データは各ノードのサンプリングで表すことができる.

4.2 Node2vec

skip-gram model を用いたグラフ埋め込み技術として node2vec が存在する．Node2vec のアルゴリズムを以下の図 10 にしめす．

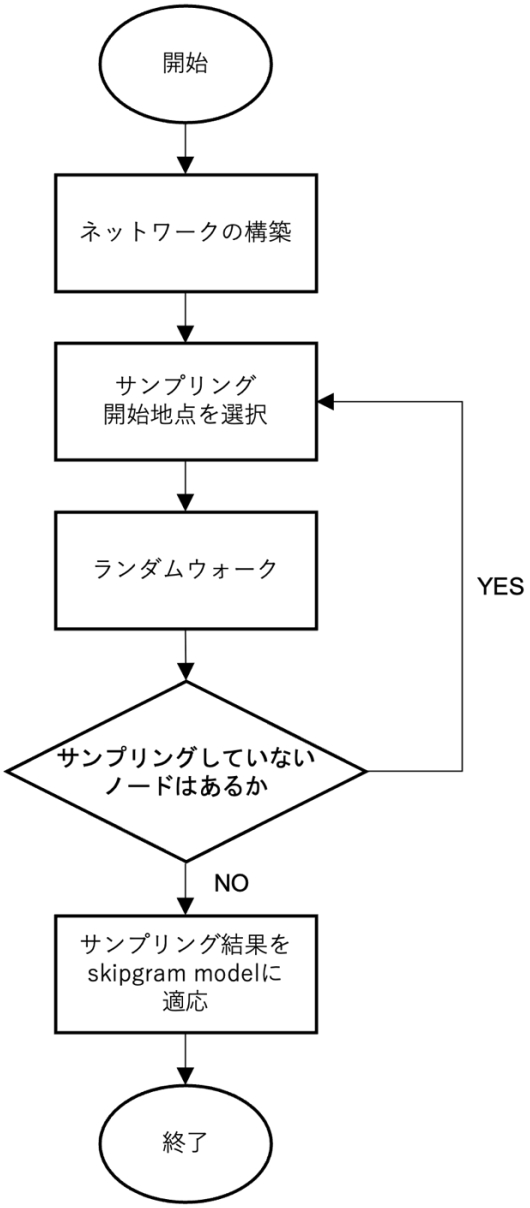


図 10：Node2Vec のフローチャート

Node2vec は用意したネットワークの探索結果を用いて，ノードの分散表現を作成する．具体的な手順としては，構築したネットワークからサンプリングの始点となるノードを選択し，ランダムウォークで任意の長さのサンプリングを行う．全てのノードからサンプリングを行えた後，それらを各ノードが空白で区切

られたテキストデータとして扱い，探索結果として保存する．保存した探索結果を skip-gram model に適応し，各ノードの分散表現を獲得する．node2vec の特徴として，ランダムウォークでノードのサンプリングを行う際，幅優先探索に似た探索と深さ優先探索に似た探索の度合いをハイパーパラメーターで制御することができる．ランダムウォークは静的エッジの重みに基づいて次のノードをサンプリングするが，ネットワークの構造を考慮し，検索手順をガイドして異なるタイプの近隣ネットワークを探索することができない．さらに，幅優先と深さ優先が競合または排他的ではない．そこでパラメーターである return parameter(以下，p)，In-out parameter(以下，q)を調整してサンプリングを行う．これらを使用した 2 次のランダムウォークを下記に定義する．また，図 11 において，エッジ(t,x)を横断し，次に探索するノード$s_i(i = 1,2,3)$を考える．d_{tx}はノード t と x 間の最短経路距離であり，d_{tx}は$\{1,2,3\}$のいずれかである．

$$a_{pq}(t,s) = \begin{cases} \frac{1}{p} if\ d_{ts} = 0 \\ 1\ if\ d_{ts} = 1 \\ \frac{1}{q} if\ d_{ts} = 2 \end{cases}$$

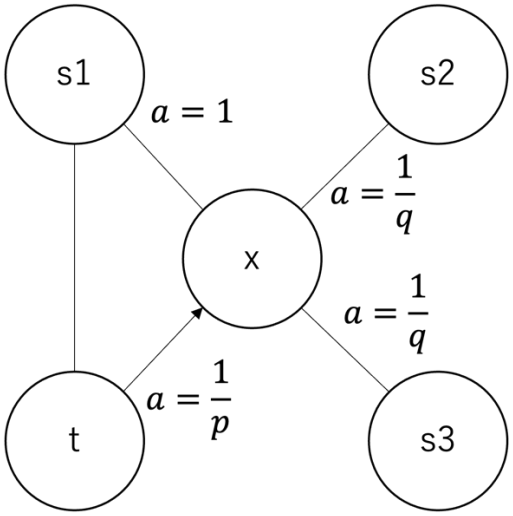


図 11：ランダムウォークの例

パラメーターp は歩行中のノードを再訪する確率を制御する．このパラメーターを高い値(>max(q,1))に設定することで，既に探索したノードをサンプリングする可能性を低くすることができる．また，低い値(<min(q,1))にすることで，サンプリングの開始ノードから離れにくくすることができる．パラメーターq は幅優先探索に似た探索と深さ優先探索に似た探索の確率を制御する．q>1 の場合，ランダムウォークはノード t に近いノードに偏り，幅優先探索に似た探索を行うことができる．対照的に，q<1 の場合はノード t からさらに離れたノードを訪問する傾向があり，深さ優先探索に似た探索を行うことができる．しかし，幅優先探索に似た探索を重視しすぎると近いノードの情報しか得ることができない．また，深さ優先探索に似た探索を重視しすぎるとサンプリング範囲が大きく深くなるので，この 2 つのバランスが重要となる．

4.3 サンプリング

node2vec のサンプリング手法とは別のアプローチとして，サービス連携ネットワークにおける，同一複合サービス優先のサンプリングと，サービス共利用ネットワークにおける共利用優先のサンプリングについて説明する．

4.3.1 同一複合サービス優先のサンプリング

サンプリングの大まかなフローチャートを以下の図 12 に示す. 同一複合サービス優先のサンプリングを用いるサービス連携ネットワークは，複合サービスにより組み合わされたことのある原子サービスの連携関係でネットワークを構築しており，それぞれの原子サービスがどの複合サービスで組み合わされたのか表されていない. また, 同一複合サービスで組み合わされているノード同士は完全ネットワークとなるので，それぞれのノードが密接になってしまい，node2vec で用いられているランダムウォークではそれぞれのノードの特徴を取得するのは非常に難しい. そこで, サンプリング結果に複合サービス情報を反映させるため，同一複合サービスを優先するサンプリング手法を行う．

サービス連携ネットワークを構築した後，サンプリングを行う視点となるノードをランダムで選択する．選択したノード（原子サービス）が含まれている複合サービスの事例をランダムに選択し，その複合サービスに複合されている原子サービスを優先してサンプリングを行う．選択した複合サービスに含まれているノードをサンプリングし終えたら，ランダムに未訪問のノードを選択し，同

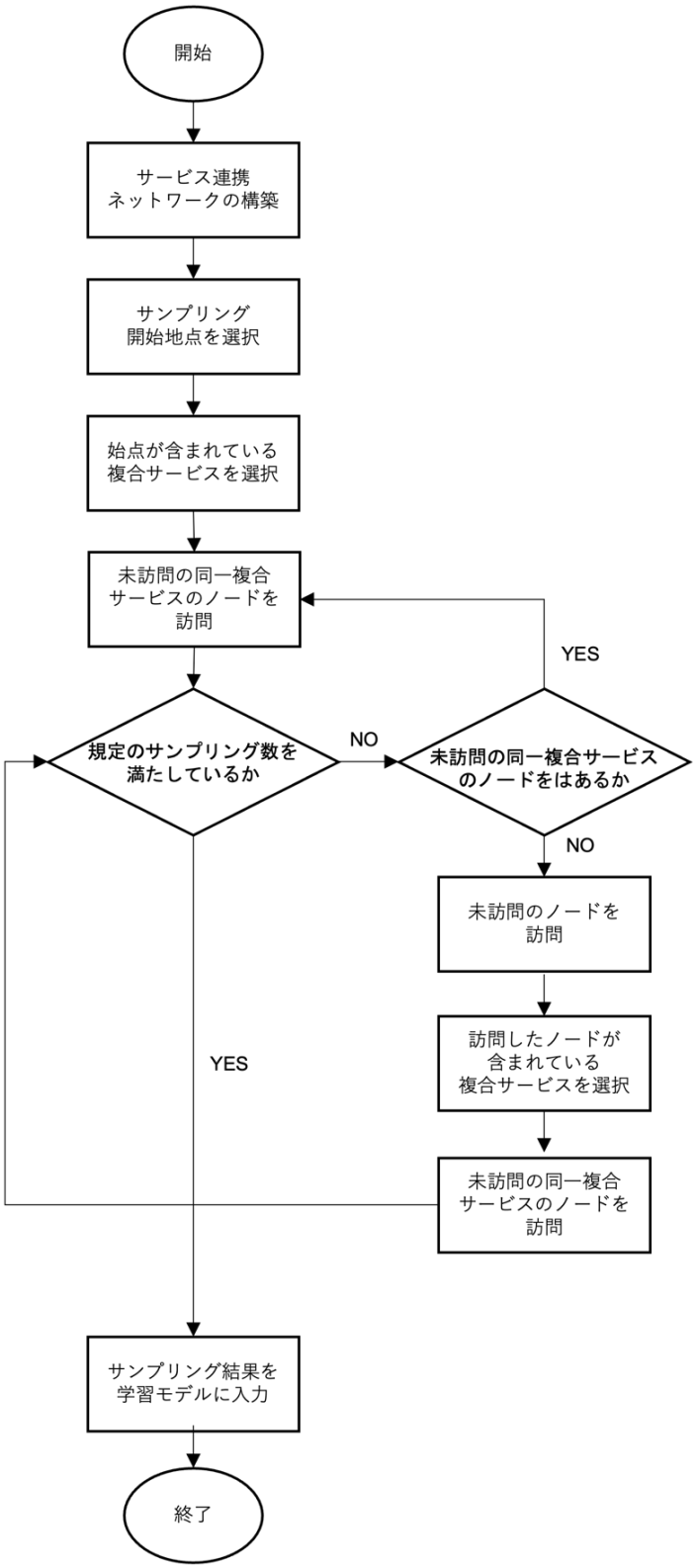


図 12：同一複合サービス優先のサンプリングのフローチャート

一複合サービス優先のサンプリングを規定回数繰り返す．サンプリングが完了したら，サンプリング結果の各ノードを空白で区切った文字列にし，テキストデータとして保存し，word2vec などの skip-gram model によって学習し，各ノードの分散表現を得ることができる．

また，以下の図 13，図 14 用いて同一複合サービス優先のサンプリングを具体的に説明する．双方ともサービス A のノードから 4 つ目のノードまで探索を行うとする．図 13 はランダムウォークを行った場合の探索結果例で，探索結果は以下のようになる．

サービス A→サービス B→サービス E→サービス G

ランダムウォークを行った場合，結果は上記のようになる場合があり，探索が同一複合サービス外に出てしまうことがある．また，図 14 は同一複合サービス優先の探索を行った場合の探索結果例で，結果は以下のようになる．

サービス A→サービス B→サービス D→サービス C

ランダムウォークを行った場合，同一複合サービス内のノードを全て探索し終える前に，他の同一複合サービスに含まれているノードを探索することがある．同一複合サービス優先のサンプリングにより，探索を開始した同一複合サービ

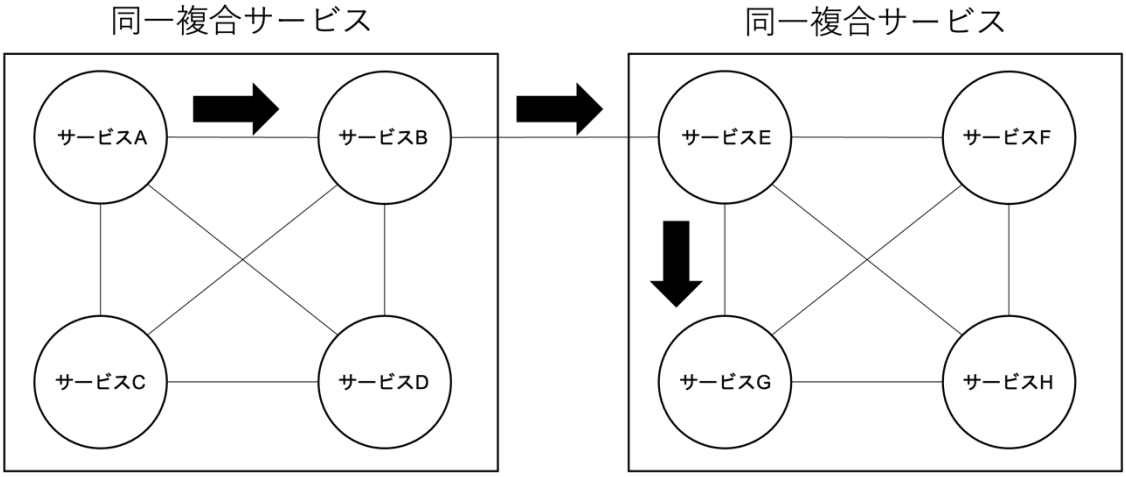


図 13：ランダムウォークを行った場合の探索

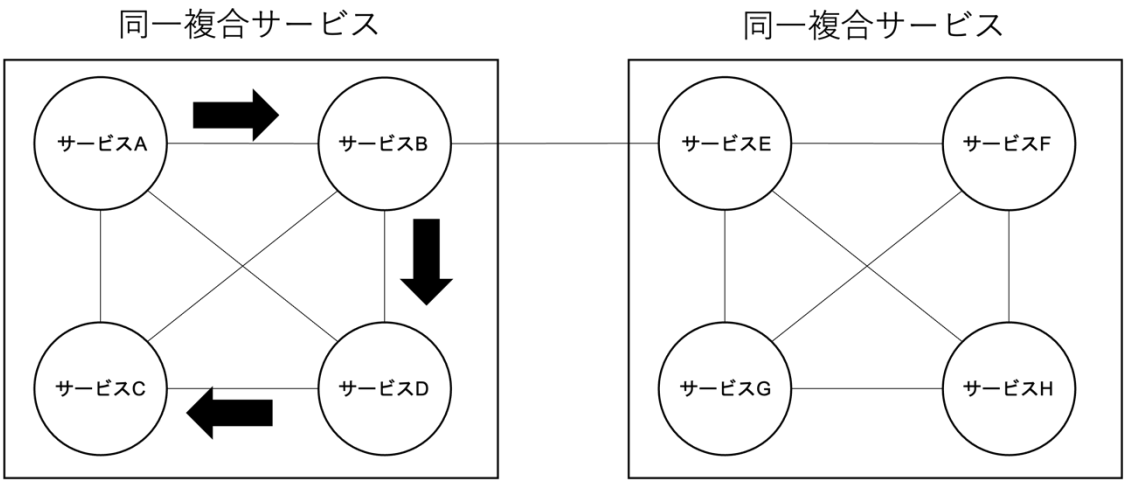


図 14：同一複合サービス優先の探索を行った場合の探索

ス（サービス A,サービス B,サービス C,サービス D）を優先して探索するため，同一複合サービス内を探索し終える前に他の同一複合サービス内に複合されているサービス（ノード）を探索することはない．よって，探索結果が，原子サービスが同一複合サービスに含まれているという特徴をより引き出すことができる．また，同一複合サービスを優先するサンプリング手法において，周辺ノードは同一複合サービスで組み合わされたノード群になるので，似た構成を持つ複合サービスに組み合わされたノード同士のベクトルは近くなることが期待できる．

4.4 異種グラフの統合

本研究ではサービス連携ネットワークとサービス共利用ネットワークの二種類のネットワークを作成し，利用している．二つのネットワークのサンプリングを行い，グラフ埋め込みを行うことで生成される分散表現を統合する必要がある．本研究では Concat と Add2，Mul2 の三つの手法を用いた．各手法について説明する．

4.4.1 Concat

サービス連携ネットワークからグラフ埋め込みによって得た原子サービスのベクトルと，サービス共利用ネットワークからグラフ埋め込みによって得た提

供者のベクトルを結合する．具体的には，原子サービスのベクトルとサービス共利用ネットワークから得た原子サービスの提供者のベクトルを統合する．以下に例を示す．

表 3：サービスの例

	タグ	ベクトル
原子サービス	Google maps	[1,2,3]
原子サービスの提供者	Google	[2,3,4]

上記の表 3 の例の場合，原子サービスである”Google maps”のベクトルと，”Google maps”の提供者である”Google”のベクトルを Concat によって統合する．”Google maps”のベクトルは[1,2,3]，”Google”のベクトルは[2,3,4]なので，Concat により統合すると，結果は以下のようになる．

$$[1,2,3] + [2,3,4] = [1,2,3,2,3,4]$$

4.4.2 Add2

サービス連携ネットワークからグラフ埋め込みによって得た原子サービスのベクトルと，サービス共利用ネットワークからグラフ埋め込みによって得た提供者のベクトルを Add2 という手法で統合する．具体的には，原子サービスのベクトルとサービス共利用ネットワークから得た原子サービスの提供者のベクトルを統合する．4.4.1 で用いた例で説明する．”Google maps”と”Google”のベクトルを統合するとしたら，2 つのベクトルを要素ごとに加算する．結果は以下のようになる．

$$[1,2,3] + [2,3,4] = [3,5,7]$$

4.4.3 Mul2

サービス連携ネットワークからグラフ埋め込みによって得た原子サービスのベクトルと，サービス共利用ネットワークからグラフ埋め込みによって得た提供者のベクトルを Mul2 という手法で統合する．具体的には，原子サービスのベクトルとサービス共利用ネットワークから得た原子サービスの提供者のベクト

ルを統合する．4.4.1 で用いた例で説明する．”Google maps”と”Google”のベクトルを統合するとしたら，2 つのベクトルを要素ごとに乗算する．結果は以下のようになる．

$$[1,2,3] + [2,3,4] = [2,6,12]$$

第5章 評価

第 3 章で説明したサービス連携ネットワークと提供者共利用ネットワークの二つの異種ネットワークを構築し，第 4 章で説明したグラフ埋め込みの手法を用いて複合サービスに組み合わされる原子サービスを，類似機能を持つサービスグループにクラスタリングする．本章では，はじめに実験データを説明する．そして，提供者教理用ネットワークに用いられる Anonymous 提供者について説明する．次に，生成クラスタと正解クラスタの対応づけのパターンについて説明する．最後に，様々な手法やそのクラスタリング精度について説明した後，それぞれの結果について考察する．

5.1 評価手法

5.1.1 実験データ

本研究では，programmableweb に掲載されている複合サービスデータ 6467 件，全 api のデータ 22469 件を用いる．取得した複合サービスのデータには，複合サービス名，複合サービスのジャンル，複合サービスの説明文，複合サービスに複合されている原子サービスの情報が json 形式で記述されている(図 15)．このフォーマットの定義を図 16 に示す．また，取得した全 api のデータには，

```
"vacationic": [
[
"travel",
"mapping",
"photos",
"video"
],
[
" Provides an overview of travel highlights…
],
[
"flickr",
"google-adsense",
"google-ad-manager",
"youtube"
],
]
```

図 15：収集した複合サービスの例

api 名，ジャンル，api の説明文，提供者，利用者の情報が json 形式で記述されている．(図 17)．このフォーマットの定義を図 18 に示す．取得したデータからサービス連携ネットワーク，サービス協利用ネットワークを構築する．

```
複合サービス名: [
[
複合サービスのジャンル
],
[
複合サービスの説明文
],
[
複合されている原子サービス
],
]
```

図 16：複合サービスのデータ形式

```
"budbee": [
[
"shipping",
"nordic",
"postal",
"products"
],
"The Budbee API returns home product delivery…
[
"budbee-ab"
],
[
”user_1",
”user_2",
”user_3"
],
```

図 17：収集した原子サービスの例

```
原子サービス名: [
[
原子サービスのジャンル
],
原子サービスの説明文
[
原子サービスの提供者名
],
[
原子サービスの利用者名
],
```

図 18：原子サービスのデータ形式

5.1.2 使用するパラメータ

本手法では node2vec などを用いてグラフ埋め込みを行う際, さまざまなパラメータを用いる必要がある．それらのパラメータの一覧とその説明を以下の表 4 に示す．従来手法において，各パラメータの適切な数値が導き出されており，その数値を本研究においても用いる．それらの数値を以下の表 5 に示す.

表 4：グラフ埋め込みに用いるパラメータ

次元数	出力するベクトルの次元数
Walk_length	ネットワーク探索の長さ
Num_walk	1 つのノードからサンプリングする回数
パラメータ p	ネットワークを探索する際，1 つ前の訪問ノードに戻る確率
パラメータ q	q<1 で深さ優先探索に似た探索を，q>1 で幅優先探索に似た探索を行う確率

表 5：実際に用いるパラメータ

次元数	128
Walk_length	30
Num_walk	50
パラメータ p	0.1
パラメータ q	q<1

5.1.3 Anonymous 提供者

Anonymous 提供者とは，提供者の情報が載っていない原子サービスの提供者である．つまり，一種類の api のユーザーにしか利用されていない提供者である．以下の図 19，図 20 にサービス共利用ネットワークにおける Anonymous 提供者と Anonymous 提供者ではない提供者の例を示す．図 19 はサービス共利用ネットワークにおける通常の提供者の例である．この提供者は複数の原子サービスをリリースしているため，複数の原子サービスの利用者によるエッジが敷かれている．対して Anonymous 提供者は一つの原子サービスしかリリースされていないため，一つの原子サービスの利用者によるエッジしか敷かれていない．

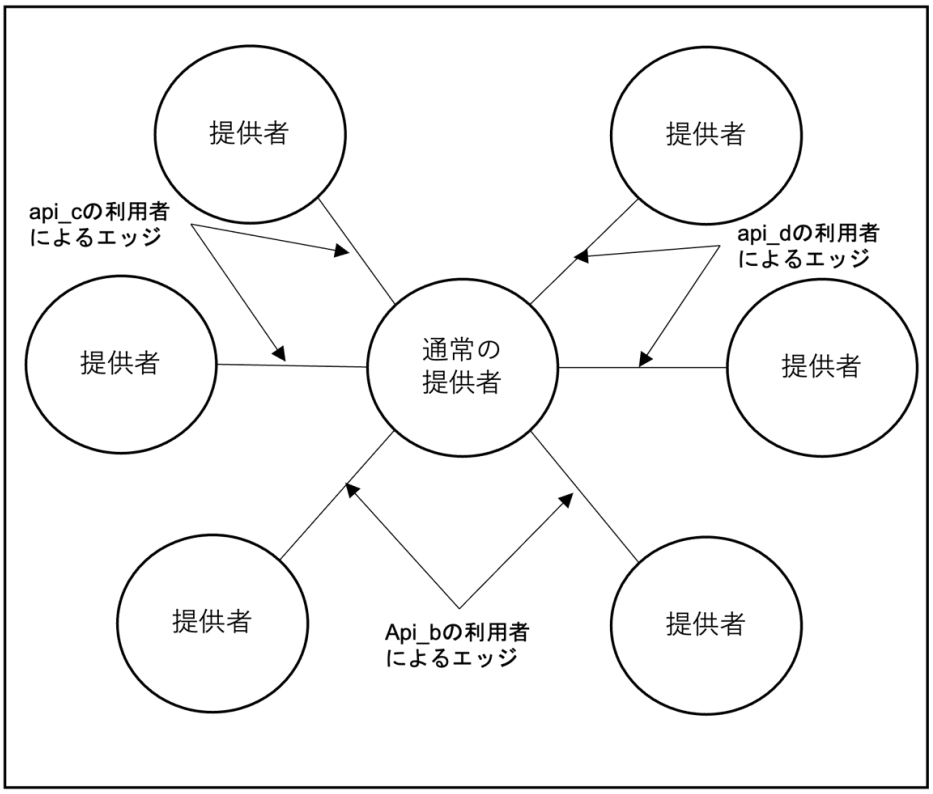


図 19：通常の提供者の例

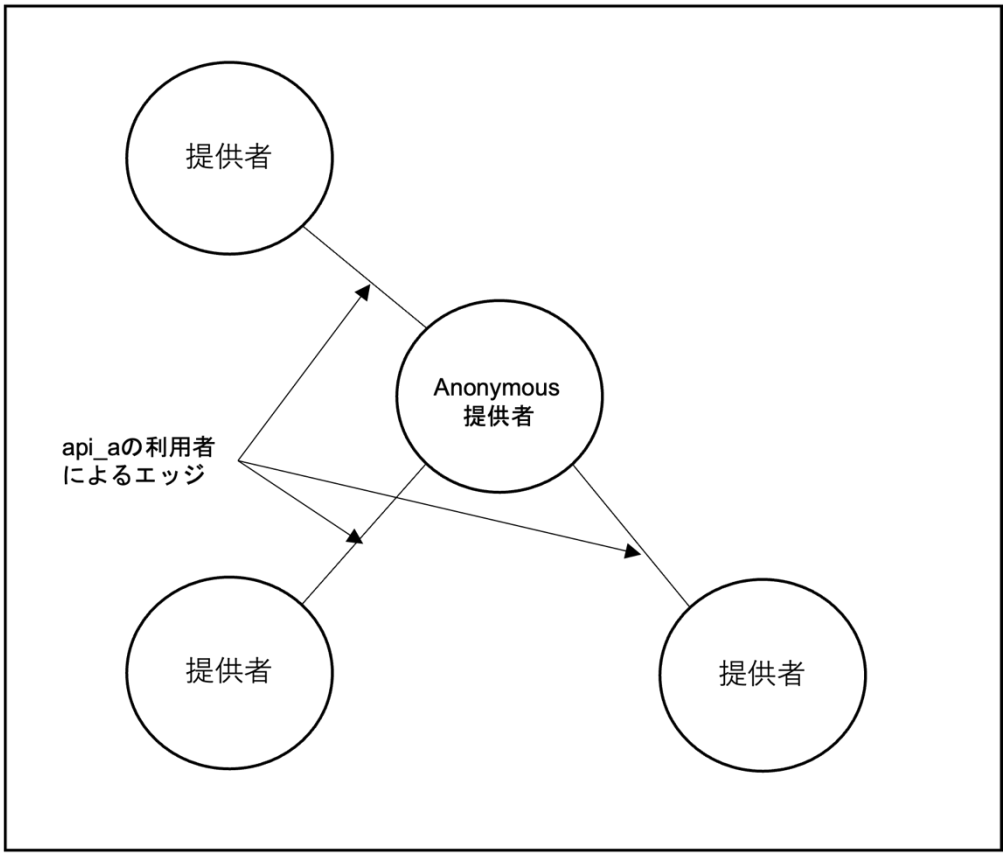


図 20：Anonymous 提供者の例

5.1.4 提供者の Anonymous 化

提供者の Anonymous 化とは，5.1.2 で説明した通常の提供者を Anonymous 提供者に変換するということである．5.1.2 の図 19 と以下の図 21 を用いて説明する．図 19 では例示された提供者は api_b，api_c，api_d の利用者によって他の提供者へのエッジが引かれている．この提供者ノードを api_b の利用者のエッジにのみ繋がれる提供者ノード，api_c の利用者のエッジにのみ繋がれる提供者ノード，api_d の利用者のエッジにのみ繋がれる提供者ノードにそれぞれ分けることによって，以下の図 21 のようになる．Anonymous 化により，多数の原子サービスをリリースしている”Google”などの巨大なノードを減らすことができる．

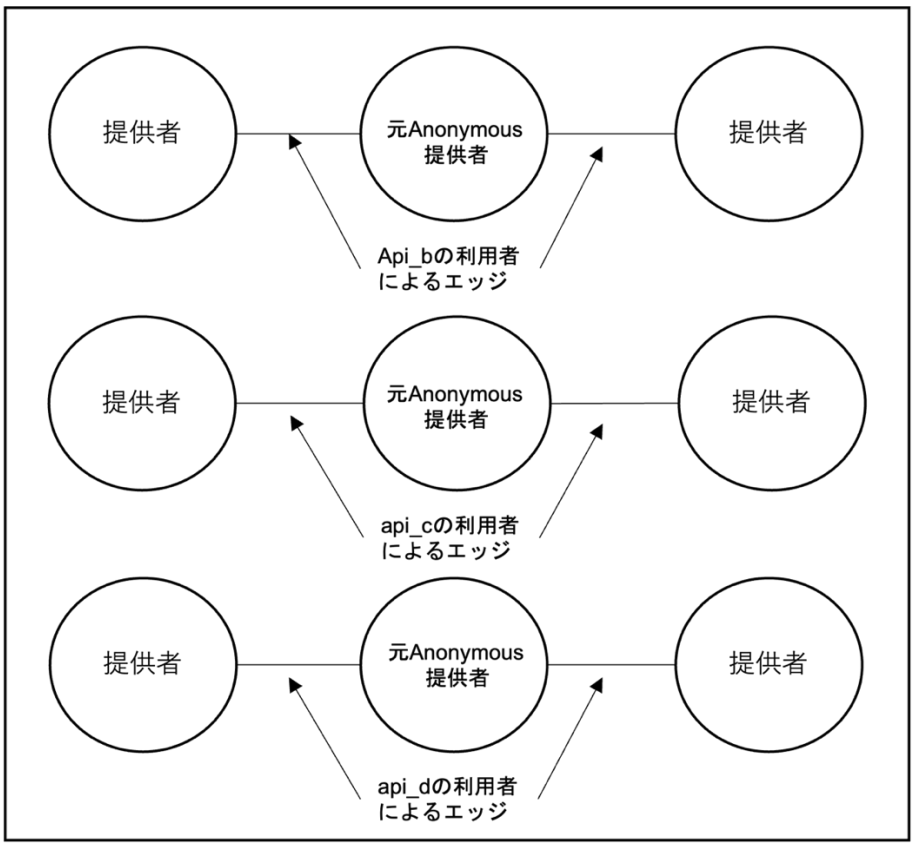


図 21：提供者の Anonymous 化の例

5.1.5 サービス連携ネットワーク

4.3.1 で説明した，サービス連携ネットワークを programmableweb から収集したデータを用いて構築した．ネットワークのトポロジーを以下に示す．ノード数は，実験に用いた原子サービスのノード数となり，535 個である．このネットワークは従来手法で用いられたネットワークと同一のものである．このサービス連携ネットワークにおいて複合サービス優先おサンプリング行い，Word2Vec を用いて埋め込みを行うことで，各ノードである原子サービスのベクトルを作り出す．

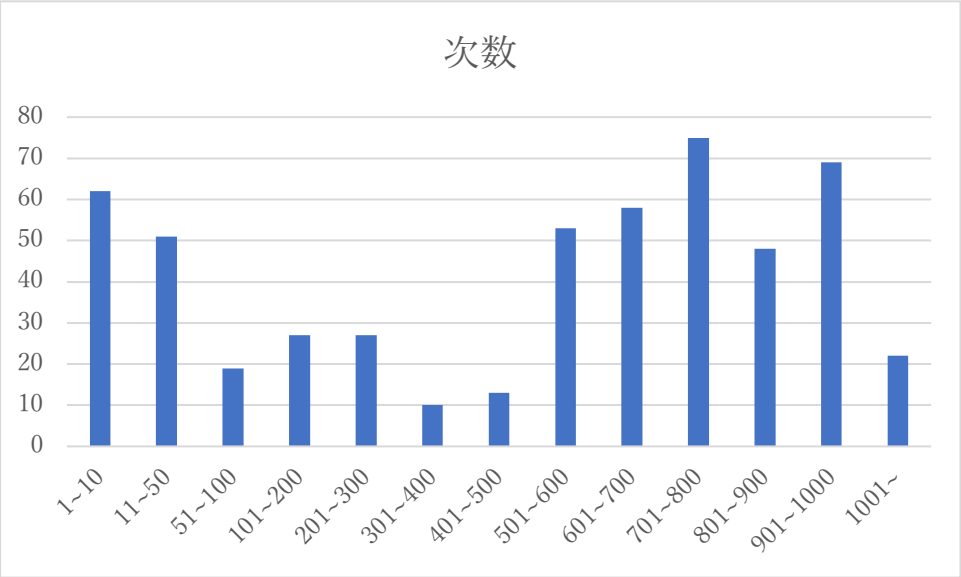


図 22：サービス連携ネットワークの次数分布

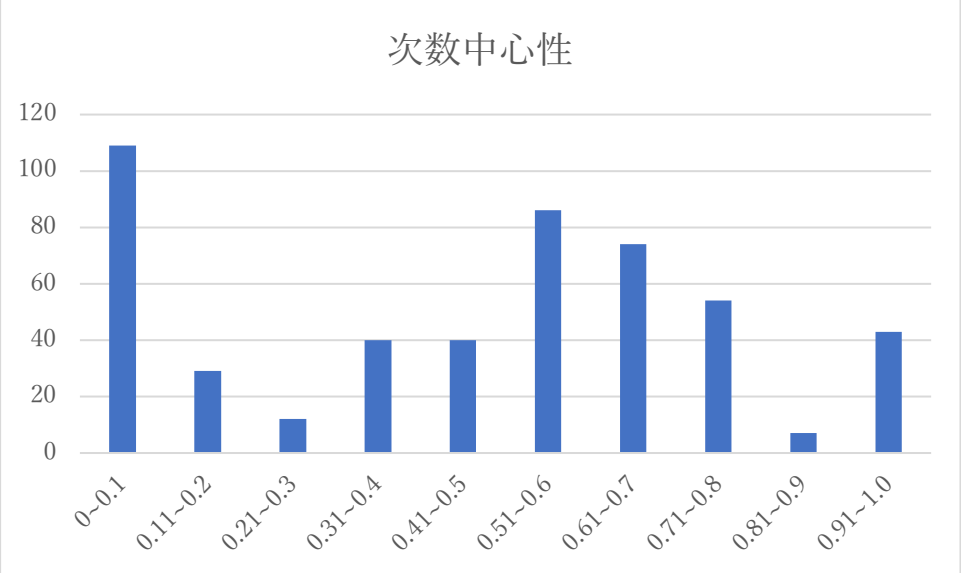


図 23：サービス連携ネットワークの次数中心性

5.1.6 サービス共利用ネットワーク

4.3.1 で説明した，サービス共利用ネットワークを programmableweb から収集したデータを用いて構築した．ネットワークのトポロジーを以下に示す．ノード数は，実験に用いた提供者の数となり，提供者の Anonymous 化前は 12472 個，

Anonymous 化後は 200954 である．Anonymous 化によって提供者のノード数が増加していることがわかる．また，Anonymous 化によって多数のエッジが接続されている巨大なノードが解体されたことから，大きい次数のノードが減少し，小さい次数のノードが増加した

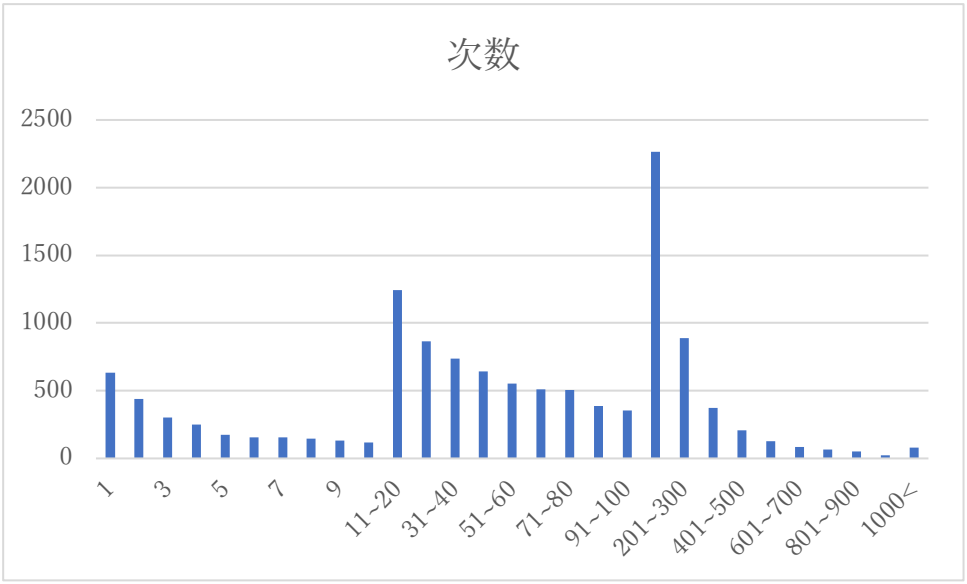


図 24：サービス共利用ネットワークの次数

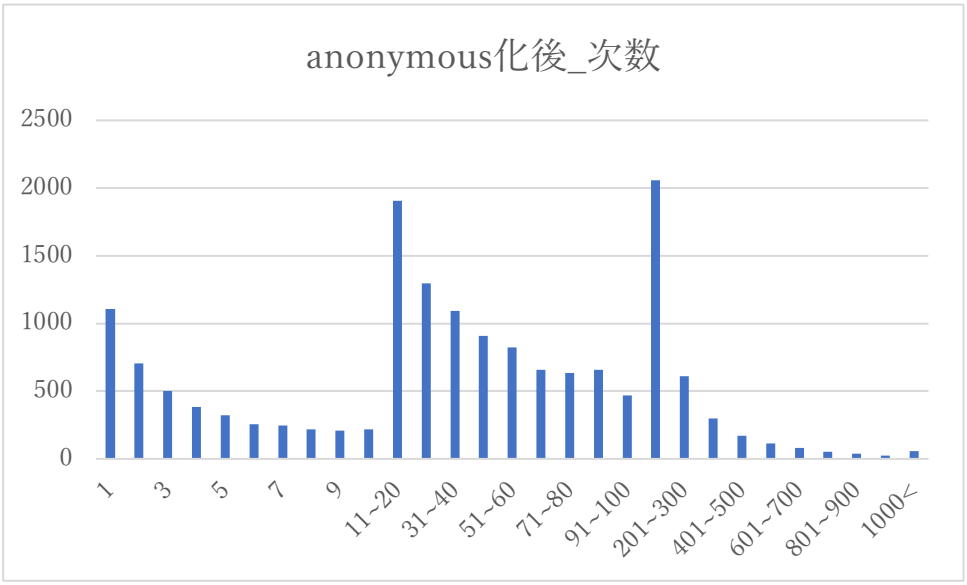


図 25：提供者共利用ネットワークの anonymous 化後の次数

5.1.7 サービス提供利用ネットワーク

4.3.1 で説明した，サービス連携ネットワークを programmableweb から収集したデータを用いて構築した．ネットワークのトポロジーを以下に示す．ノード数は，実験に用いた提供者の数と利用者の数となり，提供者の Anonymous 化前

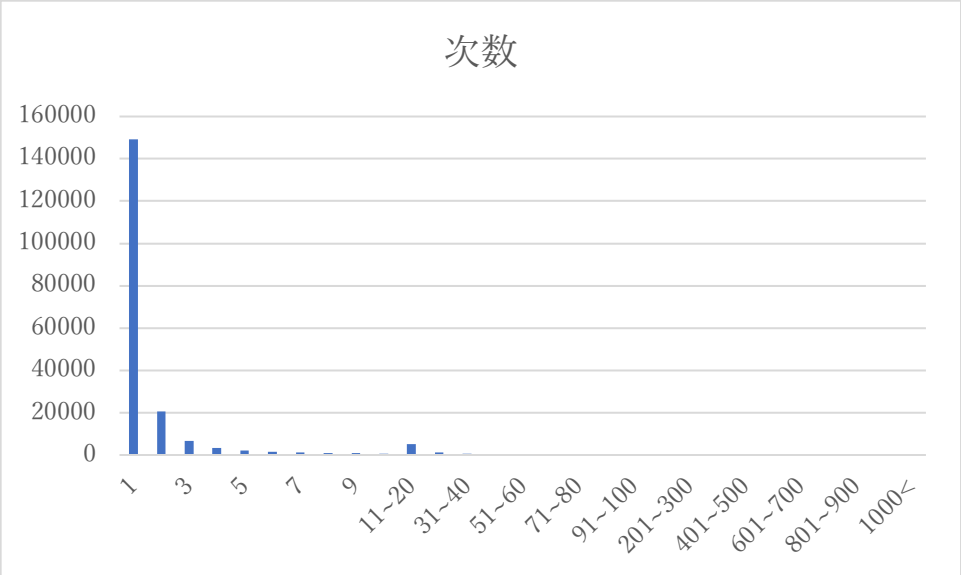


図 26：サービス提供利用ネットワークの次数

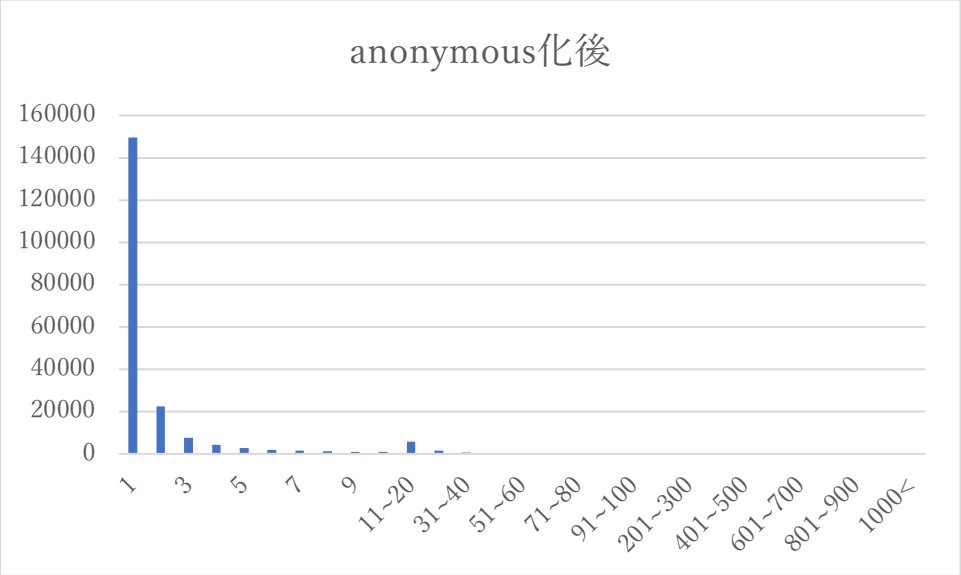


図 27：サービス提供利用ネットワークの Anonymous 化後の次数

は 195112 個，Anonymous 化後は 16123 である．Anonymous 化によって提供者のノード数が増加していることがわかる．また，Anonymous 化によって多数のエッジが接続されている巨大なノードが解体されたことから，大きい次数のノードが減少し，小さい次数のノードが増加した

5.1.8 評価指標

各手法により生成されるクラスタリング結果の有効性を指す指標として，以下の Purity 値を求めた.

$$Purity = \frac{1}{N}\Sigma_i \max_j |sc_i \cap c_j| \quad (1)$$

数式(1)では，各手法において生成したクラスタ sc と正解クラスタ c の要素を比較し，最も共通項の多い組み合わせを作り，共通項の総和をサービス総数 N で割っている．これにより，対象となる全サービスの何割が正しいクラスタに所属しているか数値的に判断することができる.ここでの，正解クラスタは Programmableweb に掲載される人手で付与された各サービスのカテゴリ情報を基に作成している.

5.1.9 生成クラスタと正解クラスタの対応づけ

5.1.6 における purity 値を計算する際の生成クラスタと正解クラスタの対応付けにおいて，以下の 4 つのパターンが想定される.

・pattern1:正解クラスタを Primary カテゴリのみで作成し，正解クラスタが重複しない対応付けを行う

・pattern2:正解クラスタを Primary カテゴリと Secondary カテゴリで作成し，正解クラスタが重複しない対応付けを行う

・pattern3:正解クラスタを Primary カテゴリのみで作成し，正解クラスタが重複する対応付けを行う

・pattern4:正解クラスタを Primary カテゴリと Secondary カテゴリで作成し，正解クラスタが重複する対応付けを行う

これら 4 つのパターンに分類される理由は以下の通りである.

・各サービスのカテゴリ情報が複数付与されている場合があるため

・複数の生成クラスタが共通の正解クラスタと対応付けされる場合がある

ため

まずは，各サービスの各サービスのカテゴリ情報が複数付与されている場合があることについて説明する．5.1.1 で紹介した，programmableweb から取得した api の情報のように，複数のカテゴリが付与されている．Programmableweb にカテゴリ情報を登録する際に，主なカテゴリを表す Primary カテゴリと他に想起されるカテゴリを表す Secondary カテゴリを付与することができるためである．5.1.1 の図 17 の例だと，原子サービスである”budbee”は Primary カテゴリである，” shipping”，Secondary カテゴリに”nordic”，”postal”，”products”が付与されている．以上の理由により，Primary カテゴリのみで正解クラスタを作成するパターンと，Primary カテゴリと Secondary カテゴリで正解クラスタを作成するパターンが存在することがわかる．

次に，複数の生成クラスタが共通の正解クラスタと対応付けされる場合があることについて説明する．purity 値の計算を行う工程で，最も共通する要素が多い生成クラスタと正解クラスタの組み合わせを見つける工程がある．生成される各クラスタが類似機能サービスでクラスタ化されている時，正解クラスタは唯一に決定する．しかし，類似機能サービスのクラスタが複数生成された場合，共通の正解クラスタと対応付けされる場合が生じる．類似機能サービスクラスタが複数生成され，対応付けされる正解クラスタが重複しないように対応付けを行った場合，2 つ目以降の類似機能サービスクラスタは要素数が次に多いカテゴリで対応付けされることが考えられる．以上の理由により，正解クラスタが重複する対応付けのパターン，正解クラスタが重複しない対応付けのパターンの２パターンが存在することがわかる．

本研究では pattern4 に絞ってクラスタリング精度を計測した．

5.1.10 サービス連携ネットワークを用いたクラスタリング精度

従来手法であるサービス連携ネットワークのみを用いたサービスクラスタリングの精度を紹介する．Purity 値とクラスタリング精度の適合率を用いて評価した．

5.1.11 提供者ベクトルのクラスタリング精度

サービス提供利用ネットワークで作成した提供者ベクトルが原子サービスにどのような影響を与えているのか調査するため，提供者ベクトルを提供者が提供している原子サービスのカテゴリごとに，クラスタリングを行った．

5.1.12 異種ネットワークを用いたクラスタリング精度

サービス連携ネットワークで生成したサービスベクトルとサービス共利用ネットワーク・サービス提供利用ネットワークで生成した提供者ベクトルを統合し，サービスの機能ごとにクラスタリングを行った．また，各埋め込みの統合方法ごとの精度を測った．

5.2 結果

5.1 で説明した評価指標を用い，グラフ埋め込みにより複合サービスに組み合わされる原子サービスを類似機能サービスグループにクラスタリングした結果を以下に示す．また，提供者ベクトルを提供者が提供している原子サービスのカテゴリごとに，クラスタリングを行った結果も示す．

1.サービス連携ネットワークを用いたクラスタリング精度

従来手法であるサービス連携ネットワークのみを用いたサービスクラスタリングの精度を測定した．結果を以下の表 6 に示す．

表 6：サービス連携ネットワークを用いたクラスタリング精度

使用するネットワーク	Purity 値	適合率
サービス連携ネットワーク	0.48412	0.51676

2.提供者ベクトルのクラスタリング精度

5.1 で説明した評価指標を用い，サービス提供利用ネットワークによって生成された提供者ベクトルのクラスタリング精度を測定した．結果を以下の表 7 に示す．適合率が 0.67 とサービスのクラスタリング精度と比較し，高い値になった．

表 7：サービス提供利用ネットワークにおける

提供者ベクトルのクラスタリング精度

	Purity 値	適合率
サービス提供利用ネットワーク	0.25233	0.67478
サービス共利用ネットワーク	0.19252	0.62485

3.異種ネットワークを用いたクラスタリング精度

各異種ネットワークを用いた場合のクラスタリング精度を下記の表 8，表 9，表 10，表 11 に示す．また，各埋め込みの統合方法，提供者を用いたネットワークの Anonymous 化の有無でのクラスタリング精度を以下の表に示す．

表 8：サービス共利用ネットワークを用いた場合のクラスタリング精度

埋め込みの統合方法	Purity 値	適合率
Concat	0.33661	0.41257
Add2	0.32681	0.38319
Mul2	0.32677	0.36165

表 9：サービス共利用ネットワークを Anonymous 化した場合の
クラスタリング精度

埋め込みの統合方法	Purity 値	適合率
Concat	0.34291	0.40399
Add2	0.35057	0.38562
Mul2	0.30268	0.32733

表 10：サービス提供利用ネットワークを利用した場合の
クラスタリング精度

埋め込みの統合方法	Purity 値	適合率
Concat	0.51588	0.54423
Add2	0.34251	0.39487
Mul2	0.33072	0.37084

表 11：サービス提供利用ネットワークを Anonymous 化した場合の
クラスタリング精度

埋め込みの統合方法	Purity 値	適合率
Concat	0.33022	0.39341
Add2	0.34857	0.40854
Mul2	0.32761	0.39760

表 6~表 10 において，最も精度が高くなったのは提供者の anonymous 化を行わずにサービス提供利用ネットワークを利用し，Concat によりベクトルを統合した手法である．purity 値が 0.51588，適合率が 0.54423 となった．

5.3 T-SNE によるクラスタリング結果の可視化

T-SNE により，サービス連携ネットワークによって作られたサービスベクトルと，サービス提供利用ネットワークによって作られた提供者ベクトルを Concat により統合した 256 次元のベクトルを 2 次元に次元削減し，可視化したものが以下の図 28 である．256 次元のベクトルを可視化するには，次元削減により，次元数を 2 次元または 3 次元に削減する必要がある．T-SNE とは，高次元データを 2 次元や 3 次元に落とし込むための次元削減アルゴリズムである．次元削減といえば古典的なものとして PCA や MDS が存在するが、それら線形的な次元削減にはいくつかの問題点があった．例えば，異なるデータを低次元上でも遠くに保つことに焦点を当てたアルゴリズムのため、類似しているデータを低次元上でも近くに保つことには弱いことや，高次元上の非線形的なデータに対しては類似しているデータを低次元上でも近くに保つことは不可能に近いといった問題点がある．そこで，T-SNE は高次元での距離分布が低次元での距

図 28：クラスタリング結果の可視化

離分布にもできるだけ合致するように変換することにより，高次元の局所的な構造を非常によく捉えられることができ，大局的な構造も可能な限り捉えることができるようになった.

第6章 考察

本章では，5.2 節で提示した結果を受けて各異種ネットワークのクラスタリング精度について考察を与えたのち，異種グラフの統合に関しての考察を行う．

6.1 異種ネットワークごとのクラスタリング結果

従来手法であるサービス連携ネットワークを用いたクラスタリングと本手法であるサービス提供利用ネットワークを利用した場合のクラスタリングを比較する．サービス提供利用ネットワークを利用した場合のクラスタリング手法は従来手法であるサービス連携ネットワークを用いたクラスタリングに比べ約6.6%の精度向上が見られる．そこで，サービス連携ネットワークで得られたサービスベクトルに加え，サービス提供利用ネットワークで得られた提供者ベクトルを Concat することでどのようにクラスタが変化したのかを例を用いて考察

表 12：music カテゴリのサービス例

	正しく分類できた	間違った分類
サービス連携関係 ネットワーク	'songkick', 'musicbrainz', 'lyricfind', 'cloudspeakers', 'musixmatch','lyrdb', 'sonicapicom', 'discogs', 'bandcamp','snocap', 'jambase'	'spokenbuzz','dailymotion', 'contentful','filestube', 'musicbrainz','guardian', 'thetvdbcom'
サービス連携関係 ネットワーク + 利用者提供者 ネットワーク	'snocap', 'rhapsody', 'lastfm', 'musixmatch', 'jambase', 'openstrands', 'cloudspeakers', 'musicbrainz', 'gruvr', 'bbc','lyricfind', 'bandcamp', 'exfm', 'sonicapicom', '7digital', 'songkick', 'lyrdb', 'discogs'	'dailymotion','thetvdbcom'

する．上記の表 12 ではサービス連携ネットワークのみを用いてクラスタリングを行った場合の music カテゴリのクラスタとサービス連携ネットワークとサービス提供利用ネットワークを用いてクラスタリングを行った場合の music カテゴリのクラスタが示されている．サービス連携ネットワークとサービス提供利用ネットワークを用いた場合のクラスタでは，サービス連携関係ネットワークのみのベクトルで正しく分類できたサービスに加え，‘rhapsody‘など，7 つのサービスを，利用者提供者ネットワークを用いることで正しく music カテゴリとして分類できていることがわかる．また，サービス連携関係ネットワークのみのベクトルで正しく分類できなかったサービスは,‘dailymotion’,‘thetvdbcom’以外全て music カテゴリ以外のカテゴリとして分類できたことがわかる．しかしながら，dailymotion’，‘thetvdbcom’のサービスはサービス連携関係ネット

表 13：music カテゴリのクラスタ例

要素数	6
適合率	1.0
カテゴリ	music
生成クラスタ	soundcloud,songkick,discogs,lyrdb,bandcamp,lyricfind

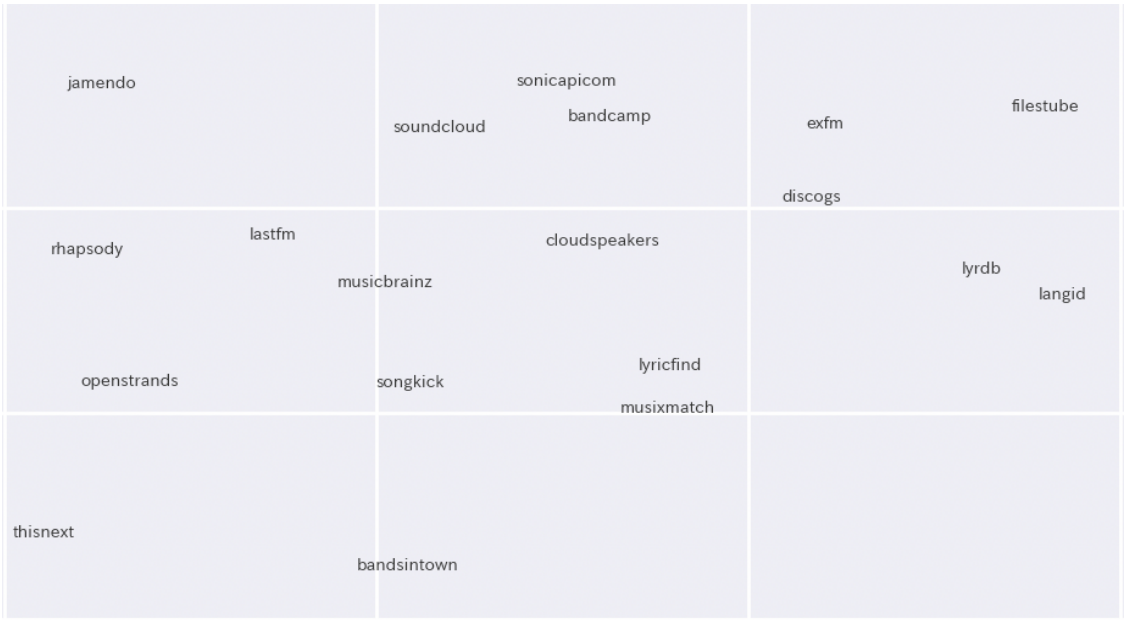


図 29：music カテゴリのクラスタ周辺の可視化

ワークと利用者提供者ネットワークで作成したベクトルであっても，誤ってmusicカテゴリとして分類されてしまっている．また，上記の図 29 は表 13 のmusicカテゴリのクラスタのベクトルや，周囲のベクトルをT-SNEにより次元削減を行い，可視化した図である．生成されたクラスタに含まれているサービスが密集していることがわかる．また，このクラスタには含まれてはいないが、周囲にも同じ music カテゴリである”lastfm”や”musicbrainz”などのサービスが存在することが分かる．

そして，以下の表 14 では，ではサービス連携ネットワークのみを用いてクラスタリングを行った場合の music カテゴリのクラスタとサービス連携ネットワークとサービス提供利用ネットワークを用いてクラスタリングを行った場合の

表 14：mappingカテゴリのサービス例

	正しく分類できた	間違った分類
サービス 連携関係 ネットワーク	'multimap', 'geocoderca', 'ip2location','geoloqi', 'geonames', 'metacarta', 'arcweb','telize', 'wikilocation', 'viamichelin', 'earthtools','tixik', 'everytrail', 'mapfluence', 'geonb','openweathermap', 'cartodb','openstreetmap'	''oodle','vast','uship', 'lyft','uber','rapleaf', 'greatschools', 'finansportalen', 'datagovuk','viator', 'geograph','yellow', 'theyworkforyou','hotelston', 'timezonedb','8coupons', 'jamendo','kiva', 'corpwatch''fcc','opendoar', 'yipit','zoominfo','doodle', 'bandsintown','eventbrite', 'activecom','highcharts'
サービス 連携関係 ネットワーク + 利用者提供者 ネットワーク	'qype', 'multimap', 'metacarta', 'panoramio', 'maplarge', 'geoloqi', 'earthtools', 'geocoderca','openstreetmap', 'cartodb', 'ip2location', 'mapfluence','openlayers', 'geonb', 'geonames', 'tixik','arcweb', 'shizzow','mapbox', 'strava'	'oodle','8coupons', 'irail','opendoar', 'viator','highcharts', 'geograph','eduroam', 'indeed','zoominfo', 'ookaboo','freebiesms', 'myvox','voxeo', 'jamendo','playme', 'elasticsearch', 'importio'

mapping カテゴリのクラスタが示されている．Music カテゴリと違い，mapping カテゴリはサービス連携関係ネットワークのみのベクトルだと正しく分類できるが，利用者提供者ネットワークも用いると正しく分類できないサービスがある．また，‘elasticsearch’ など，利用者提供者ネットワークも用いると mapping カテゴリとして誤って分類されてしまうサービスがある．このように mapping カテゴリでは，サービス連携ネットワークに加え，サービス提供利用ネットワークを用いることでクラスタに悪い影響を及ぼしているが，結果として正しく分類できているサービスの個数は増え，誤って分類されたサービスの個数は減っている．

このように，サービス連携ネットワークに加え，サービス提供利用ネットワークを用いることでクラスタの改善に繋がり，クラスタリング精度向上に寄与していることがわかる．しかし，このようにクラスタリング精度向上に寄与するような例ばかりではなく，クラスタリング結果が悪くなっている例もある．結果，クラスタリング精度向上が 6.6%にとどまっていると考えられる．

6.2 異種グラフの統合方法ごとのクラスタリング結果

5.2 において，異種グラフの統合方法ごとのクラスタリング精度が示されている．表 8 において，サービス共利用ネットワークを用いた場合のクラスタリング精度を各統合手法で比較した場合，Concat を用いた手法が最も精度が高くなっており，次いで Add2,Mul2 となっている．表 9 において，サービス共利用ネットワークを Anonymous 化した場合のクラスタリング精度を各統合手法で比較した場合，Add2 を用いた手法が最も精度が高くなっており，次いで Concat,Mul2 となっている．表 10 において，サービス提供利用ネットワークを利用した場合のクラスタリング精度を各統合手法で比較した場合，Concat を用いた手法が最も精度が高くなっており，次いで Add2,Mul2 となっている．表 11 において，サービス提供利用ネットワークを Anonymous 化した場合のクラスタリング精度を各統合手法で比較した場合，Add2 を用いた手法が最も精度が高くなっており，次いで Concat,Mul2 となっている．これらの結果をまとめると，提供者を利用したネットワークを anonymous 化した場合は Add2 の手法でサービスベクトルと提供者ベクトルを統合するのが有効的と言えるが，提供者を利用したネットワークを anonymous 化しない場合は，Concat の手法に

よりベクトルを統合するのが，最も有効的な手法だと言える．

第7章 まとめ

本研究は，従来のサービス説明文に基づくテキストベースのクラスタリング手法の代わりに提案された，複合サービスのサービス依存関係に基づくクラスタリング手法に加え，異種ネットワークを用いたクラスタリング手法を提案した．そして，異種ネットワークを用いた手法は，従来の複合サービスのサービス依存関係のみを用いてクラスタリングを行う手法よりも有効であることを示した．

本研究の貢献は以下の通りである．

サービス提供者利用者の情報を用いたネットワークの構築とサンプリング

サービス提供者利用者利用関係からサービス共利用ネットワーク，サービス提供利用ネットワークを構築し，サンプリングを行った．これらのネットワークから作成した提供者ベクトルを，後述の異種グラフの埋め込みの統合方法により，サービスベクトルと統合することによって，サービスのカテゴリごとのクラスタリング精度が向上した．

異種グラフの埋め込みの統合方法

サービスの利用関係のサンプリング結果から得られたベクトルに，提供者利用者利用関係のサンプリング結果から得られたベクトルを Concat，add2，mul2 の手法で異種グラフの埋め込みの統合を実現した．これらの手法の中で Concat の手法を用いることで，サービスクラスタリングの精度が最も高くなった．Concat により，サービスベクトルと提供者ベクトルを統合することによって，サービスのカテゴリごとのクラスタリングの精度が従来手法と比較し，6.6%向上した．

今後，実世界において複合サービスを構築する際，グラフ埋め込みを用いたラスタリング手法を用いて必要な機能を持つ原子サービスを検索する場合，ユーザーが類似機能サービスを検索して複合サービスを構築するのではなく，システムが自動で原子サービスの候補を挙げることが望ましいと考えられる．これを実現するためには，各クラスタが類似機能なクラスタであるだけでなく，入力データや出力データも近いものである必要がある．

従来手法ではサービス連携関係のみに注目してネットワークを構築し，サービスクラスタリングを行っていたが，本研究では提供者と利用者の情報にも注目して異種ネットワークを構築し，サービスクラスタリングを行った．今後，提

供者や利用者以外の情報にも注目し，新たなるネットワークを構築することでより高いクラスタリング精度を得られると考えられる．

謝辞

本研究を行うにあたり，熱心なご指導，ご助言を賜りました指導教官の村上陽平准教授並びに大久保弘基先輩に深謝申し上げます．また普段からお世話になっている社会知能研究室の皆さまにも感謝の意を表します．

参考文献

[1] Kemas Muslim Lhaksmana, Yohei Murakami, Toru Ishida, “Analysis of Large-Scale Service Network Tolerance to Cascading Failure, “IEEE Internet of Things Journal, Vol. 3, No. 6, pp. 1159-1170, 2016.

[2] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean “Efficient Estimation of Word Representations in Vector Space”, https://arxiv.org/abs/1301.3781

[3] Aditya Grover, Jure Leskovec: “node2vec: Scalable Feature Learning for Networks” ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), 2016

[4] Min Shi, Jianxun Liu, Dong Zhou, Mingdong Tang, and Buqing Cao, “WE-LDA: a word embeddings augmented LDA model for web services clustering,” in IEEE International Conference on Web Services (ICWS), 2017, pp. 9–16.

[5] Ling-liXie,Fu-zanChen,Ji-songKou:Ontology-basedsemanticwebservice clustering, Proceedings of IEEE 18th International Conference on Industrial Engineering and Engineering Management, pp. 2075-2079 (2011)

[6] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. ICLR Workshop, 2013.

[7] Guobing Zou, Zhen Qin, Qiang He, Pengwei Wang, Bofeng Zhang, Yanglan Gan: “DeepWSC: Clustering Web Services via Integrating Service Composability into Deep Semantic Features”, IEEE Transactions on Services Computing, 2020.

[8] AdityaGrover,JureLeskovec:node2vec:ScalableFeatureLearningfor Networks, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 855-864 (2016)

[9] Tom Kenter, Alexey Borisov, and Maarten de Rijke. Siamese CBOW: Optimizing word embeddings for sentence representations. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL’16).