

# 効率的な IaaS クラウドサービス発見のための オントロジクラスタリング手法

新穂一賢<sup>†</sup> 内林俊洋<sup>†</sup> アプドゥハン・ベーナディ<sup>†</sup>

**概要:** ユーザの要求に近い仮想マシンを提供する IaaS クラウドサービスを発見するためには多くの問題がある。その一つが、仮想マシンの CPU やメモリなどの名称が IaaS クラウドサービスの提供企業によって異なることがある。我々は、オントロジを使用して表現の違いを吸収することで問題解決を行った。しかし、オントロジが複雑化するとオントロジ内の検索がボトルネックとなる。そこで、オントロジ内をクラスタに分割して検索の高速化を行う。既存のクラスタリング手法を用いてクラスタリングを行ったが、意味を考慮したクラスタにならなかった。本稿は、オントロジ向けの新たなクラスタリング手法を提案し、その検証と考察を行う。

**キーワード:** クラウドコンピューティング, クラスタリング, オントロジ

## An Improved Ontology Clustering Scheme for Efficient IaaS Cloud Service Discovery

KAZUTOSHI NIHO<sup>†</sup> TOSHIHIRO UCHIBAYASHI<sup>†</sup>  
BERNADY O. APDUHAN<sup>†</sup>

### Abstract

A number of issues arise when trying to discover IaaS cloud services that best match to the user's virtual machine specifications' requirements. Among these, includes the different descriptions of CPU, memory, and other IaaS cloud services offered by different companies. This leads to the ambiguity in selecting the best match cloud services. Initially, we considered utilizing ontology technology and typical clustering method to narrow down the selection process. In this paper, we proposed an improved ontology clustering scheme, described the methodology, and discussed the verification results.

**Keywords:** Cloud Computing, Clustering, Ontology

## 1. 研究背景

意味を考慮した検索を行う技術の1つに、セマンティックウェブがある[1]。セマンティックウェブとは、コンピュータが自動的にコンピュータ間でやり取りを行うことができるようにするためのウェブ技術である。セマンティックウェブでは、データに意味情報(メタデータ)を与えることで、本当に必要な情報を検索することが可能になった。メタデータとは、データについての情報を記述したデータのことである。メタデータを位置づけするために RDF がある[2]。RDF の要素が示す意味は、人間には理解できるがコンピュータには理解できない。そこで、RDF をコンピュータに理解させるために、オントロジを利用する。オントロジとは、概念の階層構造や概念間の関係の意味を記述したものである。また、近年の情報化社会では、インターネットが普及して頻りにクラウドコンピューティングという言葉が聞くようになった。クラウドコンピューティングとは、コンピュータ処理やソフトウェア開発環境などを、ネットワーク経由で得られるサービスである。従来は、ソフトウェアや

データなどを手元のコンピュータで管理・利用していた。近年では、クラウドコンピューティングを利用して、ソフトウェアやデータなどを手元のコンピュータで管理・利用しなくてもよくなった。クラウドサービスの1つに、仮想マシンなどを提供する IaaS がある。IaaS のクラウドサービスは、同じサービスでも違う名前前で存在する。クラウドサービスとは、クラウドコンピューティングを利用するサービスのことである。同じサービスが、複数存在するため、オントロジを使って表現を統一する。扱う情報量が多くなるにつれてデータ量が増大し、オントロジを使った処理が非常に遅くなるため、クラスタにわけて検索する。クラスタにわけるためにアルゴリズムを用いてクラスタリングをする。クラスタリングとは、距離の近いものや類似度の高いもの同士をいくつかのクラスタに分割することである。クラスタリングを行うことにより、要素をクラスタにまとめて、検索に必要な時間を短縮することができる。

第2章は、目的を述べる。第3章は、クラウドコンピューティングについて述べる。第4章は、オントロジについて説明する。第5章は、クラスタリングについて説明する。

<sup>†</sup> 九州産業大学情報科学部情報科学科  
Faculty of Information Science, Kyushu Sangyo University

第6章は、既存のクラスタリングについて述べる。第7章は、類似度を用いたクラスタリングについて説明する。第8章は、提案手法の検証を行う。第9章は、今後の課題を述べる。第10章は、研究関連を述べる。

## 2. 目的

本研究の目的は、IaaS クラウドサービスを発見するためのオントロジの検索の高速化である。高速化を行うためにオントロジをクラスタリングするが、既存の手法では意味を考慮したクラスタにはならない問題があった。本稿は、意味を考慮したクラスタになるようなクラスタリング手法を提案し、検証と考察を行う。

## 3. クラウドコンピューティング

クラウドコンピューティングとは、ネットワーク上のサーバ群にあるデータやソフトウェアなどをユーザに提供する技術を表す言葉である。これにより、手元のコンピュータで管理・利用していたようなソフトウェアやデータなどをインターネットなどのネットワークを通じて、サーバ群から利用することができる。クラウドコンピューティングを用いたサービスのことをクラウドサービスと呼ぶ。

### 3.1 サービスモデル

クラウドコンピューティングのサービスモデルは大きく分けて3つあり、Software as a Service(SaaS)、Platform as a Service(PaaS)、Infrastructure as a Service(IaaS)である。SaaSは、利用者が、必要なソフトウェアや機能を必要なときに必要な分だけ利用できるサービスの形態である。PaaSは、アプリケーションソフトを実行するために、必要な実行環境のハードウェアやOSやミドルウェアをサービスとして提供する形態のことである。IaaSは、仮想化されたハードウェアやサーバを提供するサービスの形態である。

### 3.2 提供形態

クラウドコンピューティングの提供形態は、プライベートクラウド、パブリッククラウド、ハイブリッドクラウドがある。プライベートクラウドとは、企業などが自社内で利用するために構築したクラウドコンピューティング環境のことである。社員や内部の限定された利用者に向けて、リソースなどを提供することが可能になる。内部からのみ接続ができるので、セキュリティ面で優れている。パブリッククラウドとは、広く一般に提供するクラウドコンピューティング環境のことである。ハイブリッドクラウドとは、プライベートクラウドとパブリッククラウドを合わせたものである。

## 4. オントロジ

### 4.1 オントロジとは

各要素の明確な定義を行うとともに、概念間の関係を定義したものである。オントロジという言葉は、古代ギリシ

ヤ語で存在を意味する動詞の語幹である「オント」と、論を意味する「ロジ」を組み合わせた存在論を示す。哲学の分野では、「存在とは何か存在しているとはどういうことか」を問う学問である。情報学では、2000年にティム・バーナーズ・リーが提唱したセマンティックウェブにおけるアーキテクチャ階層の第4層として定義されたオントロジ層を示す。セマンティックウェブとは、データにタグを用いてデータ間の関係を定義し、オントロジを使用してコンピュータによる自動的な情報収集を目的としたものである。従来のウェブサイトをはじめとした文章検索は、単語単位での一致、または類義語を含む文章を検索するものである。オントロジを用いることで各文書に付加情報を持たせることにより、高度な検索を行うことができる。文書内にメタデータを付加することにより、検索対象となる文書が単語ではなく文書全体で大きな意味を持ったデータとして扱われる。メタデータとは、データについての情報を記述したデータのことである。

### 4.2 オントロジの構成

オントロジはオントロジ言語 (Web Ontology Language) で記述する[3]。オントロジは個体(インスタンス)、概念(要素)、属性、および関係で構成される。要素は、is-a関係によってできる階層化と要素と要素間の関係(property)で関連付けられる。図1は、オントロジの構成例を示したものである。Thingの下位はリソースとIaaSで構成される。リソースの下位は、HDDとCPUで構成される。IaaSの下位はServiceAとServiceBで構成される。HDDの下位は50GBで構成される。CPUの下位は2\_Coreと4\_Coreで構成される。例えば、ServiceAはIaaSに属しており、HDDの容量が50GB、CPUが4\_Coreのサービスである。ServiceBはIaaSに属しており、HDDの容量が50GB、CPUが2\_Coreのサービスである。

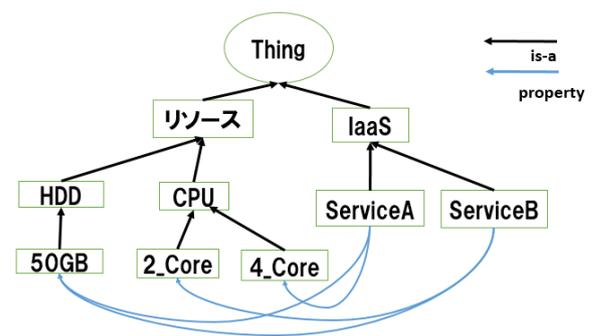


図1 オントロジの構成例

### 4.3 IaaS クラウドオントロジ

IaaS クラウドオントロジとは、Taekgyeong Hanらが提案[14]したIaaSクラウドサービスを、オントロジで構成したものである。我々は、IaaSクラウドオントロジに、SLAなどの要素を追加した。

図2はIaaSクラウドオントロジのソフトウェア、サービ

ス, リソースを示している. IaaS の下位は, InfraSoftware, IaaS Cloud Service, Computing Resource で構成される. InfraSoftware の下位は Application Environment, OS, VideoEncodingStreaming, WebHosting, Exec, BatchProcessing, Lib, Application Servers で構成される. IaaS Cloud Service 下位は, Domestic Provider, Overseas Provider で構成される. Computing Resource 下位は, Memory, RAM, Hourly Usage Rate, Monthly Usage Rate, HDD, VM, Resource Reservation, CPU clock(GHz), CPU で構成される.

図 3 は, DomesticProvider (IaaS 型のクラウドサービスを提供している国内のプロバイダ) を示している. 国内のプロバイダ数は 15 個ある. DomesticProvider の下位は, FUJITSU, Sakura Internet, ITOCHU Techno-Solutions, at+link, GMO Cloud, KDDI, IDC Frontier, NIFTY, Bit Isle, CEC, Internet Initiative, Japan Rudd, Softbank Telecom, Unit Hosting, NTTCommunications で構成される.

図 4 は, FUJITSU プロバイダとそのサービスを示している. FUJITSU の下位は, FUJITSUCloudInitiative で構成される. FUJITSUCloudInitiative の下位は, Double High, Advanced Type, High Performance Type, Economy Type, Quad High, Standard Type, FUJITSUCloudSLA で構成される. Double High の下位は, 178560JPY, 16GB, 16 Core, CentOS で構成される. Standard Type の下位は, CentOS, 32736JPY, 3.4GB, 2 Core で構成される. FUJITSUCloudSLA の下位は, 99.9Percent で構成される.

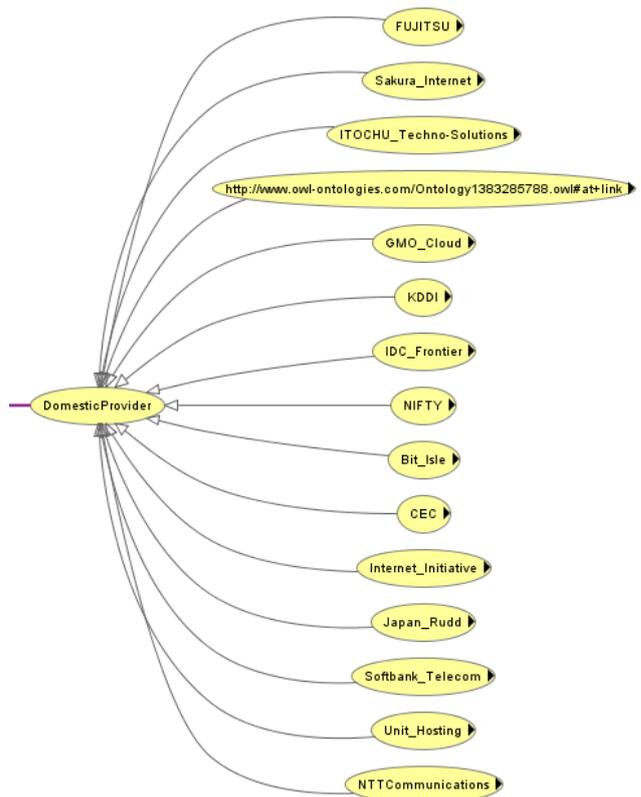


図 3 DomesticProvider 要素



図 2 IaaS クラウドオントロジの一部

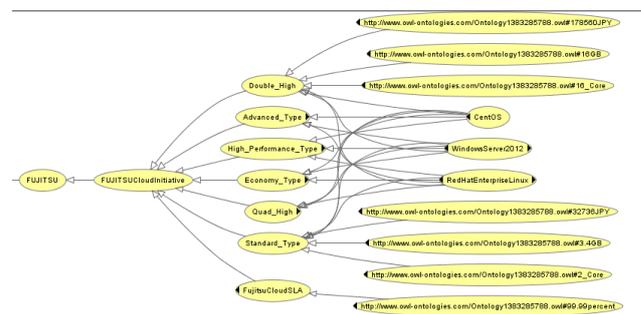


図 4 FUJITSU のプロバイダのオントロジの一部

## 5. クラスタリング

### 5.1 クラスタリングとは

与えられた要素間の座標や距離を基準として, いくつかのクラスタに分類する手段である. 要素をクラスタに分割していくことにより, 検索を高速化できる.

### 5.2 クラスタリング手法

クラスタリング手法を大きく分けると, 「階層的クラスタリング」と, 「非階層的クラスタリング」がある. また, 要素が 1つのクラスタのみに属することができる「ハードクラスタリング」と, 要素が複数のクラスタに属することができる「ソフトクラスタリング」がある.

### 5.2.1 階層的クラスタリング

N 個の要素がそれぞれ個別のクラスタに含まれた初期状態から始まる。要素  $x_1$  と  $x_2$  の距離  $d(x_1, x_2)$  からクラスタ間の距離  $d(C_1, C_2)$  を計算し、最も近いクラスタを順に合併する。クラスタが指定された数になるまで繰り返す。クラスタ間の距離  $d(C_1, C_2)$  の求め方のアルゴリズムの違いによって様々な手法がある。クラスタリング手法は、最長距離法、最短距離法、メジアン法、重心法、群平均法、ウォード法などがある。

### 5.2.2 非階層的クラスタリング

階層的クラスタリングと異なり、1 個のクラスタにすべての要素を含む初期状態から始まる。あらかじめいくつかのクラスタにわけると決めて、決めた数のクラスタに要素を分割する方法である。クラスタリング手法として k-means 法などがある。

## 6. 既存の手法を用いたクラスタリング

### 6.1 概要

IaaS クラウドオントロジを、既存のクラスタリング手法である最短距離法、最長距離法、メジアン法、重心法、群平均法、ウォード法を用いてクラスタリングする。既存の手法を用いてクラスタリングを行った際に、どのように要素がクラスタに分割されたのかを検証する。要素間の距離は、オントロジの類似度を用いる。本検証は、平成 26 年度の「オントロジとクラスタリングを用いた IaaS サービスの発見と選択の研究」の結果である[5]。

### 6.2 類似度

Andreasen らの研究[6]では、オントロジ内の要素の深さを利用した要素間の類似度を求めるアルゴリズムを提案している。式 1 は、要素  $x$  と要素  $y$  の類似度を求める式である。  $\text{sim}(x, y)$  は  $x$  と  $y$  の類似度を示している。  $\alpha(x)$  と  $\alpha(y)$  は要素の深さである。  $\alpha(x) \cap \alpha(y)$  は  $\alpha(x)$  と  $\alpha(y)$  の共通する親の深さを示している。

$$\text{sim}(x, y) = \frac{1}{2} \frac{|\alpha(x) \cap \alpha(y)|}{|\alpha(x)|} + \frac{1}{2} \frac{|\alpha(x) \cap \alpha(y)|}{|\alpha(y)|} \quad (\text{式 1})$$

式 1 を利用して、図 5 を例に挙げて計算する(式 2)。D と E の類似度を測る場合、 $x=D, y=E$  となる。 $x=D, y=E$  の共通の親が  $\alpha(x) \cap \alpha(y) = B$  である。A の深さが 1 で、B の深さが 2、C の深さが 3、D の深さが 4、E の深さが 5 である。よって、D と E の類似度は  $7/12$  となる。

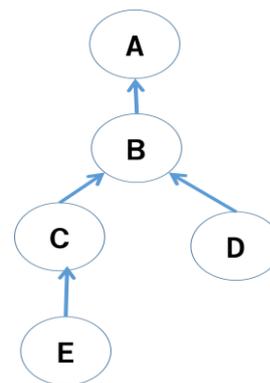


図 5 オントロジの要素

$$\text{sim}(D, E) = \frac{1}{2} \frac{|2|}{|4|} + \frac{1}{2} \frac{|2|}{|3|} = \frac{7}{12} \quad (\text{式 2})$$

### 6.3 検証方法

IaaS クラウドオントロジをクラスタリングする。クラスタリング結果を CSV ファイルに書き込む。クラスタ数は、用意した IaaS クラウドを提供しているプロバイダと同じ数である 20 個に設定する。

### 6.4 クラスタリング結果

図 6 は、IaaS クラウドオントロジをクラスタ数 20 個でクラスタリングしたものである。最短距離法とメジアン法と重心法はクラスタ 1 にほぼすべての要素が集まっている。最長距離法と群平均法も偏りがある。ウォード法が、一番偏りが少なく、均等に要素が分割されている。クラスタ数を 20 個に指定して、最短距離法、最長距離法、メジアン法、重心法、群平均法、ウォード法でクラスタリングを行った結果は図 7 のようになった。クラスタ 1 には、Copper\_4, Gold\_2, 14.2JPY, SUSE\_Linux\_Enterprise\_Server, Silver\_1 など 60 個の要素が入っている。クラスタ 2 には、Sakura\_Internet, 96000JPY, 5200JPY, Type\_6, 3GB, Type\_5, Type\_19, Type\_18 など 56 個の要素が入っている。クラスタ 3 には Type\_S, IDC\_Frontier, 90300JPY, 2.4GHz, Type\_L, Type\_M2, 7.9JPY, 84000JPY など 30 個の要素が入っている。

クラスタ数	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
最短距離法	628	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
最長距離法	59	50	34	47	24	38	32	125	47	32	28	17	14	9	15	20	11	19	22	4
メジアン法	628	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
重心法	628	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
群平均法	189	79	228	26	10	7	17	17	9	8	15	8	8	5	4	8	4	1	2	1
ウォード法	60	56	30	28	38	27	42	20	44	35	43	17	46	30	14	15	44	22	16	22

図 6 クラスタリング結果

```
Method: 6
Group 1
Copper_4 Gold_2 14.2JPY 4.1JPY SUSE_Linux_Enterprise_Server Silver_1 7.4
(60)

Group 2
Sakura_Internet 96000JPY 5200JPY Type_6 3GB Type_5 Type_19 Type_18 1
(56)

Group 3
Type_S IDC_Frontier 90300JPY 2.4GHz Type_L Type_M2 7.9JPY 84000JPY 6
(30)
```

図 7 ウォード法を用いたクラスタリング結果

## 6.5 問題点

最短距離法，群平均法，最長距離法，メジアン法，重心法は1つないし少数のクラスタに要素が固まっている．ウォード法は，各クラスタに要素が均等に分割されている．しかし，各クラスタ内の要素が意味を考慮していないことが問題である．

## 7. 類似度を利用したクラスタリング

### 7.1 提案手法

既存の手法を用いた階層的クラスタリングでは，意味を考慮したクラスタになっていない問題があった．そこで，非階層的クラスタリングを用いて IaaS クラウドオントログをクラスタリングする手法を提案する．

### 7.2 クラスタリングの流れ

すべての要素を1つのクラスタにまとめる．空のクラスタを1つ作る．最も類似度の低い要素の組を探す．片方の要素を空のクラスタへ追加する．要素の数/2 だけ繰り返す．同様の作業を繰り返して，2つのクラスタを4つのクラスタにわけける．この流れを指定したクラスタ数になるまで繰り返していく．クラスタ  $n$  とクラスタ  $n-1$  は類似度が高くなる．

手順：

1. すべての要素を1つのクラスタにまとめる．
2. 新しい空のクラスタを作る．
3. 最も類似度の低い要素の組を探す．片方の要素を空のクラスタに移す．
4. 要素の数/2 だけ繰り返す．
5. 指定したクラスタ数になるまで2から4を繰り返す

図8は，要素数16のオントログに提案手法を用いて4つのクラスタに分割した例である．すべての要素を1つのクラスタ1にまとめたものである．空のクラスタ2を作る．クラスタ内で，最も類似度が低い要素の組を探して，片方の要素を空のクラスタ2に移す．クラスタ1に入っていたすべての要素を要素の数/2 だけ繰り返す．分割が終わると，空のクラスタ3と4を作る．指定したクラスタ数になるまで最も類似度が低い要素の組を探し，片方の要素を空クラスタ3と4に移す．指定したクラスタ数4になると終了する．

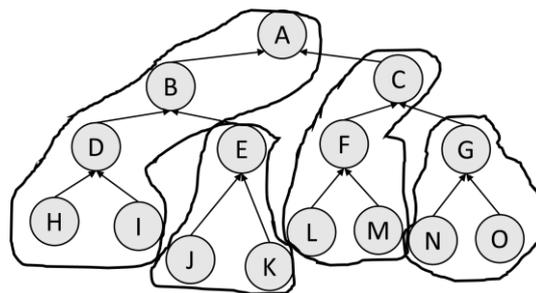


図8 クラスタリング例

## 8. 提案手法の検証

### 8.1 検証方法

提案したクラスタリング手法を使用して，IaaS クラウドオントログをクラスタリングする．クラスタリングした結果とウォード法を用いてクラスタリングした結果と比較する．また，クラスタを利用しない場合と提案手法を使用したクラスタを利用した場合の検索速度を比較する．

### 8.2 検証環境

検証には，OS が Windows 10，プロセッサが Intel(R) Core(TM) i7-4770 CPU 3.40GHz，メモリが 8.0GB，のコンピュータを使用する（表1）．

表1 使用するコンピュータ

OS	Windows 10
プロセッサ	Intel(R)Core(TM)i7-4770CPU@3.40GHz
実装メモリ(RAM)	8.0GB

### 8.3 結果

図9は，ウォード法を使用したクラスタリングの結果の一部である．クラスタ19に要素が偏り43個となっていることがわかる．

```

Group 19
1,5GB V15LC 1.5_Core Internet_Initiative V15LR 8000.PY
(6)
Group 19
Gold_2 4.1.PY Silver_1 1.250Hz Yes_Resource_Reservation 1084GB 45.2.PY Copper_3 Copper_2 Gold_1 33.6.PY 1.350SD 38.8.PY
Copper_1 Linux_Any 0.680SD 21.0.PY Platinum_2 16.8.PY Bronze_4 225GB 74.6.PY 0.060SD 0.800SD 5.3.PY 8.9.PY Gold_4
Silver_4 Platinum_1 0.360SD Bronze_3 66.2.PY 0.030SD 0.120SD 1.800SD 11.8.PY 72.5.PY 910GB 32.6.PY Silver_2
No_Resource_Reservation Bronze_2 Bronze_1
(43)
Group 20
30-NEXT_Virtual_Server_Type2 Availability 38.3.PY 75000.PY Bit_Isle 45000.PY Server_Dr-demand_NEXT 40GB Server_Dr-
demand_NEXT_3Lk 30-NEXT_Virtual_Server_Type2
(10)

```

図9 ウォード法の結果 (1)

図10は，提案手法の結果の一部である．クラスタ18, 19, 20のすべてにおいて，偏りが少ないことが分かる．

```
Cluster 18: High_Memory_2VL Silver_4 Gold_3 Platinum_1 Bronze_3 Type_1 Standard 0.5GB 64500JPY 13600JPY
Cluster 19: IBM_SLA Medium_Type20 Spec_Type18 Type_12 Type_M 11J_G10_Component_Nation_Service 18800JPY 182000JPY 5300JPY
Cluster 20: IBMPowerSystemsOS MaxOS BatchProcessing Exec ApplicationServers Option_8 Option_7 5250JPY 290000JPY 25000JPY
```

図 10 提案手法の結果 (1)

図 11 は、ウォード法の結果の一部である。クラスタに含まれる要素も少なく、要素の意味を考慮したクラスタになっていないことがわかる。

```
Group 62
  x86Option VM
  (2)
Group 63
  Type_XS 0.8GHz 3000JPY
  (3)
Group 64
  Industria High_Performance_Medium Industria_SLA 16GB 17000JPY Japan_Rudd
  (6)
```

図 11 ウォード法の結果 (2)

図 12 は、提案手法の結果の一部である。クラスタ 31 は OS の Windows 関連の要素が集まっている。クラスタ 31 とクラスタ 32 は類似度が高いクラスタとなっている。

```
Cluster 31: WindowsServer2008 Android Windows6 WindowsServer2008 Windows2000 Windows6 Windows6 WindowsXP Windows OS: OS
Cluster 32: MobileLinux EmbeddedLinux PalmOS IbmAIX6.1 Linux WindowsE iPhoneOS Unix Embedded IBMPowerSystemsOS OS
Cluster 33: 60GB 64bit S AmazonEC2SLA 1200GB 788.40USD Ubuntu 209.0JPY 29.4JPY 100+Large Hourly_Usage_Rate OS HDD
```

図 12 提案手法の結果 (2)

提案手法は、クラスタ数を 16 個, 32 個, 64 個, 128 個に指定して検証を行った。クラスタ数が 16 個, 32 個, 64 個, 128 個すべてに偏りは起きなかった。クラスタ数が 128 個だと、クラスタ内の要素が少ないので、判別できなかった。

表 2 から表 4 は、ウォード法と提案手法を比較したものである。ウォード法は、クラスタ内の要素数にばらつきあり、偏りが起きている。提案手法は、クラスタ内の要素数にばらつきがなく、偏りが起きていない。

表 2 ウォード法と提案手法の結果表示 (1)

クラスタ数	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
ウォード法	9	15	4	16	2	10	9	16	14	5	20	32	4	7	43	13	15	6	43	10
提案手法	10	10	10	10	10	10	11	10	10	10	10	10	10	11	10	10	10	10	11	10

表 3 ウォード法と提案手法の結果表示 (2)

クラスタ数	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
ウォード法	7	6	9	10	11	15	8	12	25	9	22	8	4	8	10	11	4	10	5	6
提案手法	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	11

表 4 ウォード法と提案手法の結果表示 (3)

クラスタ数	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
ウォード法	12	4	4	19	12	3	22	5	5	4	5	4	4	3	4	5	4	12	5	3	4	2	3	6
提案手法	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	11

次に、検索速度の比較を行う。検索の条件は、「CPU が 2\_Core」、「OS が CentOS」、「Memory が 512MB」、「HDD が 50GB」である。図 13 は、クラスタを利用しない場合の結果である。〈Point〉の列の 3 は、CEC 社のサービス Option\_6, Option\_5, Option\_4, Option\_1 と伊藤忠テクノソリューションズ社のサービス Specification\_2 がユーザの要求した条件に 3 つ当てはまっていることを示している。Point が高いほどユーザが求めているサービスに近づいていく。次に Microsoft 社のサービス M が 2Point を示している。よって、ユーザが要求した要素に一番近いサービスは、CEC 社の Option 6, Option 5, Option 4, Option 1, ITOCHU TechnoSolutions 社の Specification 2 である。次に、ユーザの要求に近いサービスは、Microsoft 社の M, GOGRID 社の Small などの Point の一致が 2 を示しているサービスである。ユーザの要求に遠いサービスは、Microsoft 社の XS, IBM 社の Bronze 1 などの Point の一致が 1 を示しているサービスである。

図 14 は、提案手法を使用したクラスタを利用して検索した結果を示している。検索結果はクラスタを利用しない場合と同じになる。

<Ponit>	provider	Service
3	CEC	Option_6
3	CEC	Option_5
3	CEC	Option_4
3	CEC	Option_1
3	ITOCHU_TechnoSolutions	Specif
2	Microsoft	M
2	GOGRID	Small
2	GOGRID	Medium
2	at+link	at+link_Case4
2	at+link	at+link_Case3

図 13 クラスタリングなしの検索結果 (一部)

<Ponit>	provider	Service
3	CEC	Option_6
3	CEC	Option_5
3	CEC	Option_4
3	CEC	Option_1
3	ITOCHU_TechnoSolutions	Specif
2	Microsoft	M
2	GOGRID	Small
2	GOGRID	Medium
2	at+link	at+link_Case4
2	at+link	at+link_Case3

図 14 類似度+クラスタリングありの検索結果 (一部)

図 15 は、CEC 社のサービスである Option\_6 を示している。ユーザが要求して条件に一致したものは、「CPU が 2\_Core」、「OS が CentOS」、「HDD が 50GB」である。よって、一致が 3 であることがわかる。

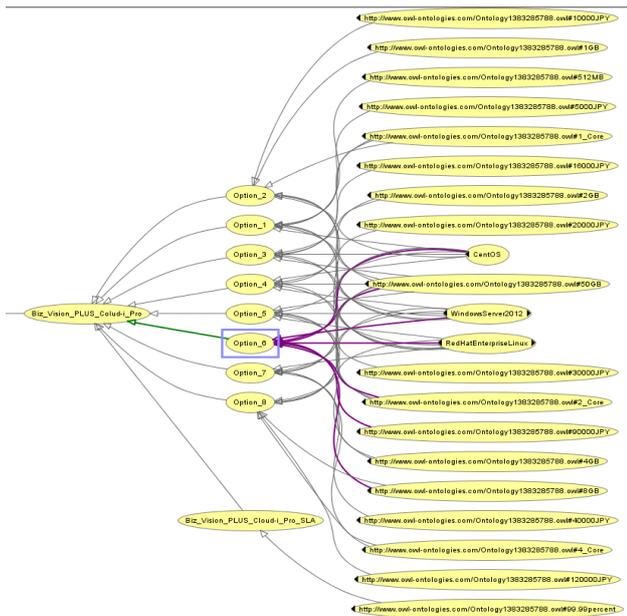


図 15 CEC のオントロジ (Option\_6)

図 16 は、伊藤忠テクノソリューションズ社のサービスである Specification\_2 を示している。ユーザが要求して条件に一致したものは、「CPU が 2\_Core」、「OS が CentOS」、「HDD が 50GB」である。よって、一致が 3 であることがわかる。

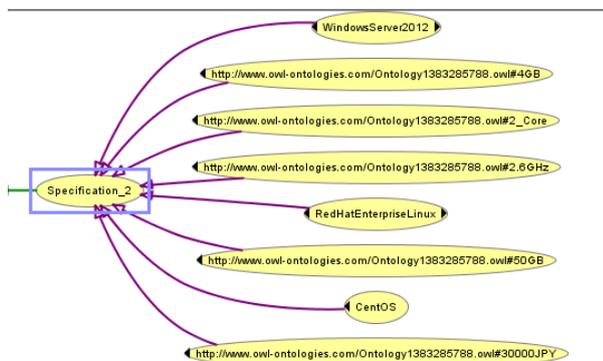


図 16 ITOCHU のオントロジ (Specification\_2)

図 17 は、検索を 50 回行った平均の検索速度をグラフにしたものがある。クラスタを利用しない場合の検索速度の平均は、88.69ms である。クラスタを利用した場合の検索速度の平均は、47.56ms である。クラスタを利用した検索のほうが、約 2 倍速くなっていることが分かる。

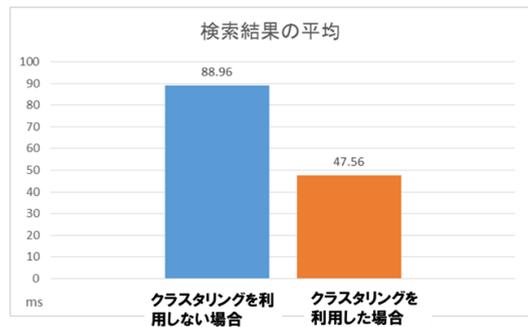


図 17 クラスタリングを利用しない場合と利用した場合の検索結果

## 8.4 考察

要素に対してクラスタ数は多すぎるとクラスタ内に入る要素が少なくなる。これにより、要素をクラスタ内にかけているだけで意味を考慮したクラスタにならない。また、クラスタ数が少なすぎるとクラスタ内に多くの要素が入ることになる。要素が多くなるにつれてクラスタ数も多くなっていくが、クラスタ数が多すぎても意味を考慮したクラスタではなくなる。適切なクラスタ数を見極める必要がある。また、クラスタを利用しない場合とクラスタを利用した場合の検索速度の比較を行い、クラスタを利用した検索のほうが高速であることが分かった。

## 9. まとめと今後の課題

本研究は、IaaS クラウドサービスを発見するためのオントロジの検索の高速化のためのクラスタリング手法の提案を行った。既存の手法を用いたクラスタリングとは異なり、要素の意味を考慮したクラスタに分割する。提案手法は、IaaS クラウドオントロジをクラスタリングしたものに比べて、偏りのないクラスタになった。既存の手法に比べて、要素の意味を考慮したクラスタに変更することができたが、まだ完全ではない。オントロジとクラスタを用いてユーザの要求を満たすサービスの検索の高速化を行うことは達成できたと言える。

これまで、ハードクラスタリングで検証を行ってきたが、ソフトクラスタリングで検証してみたほうがいいのではないかと考える。なぜなら、要素は 1 つのクラスタにしか入らないものと、複数のクラスタに入るものがあるからである。要素の意味を考慮したクラスタを作る手法として、ソフトクラスタリングの Fuzzy - c - means 法などがある。要素の意味を考慮したクラスタを作ることができれば、さらに正確な検索結果を表示できると考える。

## 10. 関連研究

丹羽 智史らの研究[7]では、Folksonomy を対象としたタグのクラスタリングを行っている。Folksonomy は、Social Tagging の一種で、管理者ではなくユーザがタグ付けをして

文書を分類する。非常に多くのタグが存在するため、クラスタリングを行うことで検索の精度を高めることを目的としている。タグ間の相関度と対立度を求めてクラスタリングに使用している。クラスタリングは非階層的クラスタリングを使用している。本研究と非階層的クラスタリングを用いている点は同じだが、クラスタリングの対象が Folksonomy という点が異なる。

## 参考文献

- [1] T.Berners-Lee, J.Hendler, and O.Lassila, "The Semantic Web". Scientific American.284(5), pp.44-55, 2001.
- [2] "RDF Scheme", <http://www.w3.org/TR/rdf-schema/>.
- [3] "W3C : OWL Web Ontology Language Overview", <http://www.w3.org/TR/owl-features/>.
- [4] Taekgyeong, Han, Kwang Mong Sim, "An Ontology-enhanced Cloud Service Discovery System", International MultiConference of Engineers and Computer Scientists (IMEC 2010), Hong Kong, pp. 644-649, 2010.
- [5] T. Uchibayashi, B. O. Aduhan, N. Shiratori, "Towards a cloud ontology clustering mechanism to enhance IaaS service discovery and selection, The 2015 International Conference on Computational Science and Its Applications (ICCSA 2015)" pp545-556, 2015.
- [6] Troels Andreasen, Henrik Bulskov, Rasmus Knappe, "From Ontology over Similarity to Query Evaluation", in Proceedings of the 2nd CoLogNET-ElsNET Symposium—Questions and Answers: Theoretical and Applied Perspectives, R. Bernardi and M. Moortgat, Eds., pp. 39–50, 2003.
- [7] 丹羽 智史, 土肥 拓生, 本位田真一, "Folksonomy の 3 部グラフ構造を利用したタグクラスタリング", セマンティックウェブとオントロジー研究会, 2006.